




Онлайн-образование

Не забыть включить запись!





Меня хорошо видно && слышно?

Ставьте  , если все хорошо
Напишите в чат, если есть проблемы

Правила вебинара



Активно участвуем



Задаем вопрос в чат или голосом



Off-topic обсуждаем в Slack #канал группы или #general



Вопросы вижу в чате, могу ответить не сразу



Сетевые пакеты, VLAN, LACP

Викирюк Павел

Системный инженер

Маршрут вебинара

Отладка сетевого стека в Linux



VLAN



LACP

Цели занятия | После занятия вы сможете

1 Более тонко настраивать параметры сетевого стека в ядре Linux

2 Понимать, что такое VLAN и в каких случаях его нужно использовать

3 Понимать, что такое LACP, а также различия между bonding и teaming

СМЫСЛ | Зачем вам это уметь

1 Для более глубоких знаний сетевого стека вообще, и применительно к ОС Linux - в частности

2 Чтобы эффективно строить сетевую инфраструктуру с максимальной стабильностью и безопасностью

3 Чтобы обеспечивать балансировку и отказоустойчивость трафика в том числе на транспортном уровне



Отладка сетевого стека в Linux

Настройка параметров ядра

<https://www.kernel.org/doc/Documentation/networking/ip-sysctl.txt>

1. ARP отвечает по любому интерфейсу, чтобы отключить это:

```
sysctl -w net.ipv4.conf.all.arp_filter = 1  
sysctl -w net.ipv4.conf.all.arp_ignore = 2
```

2. Проверка обратного адреса. Если пакет пришел не с того интерфейса, на который указывает FIB (таблица маршрутизации), то пакет отбрасывается. При асимметричном роутинге надо отключать (но только в таком случае!):

```
sysctl -w net.ipv4.conf.<интерфейс>.rp_filter = 2
```

3. По-умолчанию включена поддержка Zeroconf, чтобы отключить:

```
echo NOZEROCONF=yes >> /etc/sysconfig/network
```




IP MTU / TCP MSS

IP MTU / TCP MSS

Вручную проверяем подбором значения размера пакета.
Linux всегда ставит DF (don't fragment) бит:

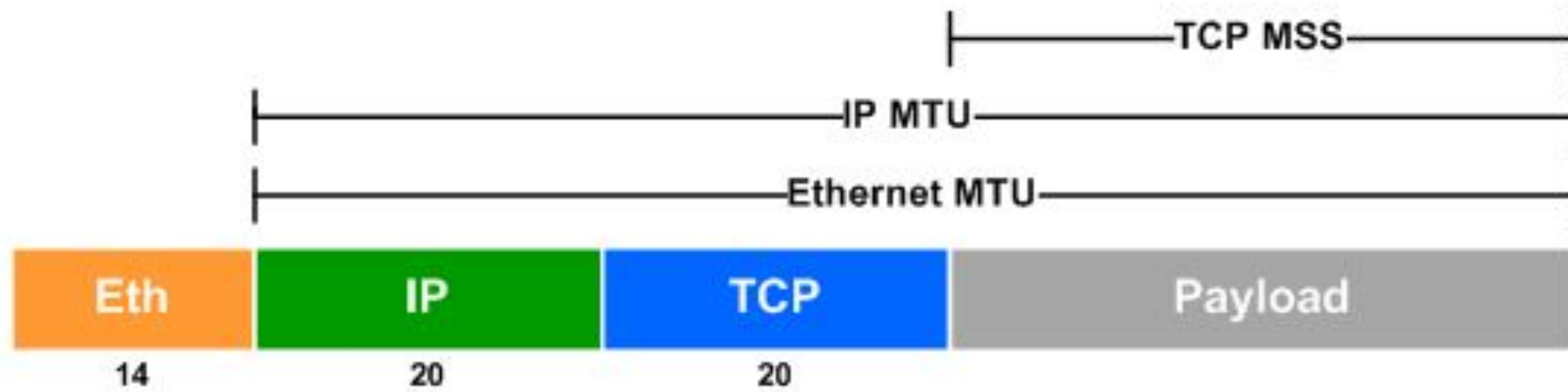
```
ping -s <size>
```

Ядро само определяет и запоминает PATH MTU (алгоритм PMTUD, потому и всегда DF установлен) для каждого элемента FIB.
Но иногда ему надо помогать:

1. Выставить MTU на интерфейсе
2. Уменьшить TCP MSS

```
iptables -I FORWARD -p tcp --tcp-flags SYN,RST SYN -j TCPMSS  
--clamp-mss-to-pmtu
```


IP MTU / TCP MSS



The background of the slide is an aerial photograph of a city skyline, likely New York City, with numerous skyscrapers. The image is overlaid with a semi-transparent blue layer. A network of white lines and dots, resembling a globe or a data network, is visible across the blue area. The title text is centered in this blue area.

TCP Window Size / Window Scaling

TCP Window Size / Window Scaling

На очень плохих каналах можно сразу отключить:

```
net.ipv4.tcp_window_scaling = 0
```

Косвенно на window scaling влияет параметр:

```
/proc/sys/net/ipv4/tcp_rmem
```

net.ipv4.tcp_wmem, net.ipv4.tcp_rmem — настройки Read и Write буферов выглядят одинаково. Это три числа в БАЙТАХ, “min default max” — минимальный гарантированный размер буфера, размер по умолчанию и максимальный размер, больше которого система не даст буферу вырасти

The background of the slide features an aerial view of a city skyline, likely New York City, with numerous skyscrapers. The image is overlaid with a semi-transparent blue and teal gradient. A network of white lines and dots is visible, particularly on the left side, suggesting a digital or technological theme.

Socket backlog

Socket backlog

<https://linux.die.net/man/2/listen>

/proc/sys/net/core/somaxconn - размер очереди установленных соединений ожидающих обработки accept().

/proc/sys/net/ipv4/tcp_max_syn_backlog - размер очереди не установленных соединений



Time wait

Time wait

Период ожидания потерявшихся пакетов, когда сокет еще не закрыт. По-умолчанию = $2 * \text{MSL} = 1$ минута. На высоконагруженных системах можно уменьшить до 10-15 секунд, меньше не стоит.

```
/proc/sys/net/ipv4/tcp_fin_timeout
```

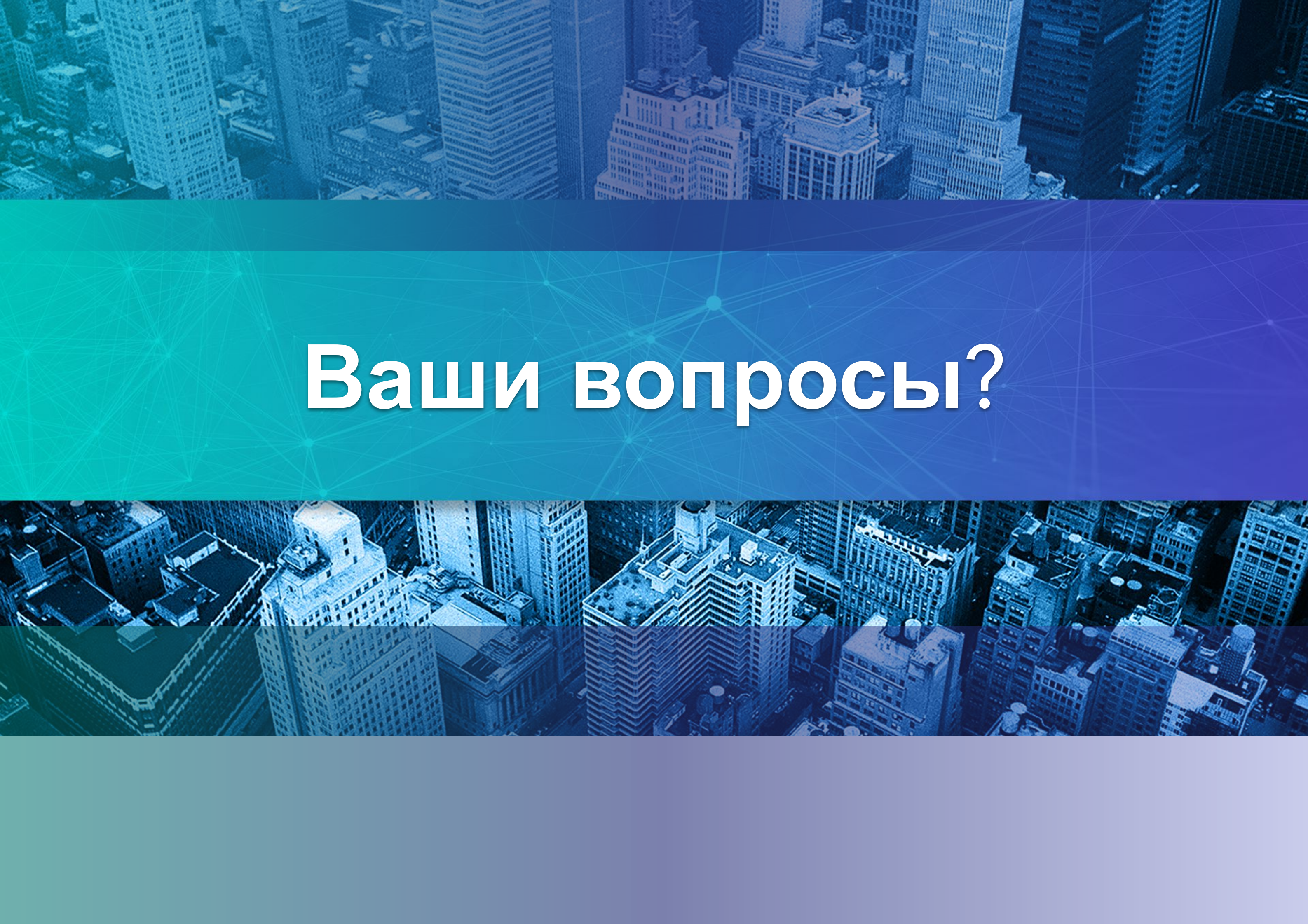
Если вы всё равно испытываете трудности, то стоит добавить еще адресов для приёма и обработки трафика

Параметр, позволяющий переиспользовать сокеты для исходящих соединений

```
net.ipv4.tcp_tw_reuse = 0
```

На балансировщиках стоит еще увеличить диапазон портов, доступных для исходящих соединений:

```
/proc/sys/net/ipv4/ip_local_port_range
```

Ваши вопросы?

Маршрут вебинара

Отладка сетевого стека в Linux



VLAN



LACP



VLAN



Вопрос к аудитории:

Что такое VLAN?

VLAN

VLAN (аббр. от англ. **Virtual Local Area Network**) — топологическая («виртуальная») локальная компьютерная сеть, представляет собой группу хостов с общим набором требований, которые взаимодействуют так, как если бы они были подключены к широковещательному домену, независимо от их физического местонахождения

<https://ru.wikipedia.org/wiki/VLAN>

Более простой вариант определения:

VLAN - механизм для создания логической топологии сети, не зависящей от ее физической топологии

Необходимость применения VLAN:

- изоляция сегментов сети
- гибкое разделение хостов на группы
- сокращение широковещательного трафика
- увеличение безопасности и управляемости сети

Особенности:

- не требуется физическое перемещение устройств
- требуется использование коммутаторов с поддержкой VLAN

VLAN

IEEE 802.1Q — открытый стандарт, который описывает процедуру тегирования трафика для передачи информации о принадлежности к VLAN по сетям стандарта IEEE 802.3 Ethernet

https://ru.wikipedia.org/wiki/IEEE_802.1Q

Особенности:

- используется процедура тегирования трафика

VLAN

IEEE 802.1Q — открытый стандарт, который описывает процедуру тегирования трафика для передачи информации о принадлежности к VLAN по сетям стандарта IEEE 802.3 Ethernet

https://ru.wikipedia.org/wiki/IEEE_802.1Q

Особенности:

- используется процедура тегирования трафика
- тэги инкапсулируются в ethernet кадра

VLAN

IEEE 802.1Q – открытый стандарт, который описывает процедуру тегирования трафика для передачи информации о принадлежности к VLAN по сетям стандарта IEEE 802.3 Ethernet

https://ru.wikipedia.org/wiki/IEEE_802.1Q

Особенности:

- используется процедура тегирования трафика
- тэги инкапсулируются в ethernet кадра
- поле для **Vlan ID** (VID) в тэге - всего 12 бит

VLAN

IEEE 802.1Q – открытый стандарт, который описывает процедуру тегирования трафика для передачи информации о принадлежности к VLAN по сетям стандарта IEEE 802.3 Ethernet

https://ru.wikipedia.org/wiki/IEEE_802.1Q

Особенности:

- используется процедура тегирования трафика
- тэги инкапсулируются в ethernet кадра
- поле для **Vlan ID** (VID) в тэге - всего 12 бит
- после добавления тэга пересчитывается контрольная сумма кадра

VLAN

IEEE 802.1Q – открытый стандарт, который описывает процедуру тегирования трафика для передачи информации о принадлежности к VLAN по сетям стандарта IEEE 802.3 Ethernet

https://ru.wikipedia.org/wiki/IEEE_802.1Q

Особенности:

- используется процедура тегирования трафика
- тэги инкапсулируются в ethernet кадра
- поле для **Vlan ID** (VID) в тэге - всего 12 бит
- после добавления тэга пересчитывается контрольная сумма кадра
- максимальное количество VID - **4096**, а точнее 4094, так как VID **0** и **4095** зарезервированы и не используются

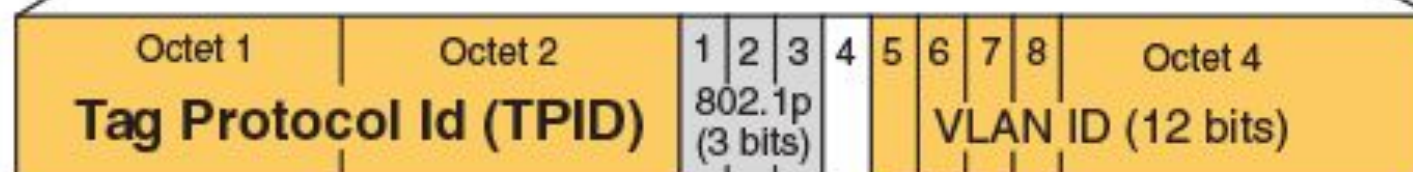
VLAN

Untagged Packet Format

6 bytes Destination Address	6 bytes Source Address	2 bytes Type Field	Up to 1500 bytes Data Field	4 bytes CRC	Ethernet II
6 bytes Destination Address	6 bytes Source Address	2 bytes Length Field	Up to 1496 bytes Data Field	4 bytes CRC	IEEE 802.3

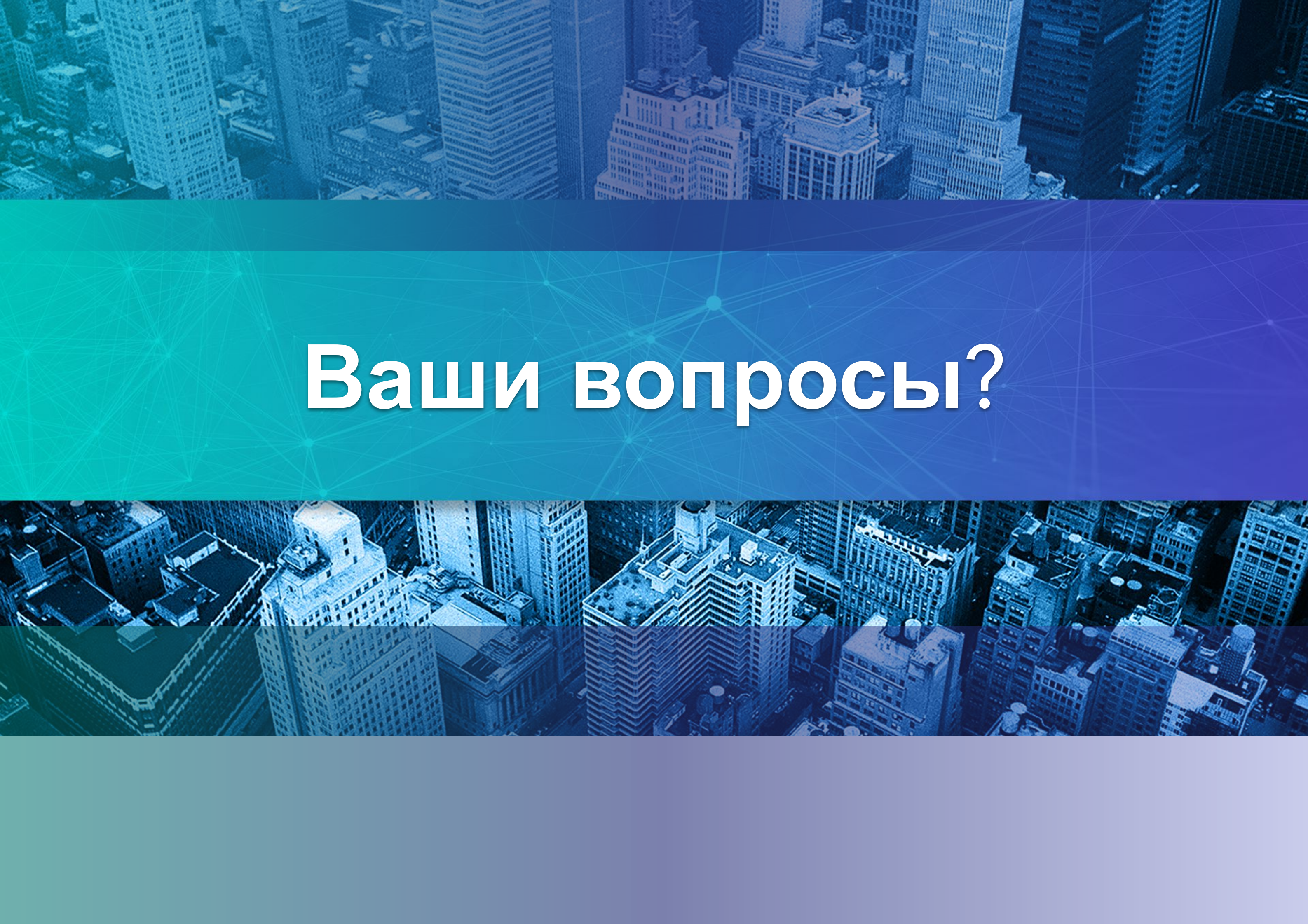
802.1q Tagged Packet Format

6 bytes Destination Address	6 bytes Source Address	4 bytes 802.1q Tag	2 bytes Type Field	Up to 1500 bytes Data Field	4 bytes CRC	Ethernet II with 802.1q tag
6 bytes Destination Address	6 bytes Source Address	4 bytes 802.1q Tag	2 bytes Length Field	Up to 1496 bytes Data Field	4 bytes CRC	IEEE 802.3 with 802.1q tag



Принцип работы:

1. Заголовок 802.1q размером 32 бита добавляется внутрь заголовка ethernet фрейма и увеличивает размер фрейма на 4 октета (32 бит)
2. Заголовок 802.1q сдвигает поле **Type**, в котором находится номер протокола, благодаря чему тэгированные пакеты могут обрабатываться неуправляемыми коммутаторами
3. При посылке через vlan-интерфейс и передаче в родительский интерфейс к фрейму добавляется 802.1q заголовок и контрольная сумма фрейма пересчитывается
4. При приеме фрейма на интерфейс система смотрит наличие у него 802.1q заголовка и, если нет, отправляет его на физический интерфейс (**native vlan**), а если есть - в соответствующий vlan-интерфейс (**trunk**)

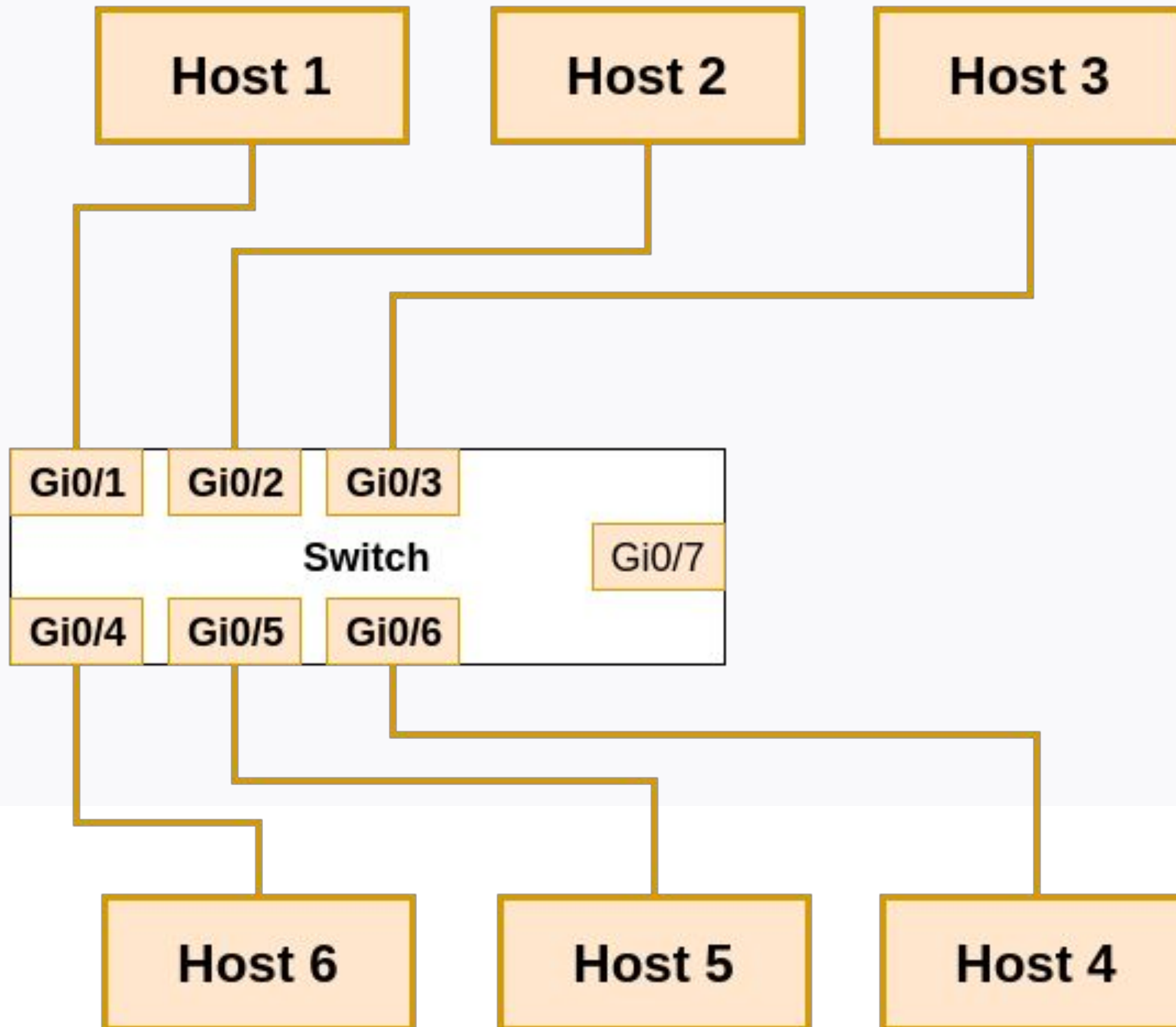


Ваши вопросы?

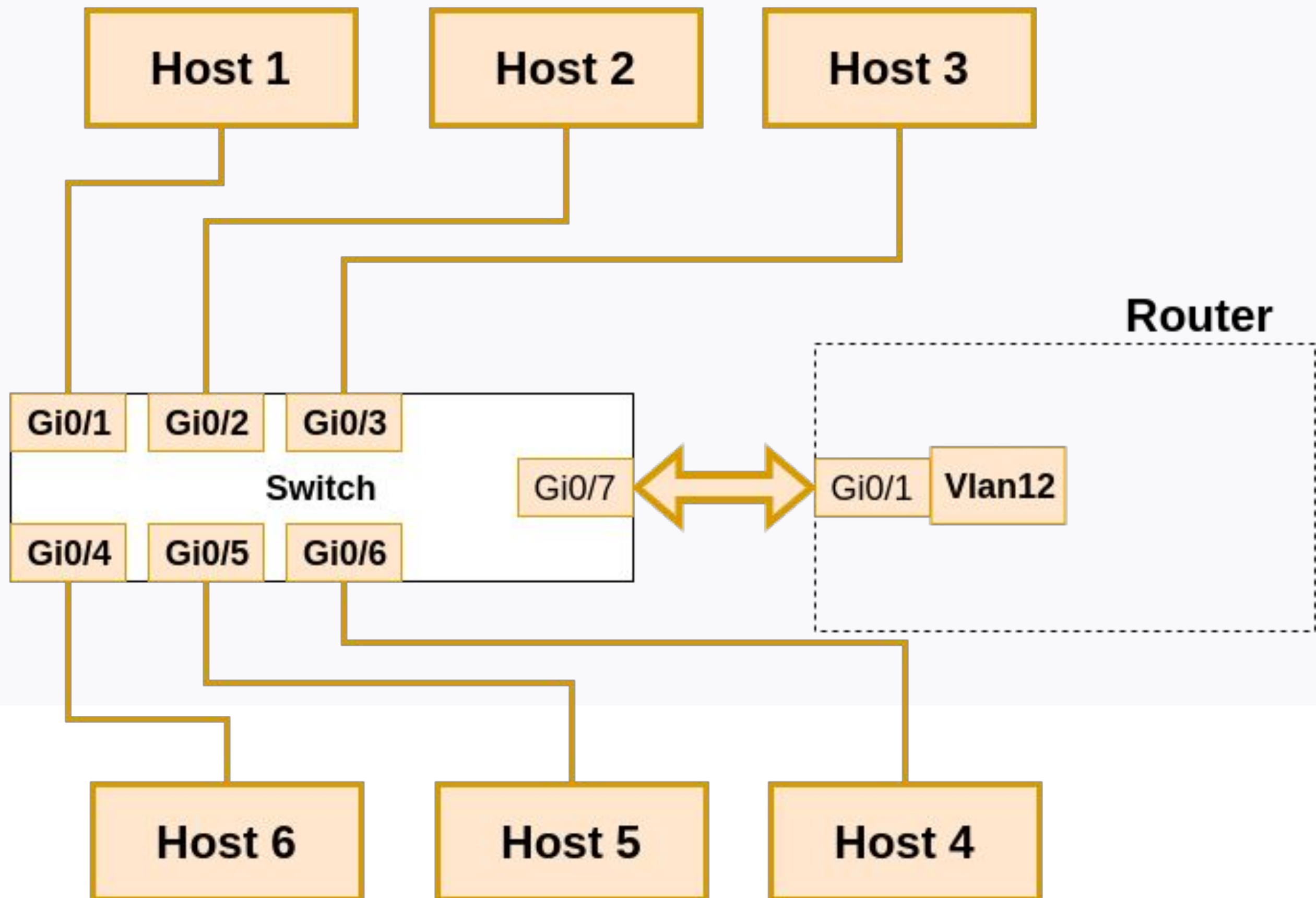


VLAN: режимы портов

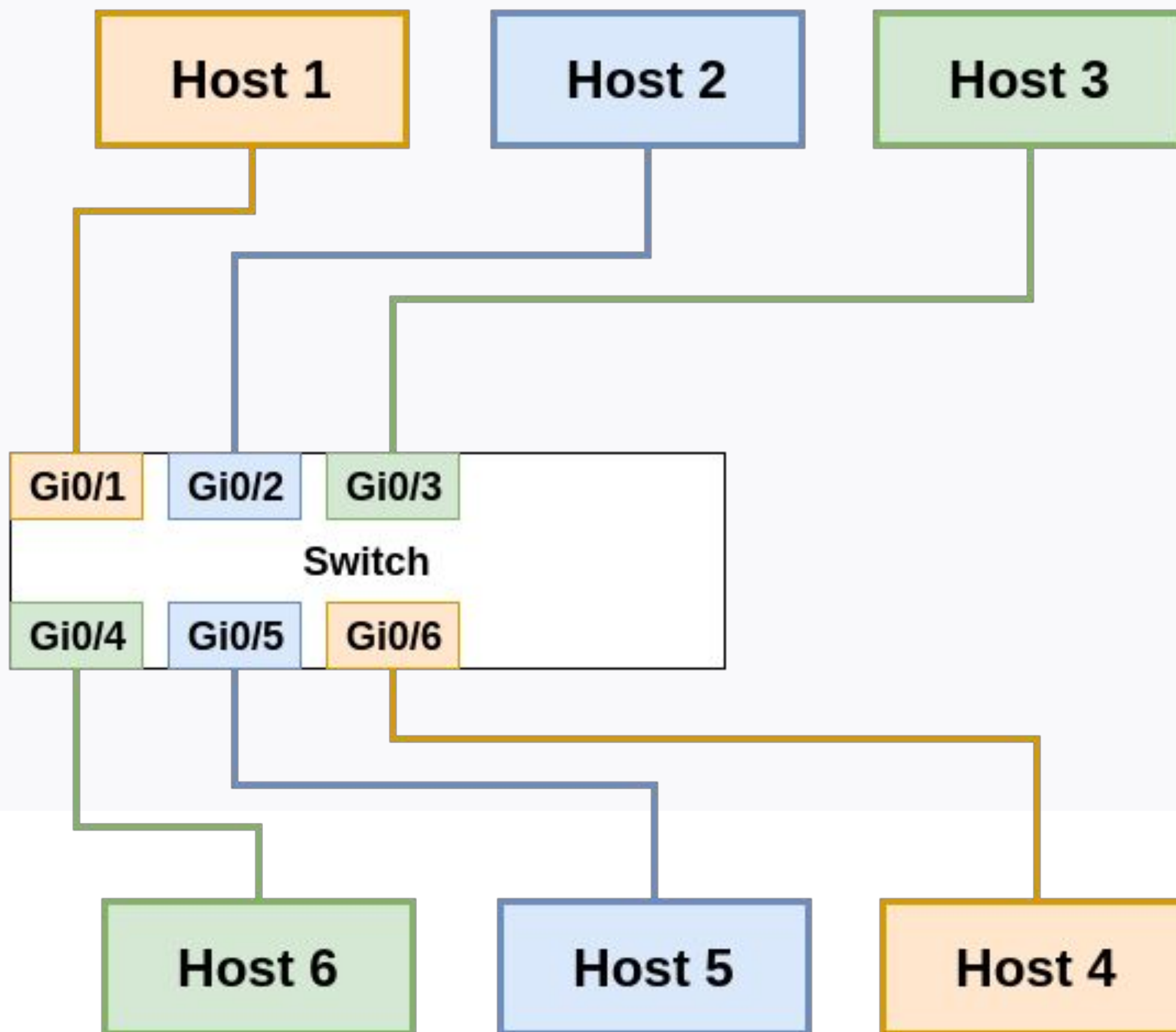
VLAN: режимы портов



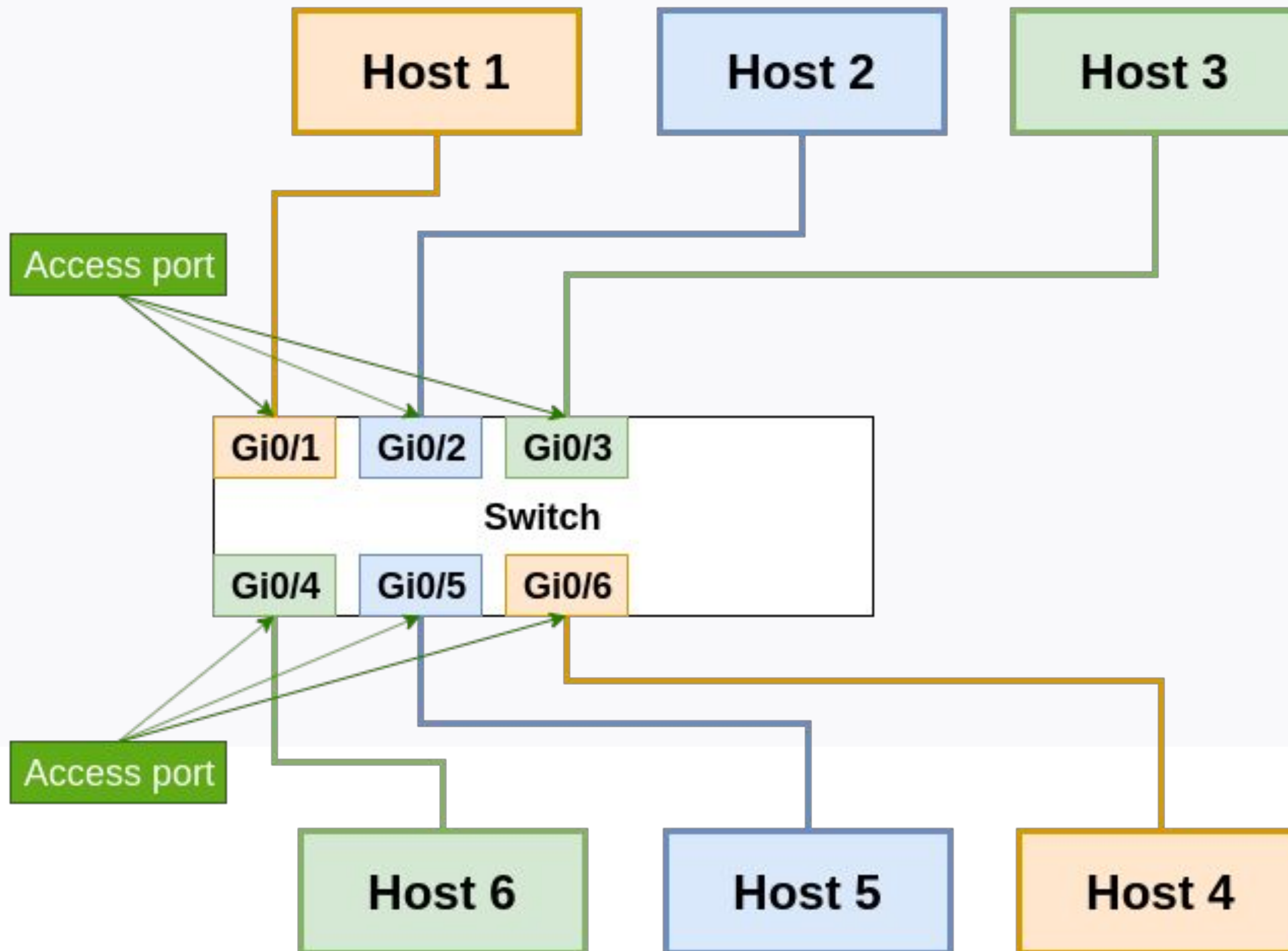
VLAN: режимы портов



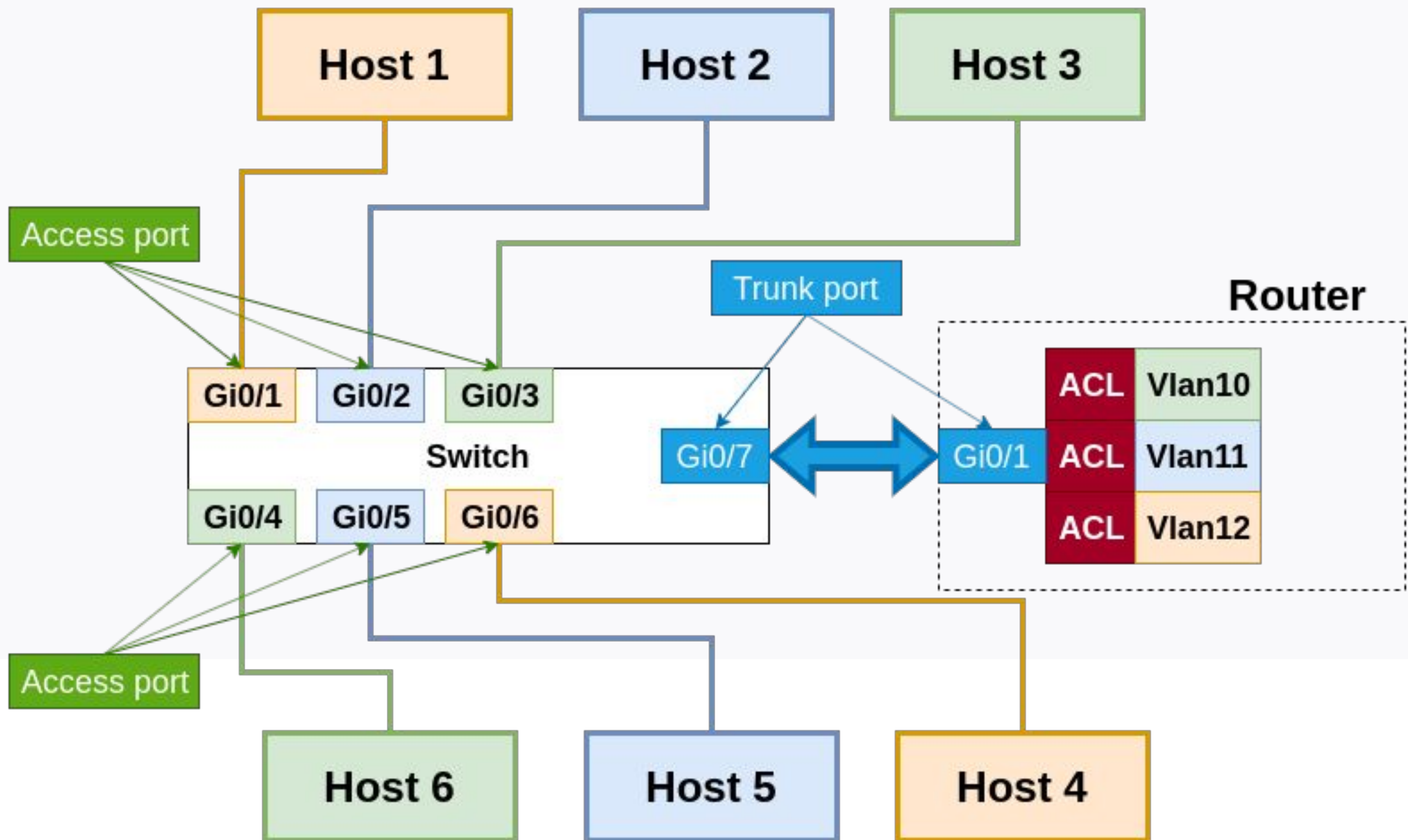
VLAN: режимы портов



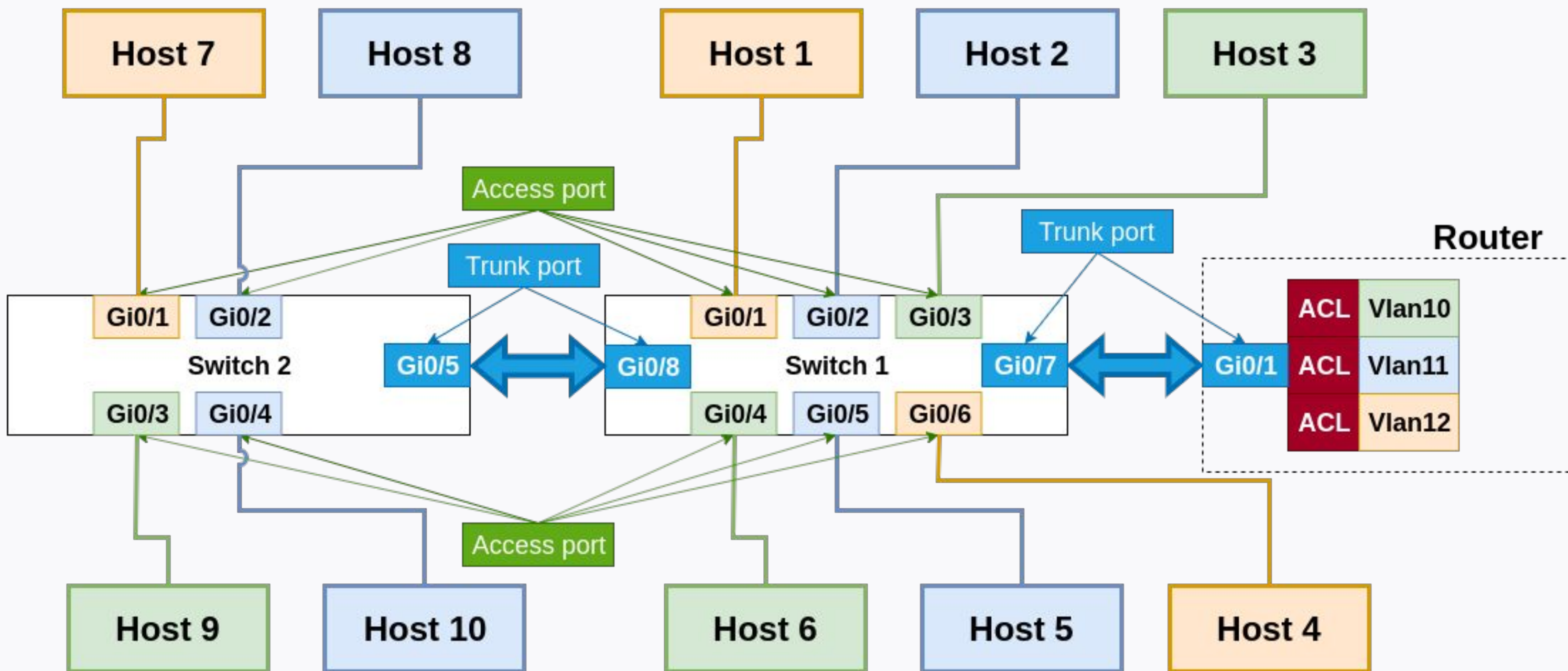
VLAN: режимы портов



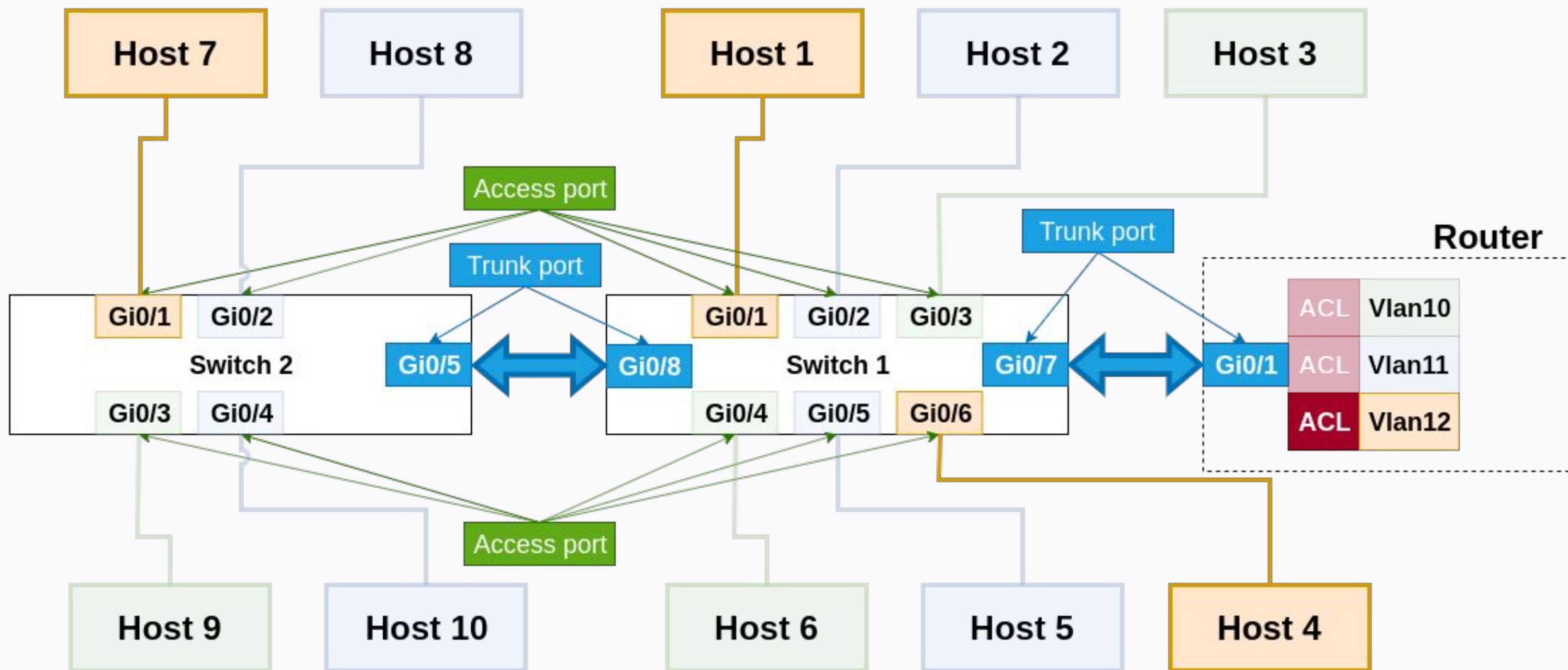
VLAN: режимы портов



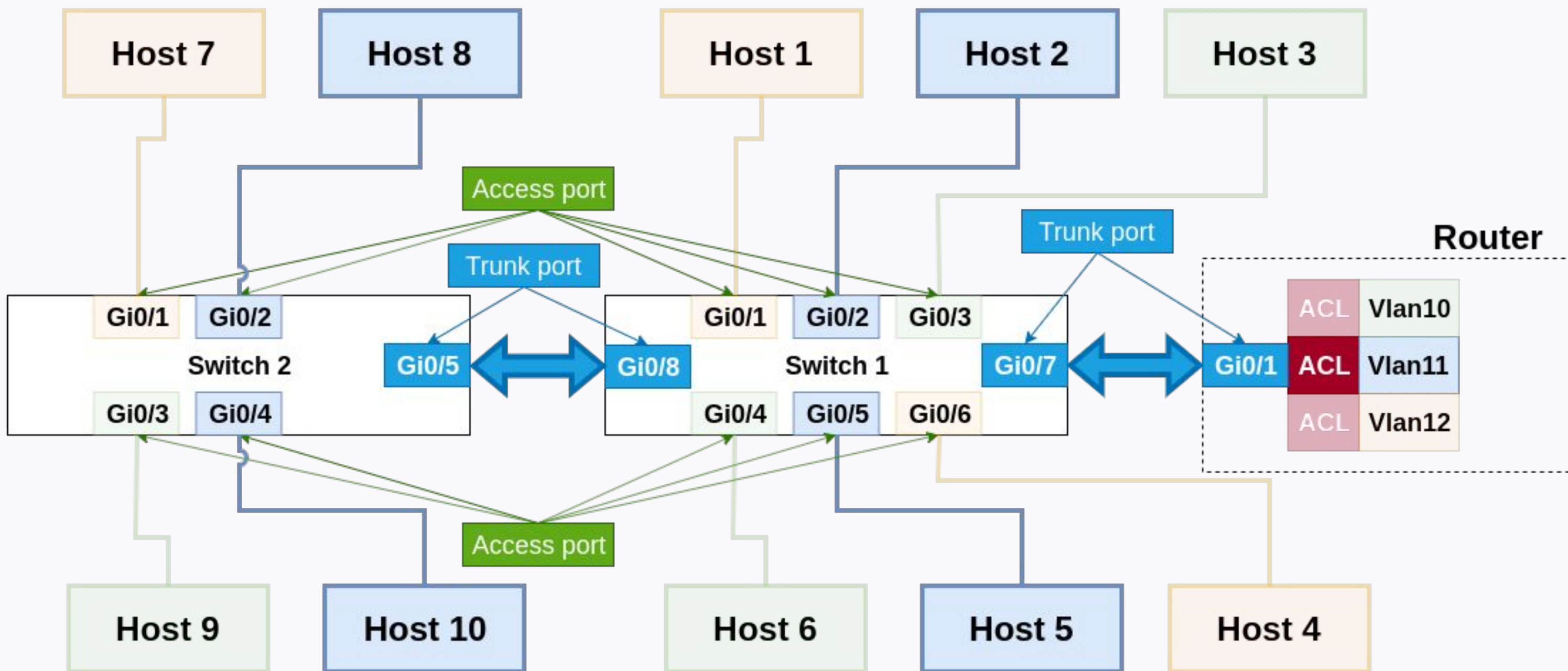
VLAN: режимы портов



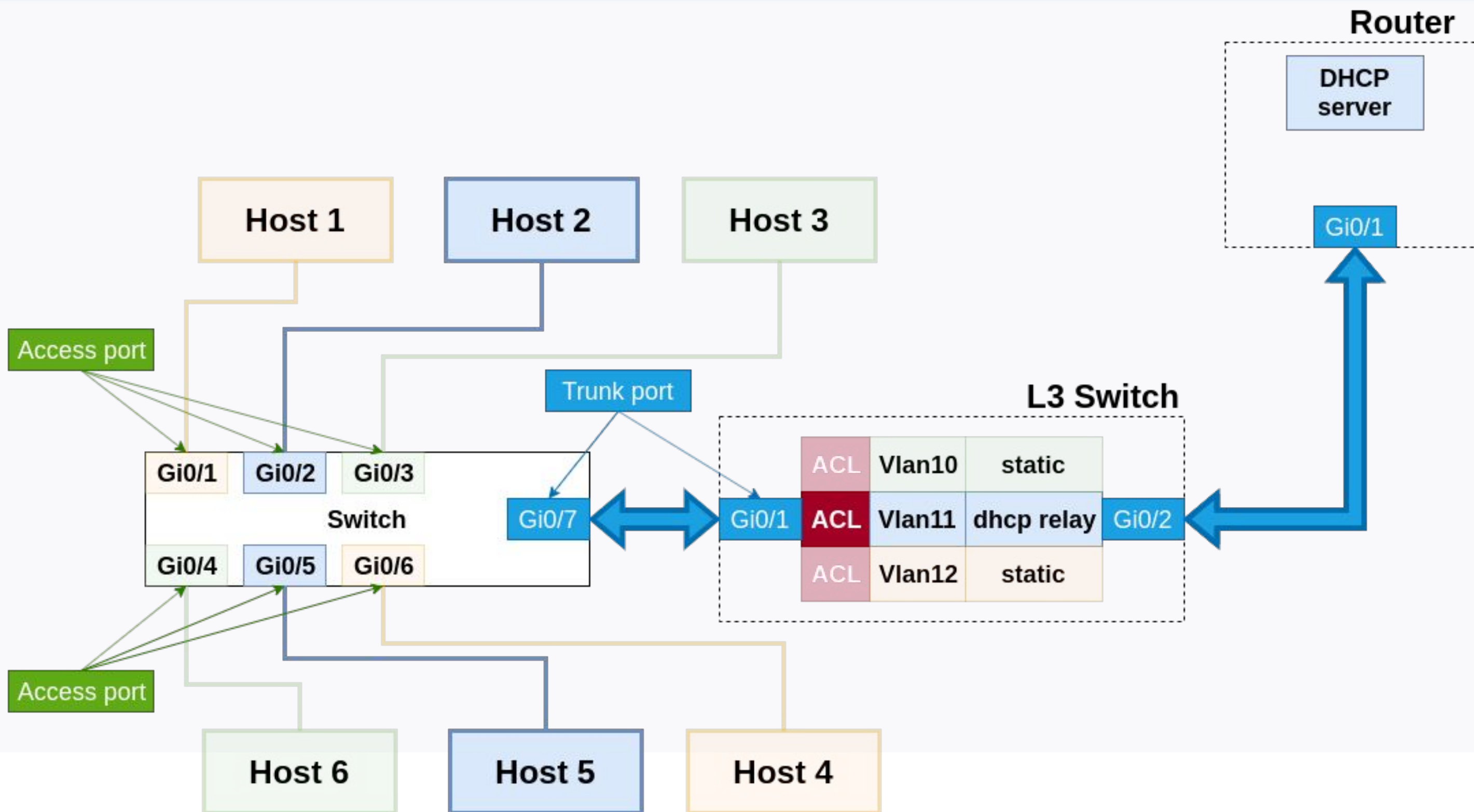
VLAN: режимы портов



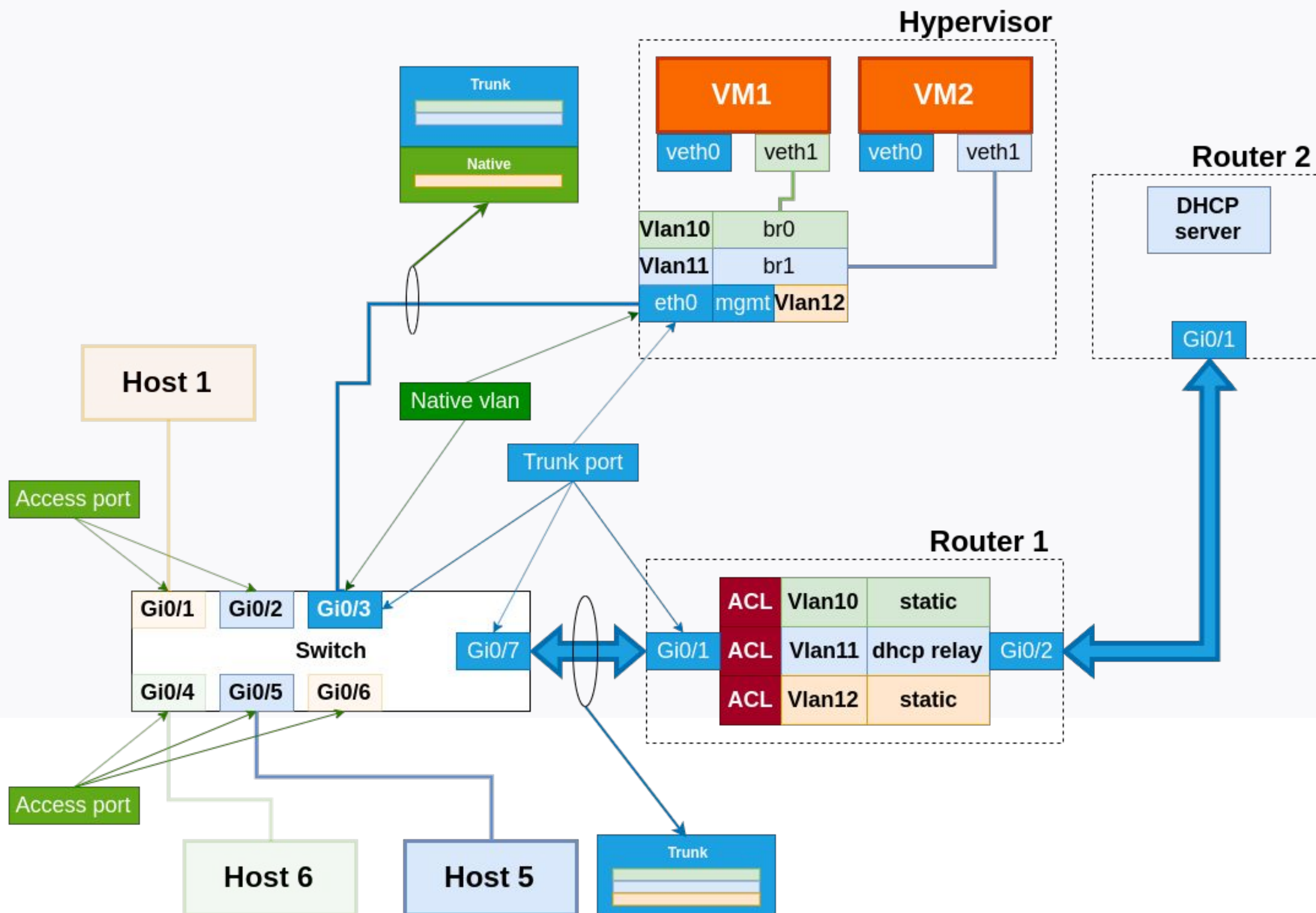
VLAN: режимы портов



VLAN: режимы портов



VLAN: режимы портов





VLAN: настройка

VLAN: настройка

Настройка с помощью vconfig:

```
# Устанавливаем пакет vconfig
yum install vconfig
# Добавляем VLAN 5 на интерфейс eth0
vconfig add eth0 5
# Добавляем адрес на vlan-интерфейс
ip add add 10.10.30.1/30 dev eth0.5
# Поднимаем интерфейс
ifconfig eth0.5 up
```


VLAN: настройка

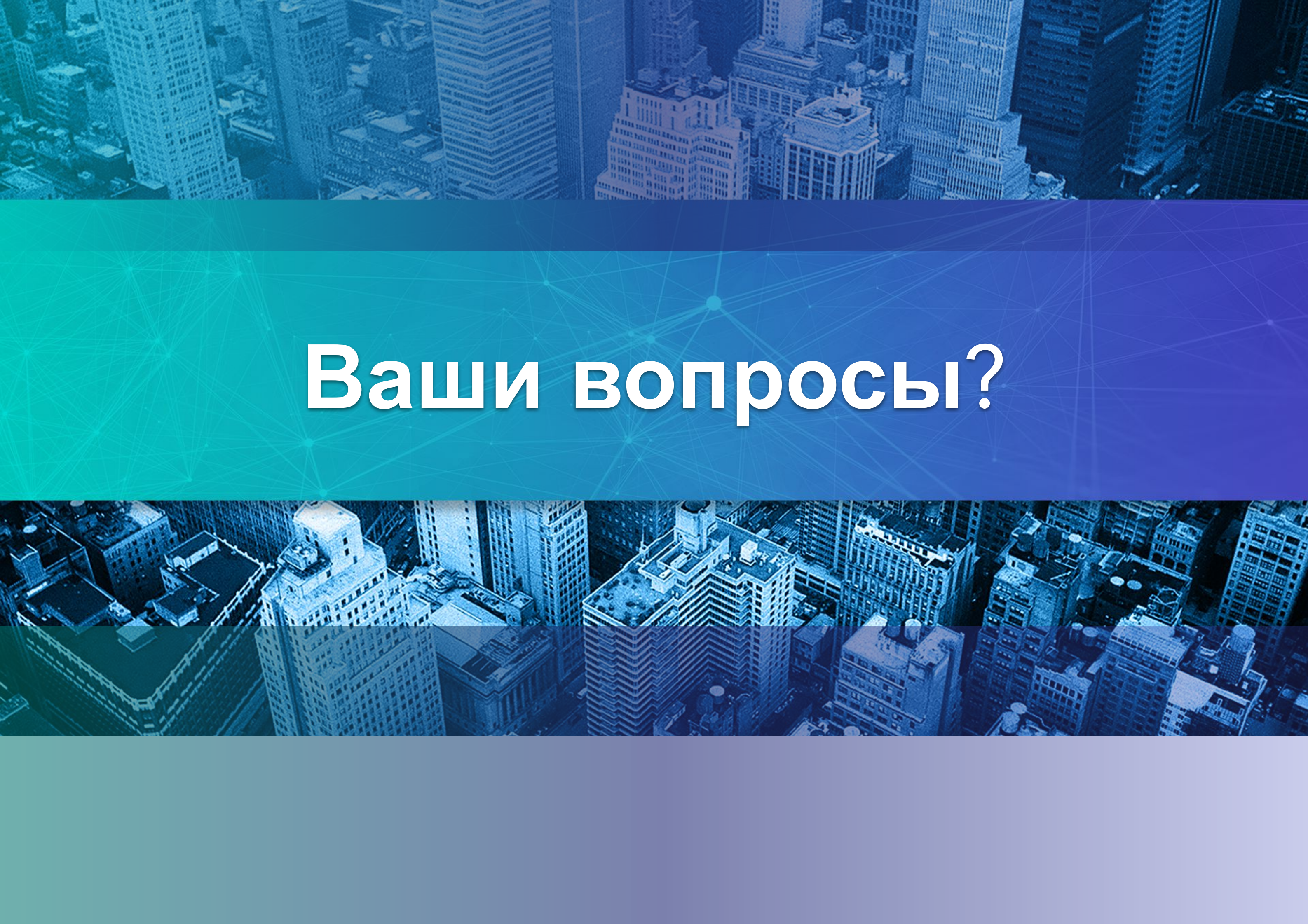
Настройка с помощью конфигов network-scripts:

```
# Конфиг /etc/sysconfig/network-scripts/ifcfg-vlan10
ONBOOT=yes
TYPE=Ethernet
VLAN=yes
VLAN_NAME_TYPE=DEV_PLUS_VID_NO_PAD
DEVICE=vlan10
PHYSDEV=eth0
VLAN_ID=10
BOOTPROTO=static
IPADDR=192.168.0.15
NETMASK=255.255.255.0
NM_CONTROLLED=no
```


VLAN: настройка

Настройка с помощью nmcli:

```
# Добавляем vlan-интерфейс с именем eth0.12 и VID 12
nmcli con add type vlan con-name eth0.12 dev eth0 id 12
# То же самое, но добавляется ip адрес на vlan-интерфейс
nmcli con add type vlan con-name eth0.12 dev eth0 id 12 ip4
192.168.100.1/24
# Смотрим существующие интерфейсы
nmcli connection
nmcli device
```

Ваши вопросы?



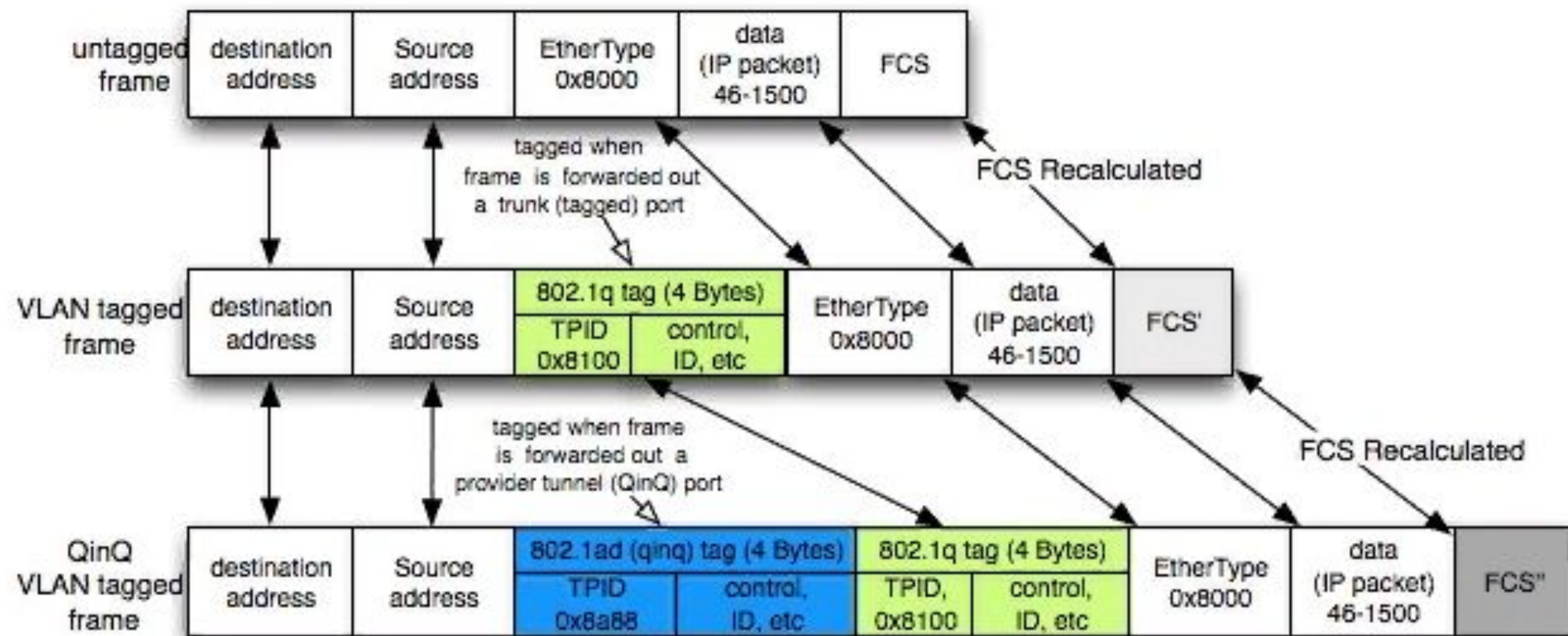
VLAN: QinQ

VLAN: QinQ

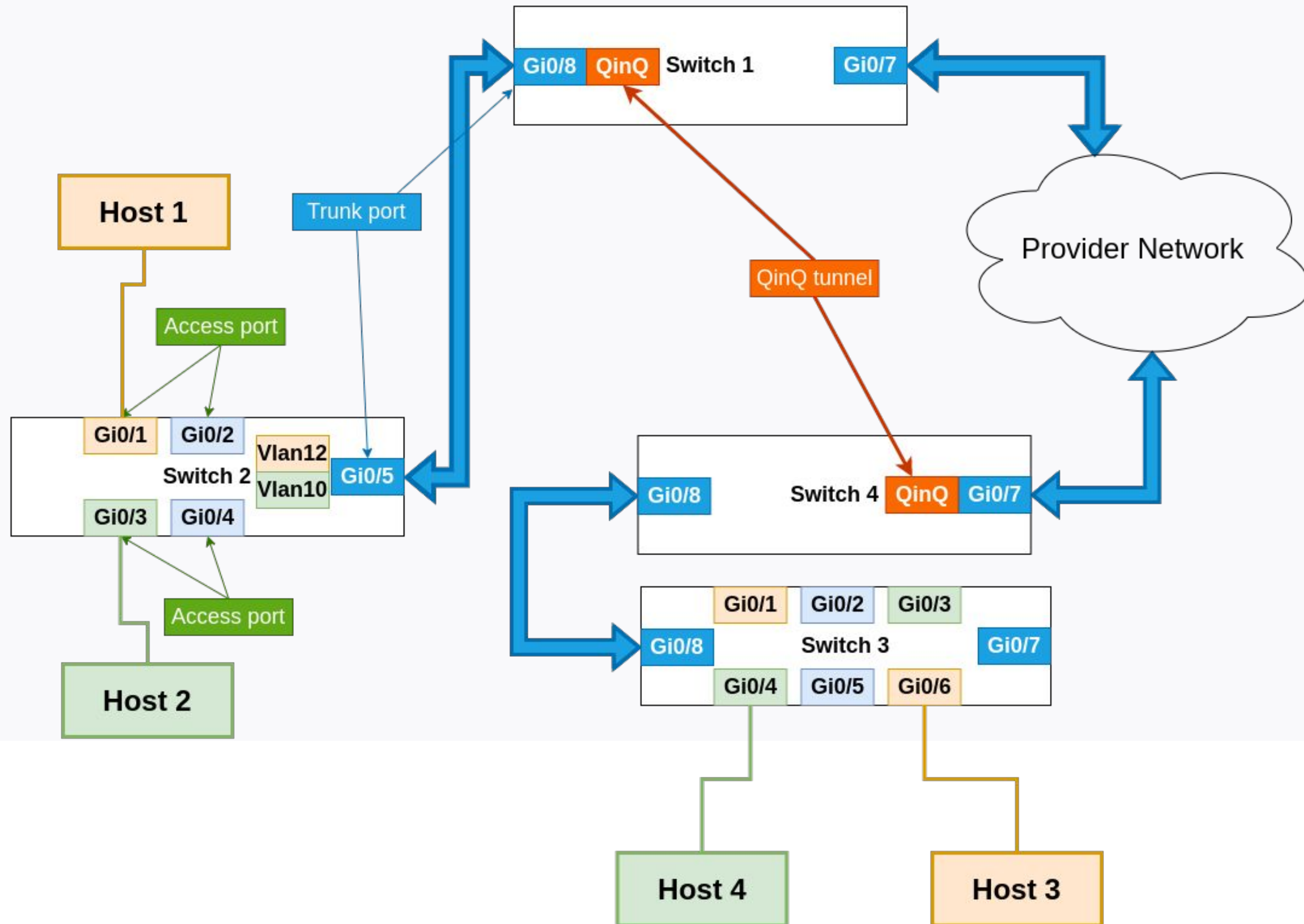
QinQ (IEEE 802.1QinQ) — расширение к стандарту IEEE 802.1Q, описывающее как тегированный трафик может передаваться внутри уже тегированного по 802.1Q трафика.

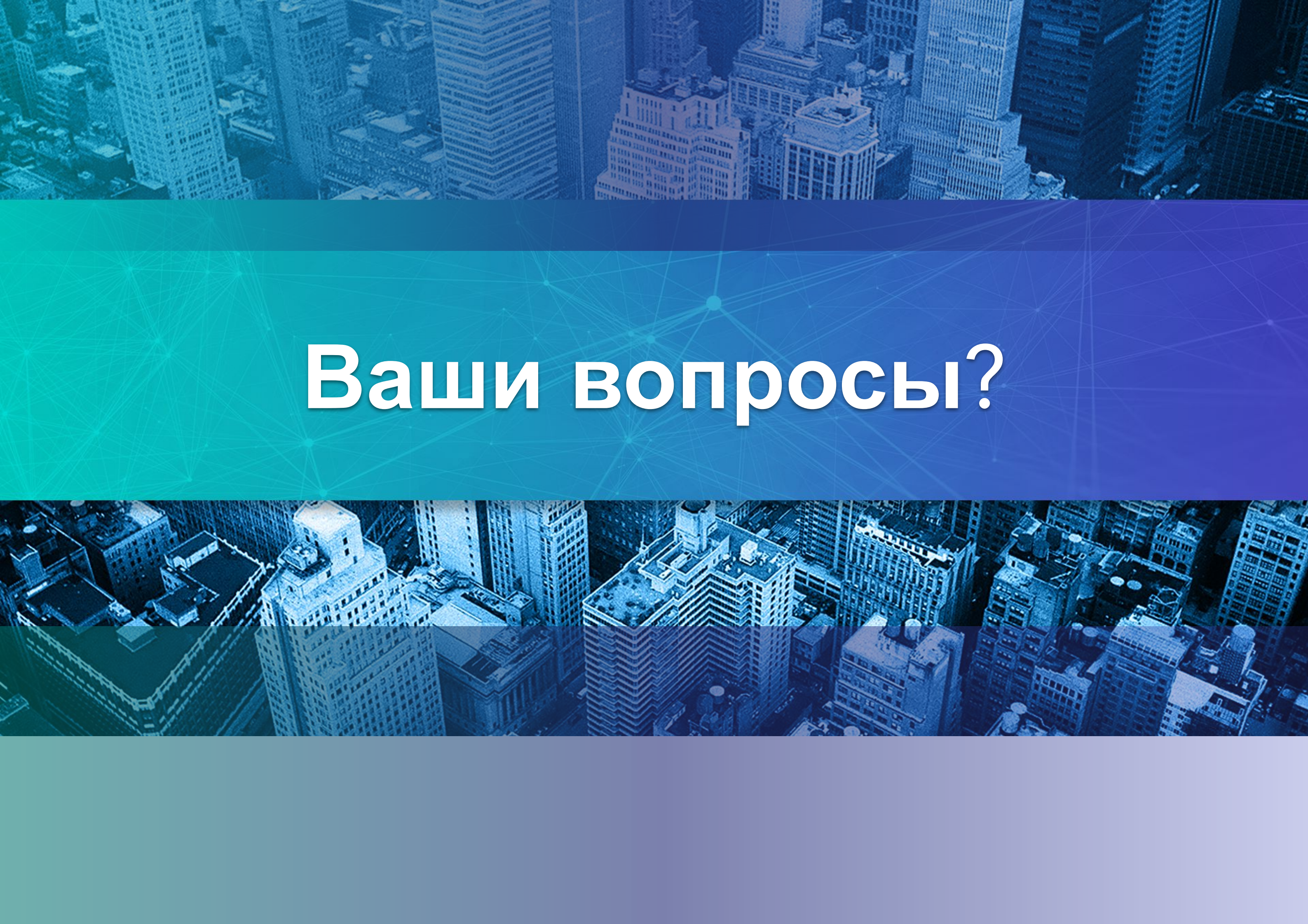
- эта технология имеет большое значение для построения Metro Ethernet-сетей
- для использования QinQ-инкапсуляции требуется поддержка со стороны коммутатора

VLAN: QinQ



VLAN: QinQ



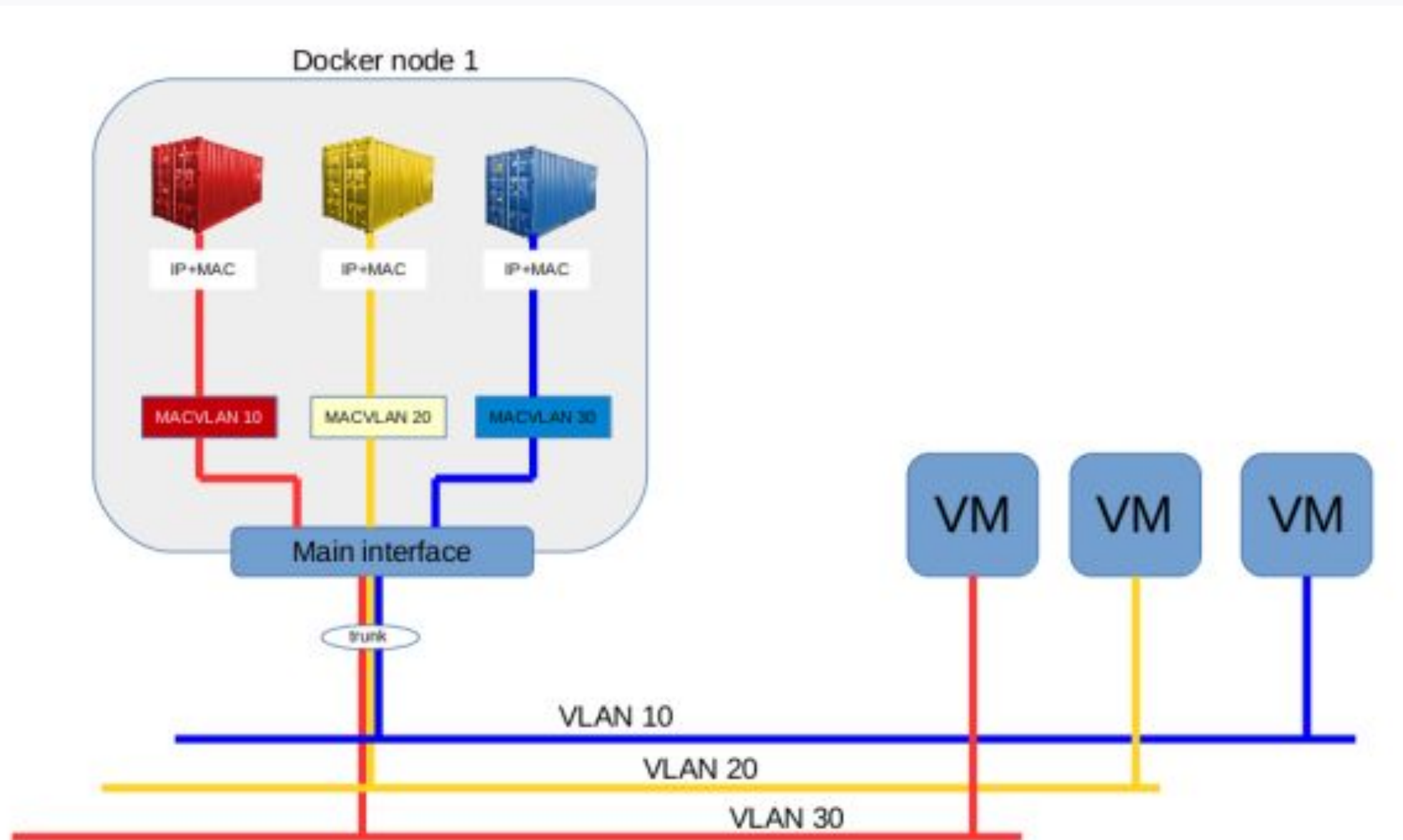


Ваши вопросы?

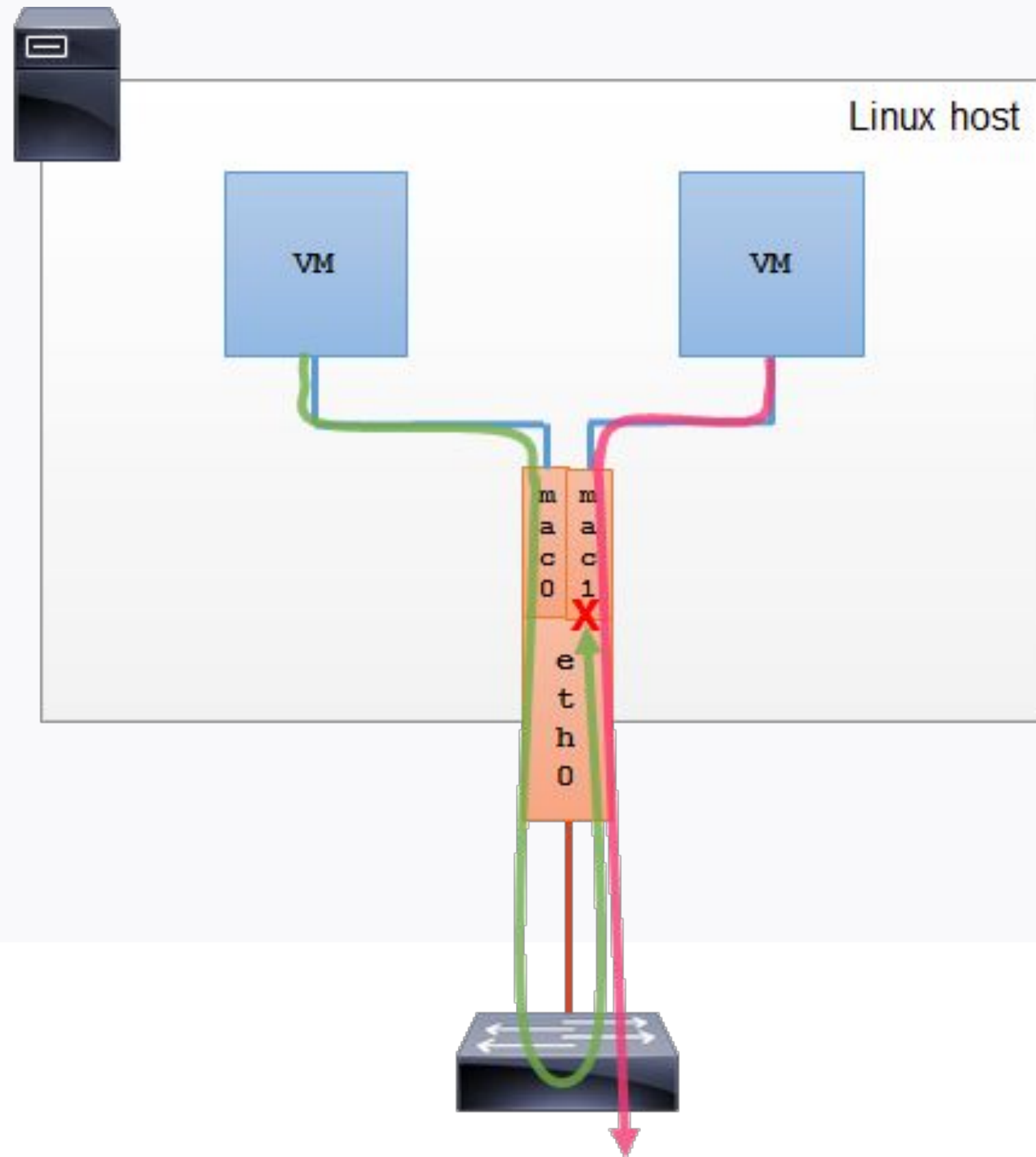
The image features a high-angle, aerial view of a dense urban skyline, likely New York City, with numerous skyscrapers and buildings. The entire image is overlaid with a semi-transparent blue and teal gradient. A network of white lines and dots, resembling a digital or data network, is superimposed over the cityscape. The word "MACVLAN" is prominently displayed in the center in a large, white, sans-serif font.

MACVLAN

MACVLAN



MACVLAN



Маршрут вебинара

Отладка сетевого стека в Linux




VLAN



LACP

The background of the slide is a high-angle, aerial photograph of a dense urban skyline, likely New York City, featuring numerous skyscrapers. The image is tinted with a blue and teal color palette. A semi-transparent network of white lines and dots is overlaid on the image, particularly concentrated in the upper half. The text 'LACP' is centered in the middle of the slide.

LACP



**Вопрос к аудитории:
Что такое LASP и как
применяется?**

Агрегирование каналов (англ. link aggregation) — технологии объединения нескольких параллельных каналов передачи данных в сетях Ethernet в один логический, позволяющие увеличить пропускную способность и повысить надёжность. В различных конкретных реализациях агрегирования используются альтернативные наименования: **транкинг портов** (англ. port trunking), **связывание каналов** (link bundling), **склейка адаптеров** (NIC bonding), **сопряжение адаптеров** (NIC teaming)

https://ru.wikipedia.org/wiki/Агрегирование_каналов

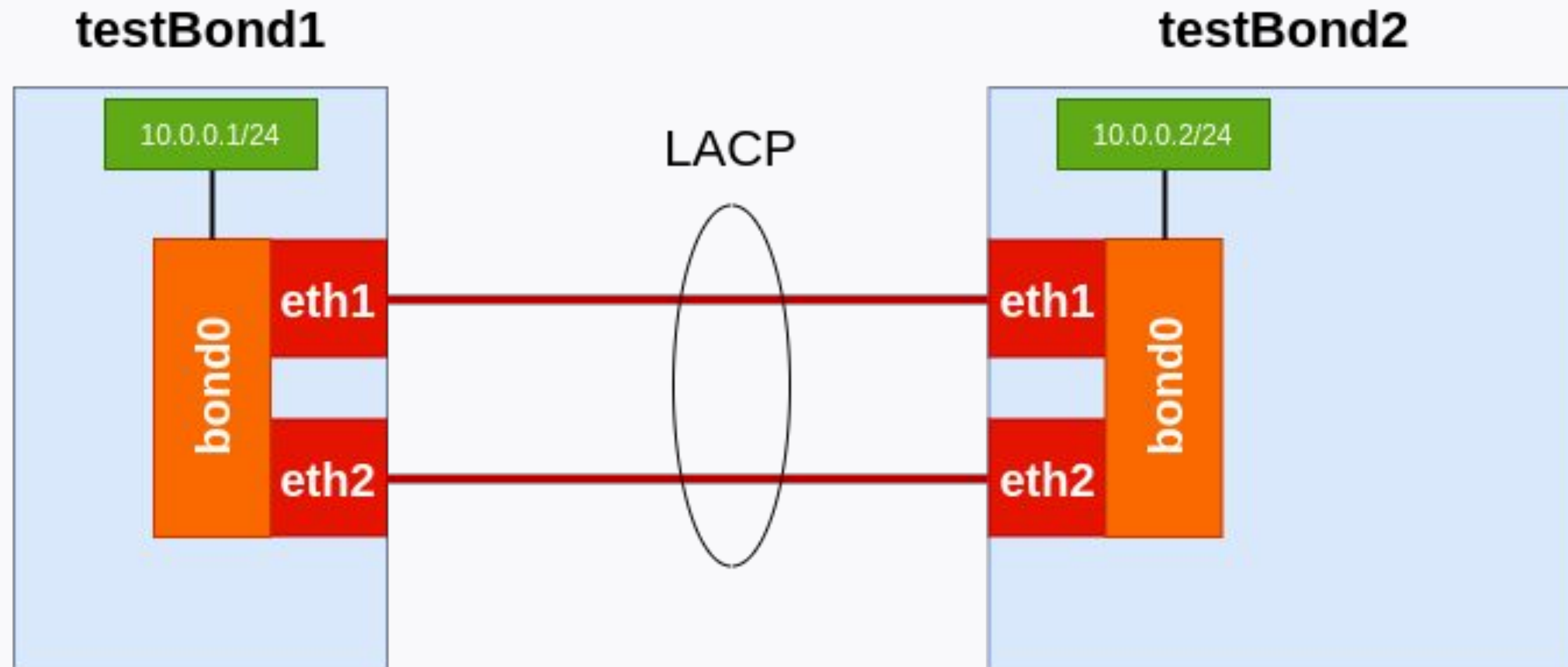
LACP (англ. **link aggregation control protocol**) — открытый стандартный протокол агрегирования каналов, описанный в документах **IEEE 802.3ad** и **IEEE 802.1aq**. Многие производители для своих продуктов используют не стандарт, а патентованные или закрытые технологии, например, Cisco применяет технологию **EtherChannel** (разработанную в начале 1990-х годов компанией Kalpana), а также нестандартный протокол PAgP

https://ru.wikipedia.org/wiki/Агрегирование_каналов

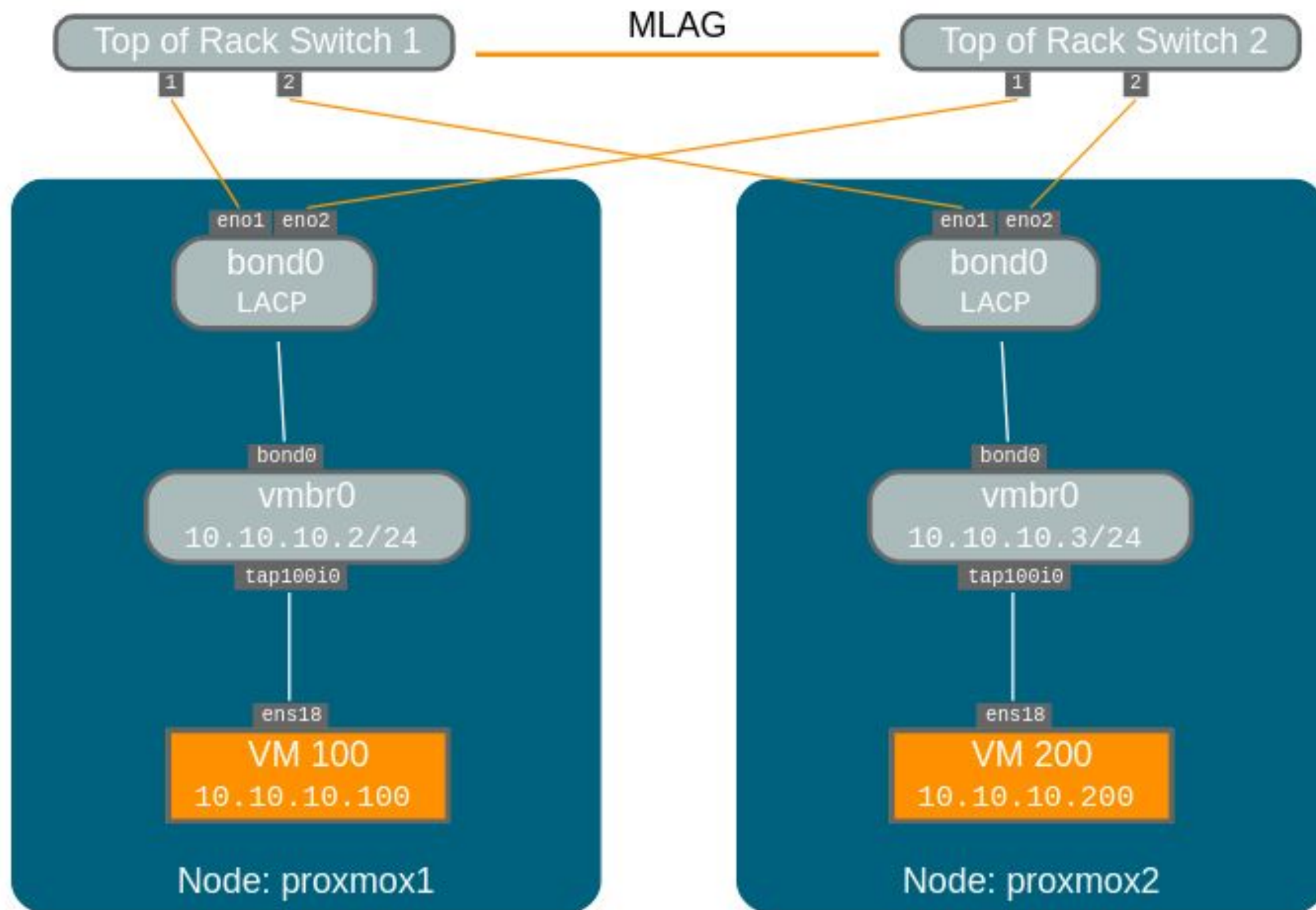


Bonding

Bonding



Bonding



Bonding

Bonding - метод агрегации каналов в Linux

- работает на уровне ядра
- позволяет объединить 2 и более сетевых интерфейса в один логический интерфейс
- позволяет обеспечить отказоустойчивость канала
- позволяет обеспечить распределение нагрузки и балансировку
- позволяет увеличить пропускную способность

Bonding

Режимы работы:

Режим	Тип	Fail Tolerance	Balancing
0	Round Robin	-	+
1	Active Backup	+	-
2	XOR [exclusive OR]	+	+
3	Broadcast	+	-
4	Dynamic Link Aggregation	+	+
5	Transmit Load Balancing (TLB)	+	+
6	Adaptive Load Balancing (ALB)	+	+

Bonding

Настройка с помощью nmcli:

Просмотр сетевых интерфейсов

```
nmcli con
```

Задаем интерфейс bond0, задаем режим и ip-адрес

```
nmcli con add type bond con-name bond0 ifname bond0 mode  
active-backup ip4 10.16.10.7/24
```

Добавляем сетевые интерфейсы в логический интерфейс

```
nmcli con add type bond-slave ifname eth0 master bond0
```

```
nmcli con add type bond-slave ifname eth1 master bond0
```

Последовательно поднимаем интерфейсы

```
nmcli con up bond-slave-eth0
```

```
nmcli con up bond-slave-eth1
```

```
nmcli connection up bond0
```


Bonding

Настройка с помощью конфигов:

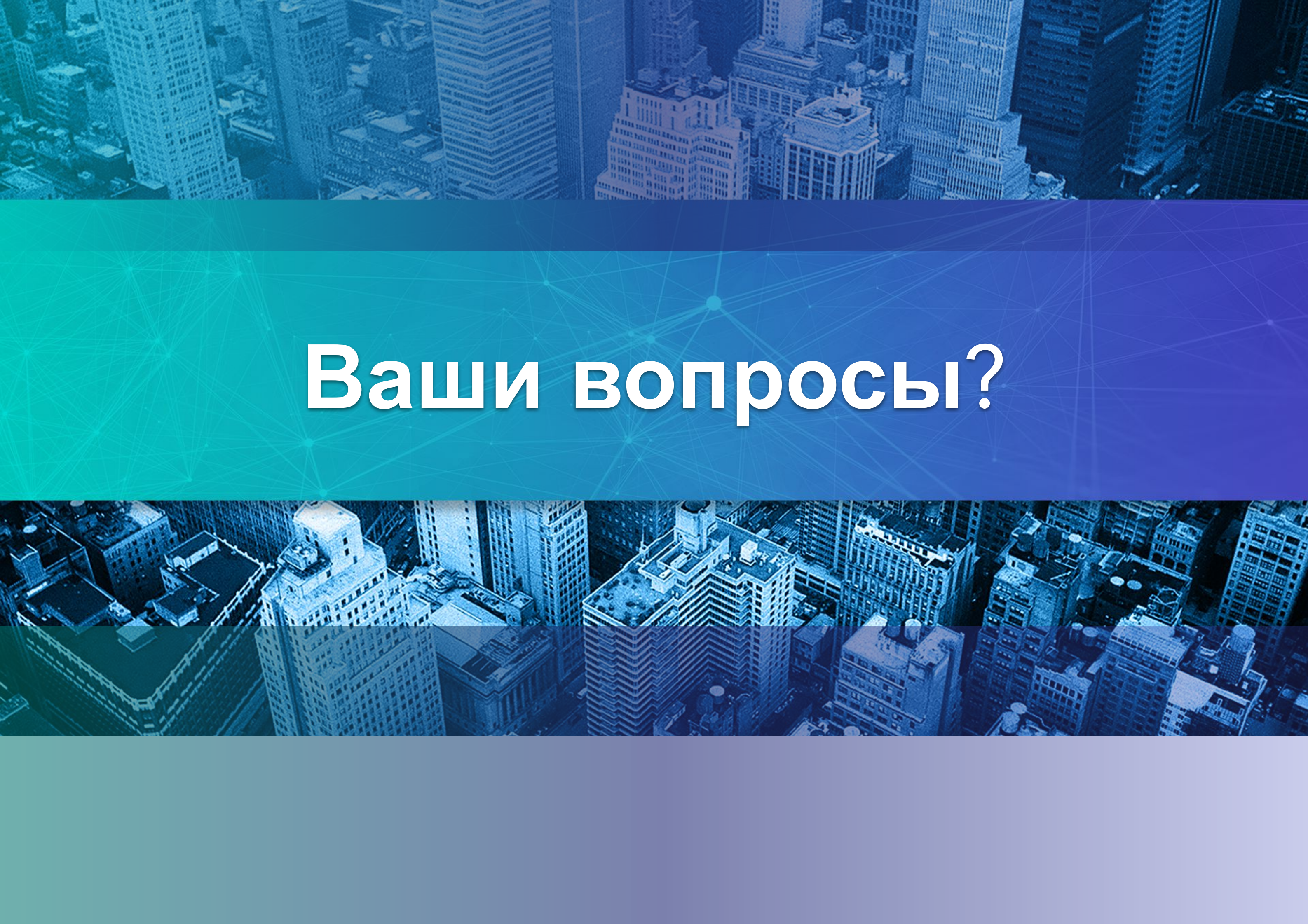
```
cat /etc/sysconfig/network-scripts/ifcfg-bond0
DEVICE=bond0
NAME=bond0
TYPE=Bond
BONDING_MASTER=yes
IPADDR=10.0.0.1
NETMASK=255.255.255.0
ONBOOT=yes
BOOTPROTO=static
BONDING_OPTS="mode=1 miimon=100 fail_over_mac=1"
NM_CONTROLLED=no
USERCTL=no
```


Bonding

Настройка с помощью конфигов:

```
cat /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE=eth1
ONBOOT=yes
BOOTPROTO=none
MASTER=bond0
SLAVE=yes
NM_CONTROLLED=no
USERCTL=no
```

```
cat /etc/sysconfig/network-scripts/ifcfg-eth2
DEVICE=eth2
ONBOOT=yes
BOOTPROTO=none
MASTER=bond0
SLAVE=yes
NM_CONTROLLED=no
USERCTL=no
```

Ваши вопросы?

The background of the slide is a high-angle, aerial photograph of a dense urban skyline, likely New York City, featuring numerous skyscrapers and buildings. The image is heavily color-graded with shades of blue and teal. A semi-transparent horizontal band with a network-like pattern of white dots and lines is overlaid across the middle of the image. The word "Teaming" is centered within this band in a white, sans-serif font.

Teaming

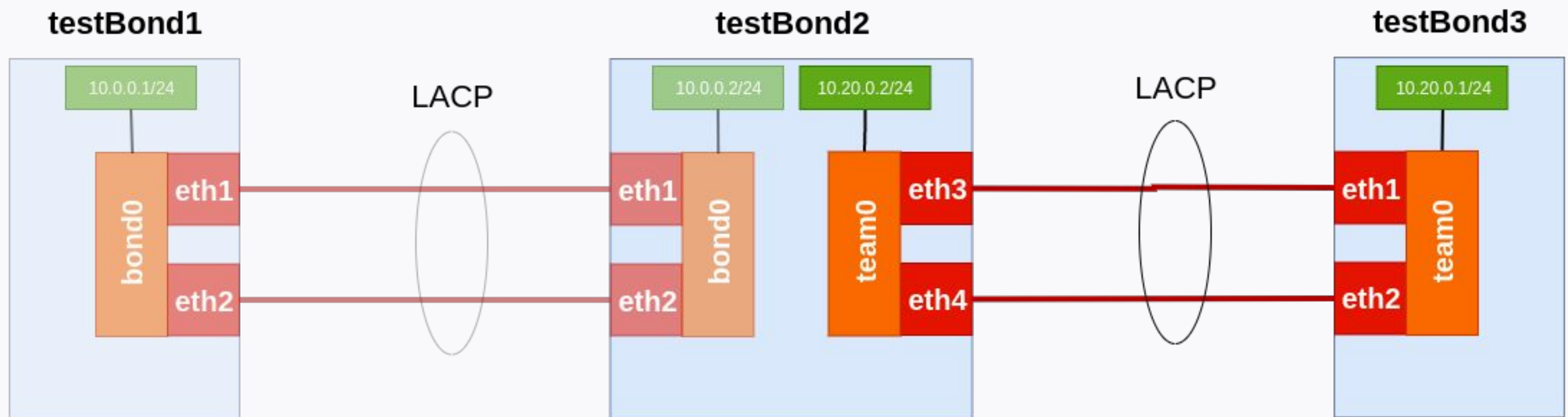
Teaming

Teaming - метод агрегации каналов в Linux

- работает на уровне ядра
- позволяет объединить 2 и более сетевых интерфейса в один логический интерфейс
- позволяет обеспечить отказоустойчивость канала
- позволяет обеспечить распределение нагрузки и балансировку
- позволяет увеличить пропускную способность

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/networking_guide/sec-comparison_of_network_teaming_to_bonding - таблица сравнения bonding`а и teaming`а

Teaming



Teaming

Настройка с помощью nmcli:

Просмотр сетевых интерфейсов

```
nmcli con
```

Задаем интерфейс bond0, задаем режим и ip-адрес

```
nmcli con add type team con-name team0 ifname team0 mode  
active-backup ip4 10.16.10.7/24
```

Добавляем сетевые интерфейсы в логический интерфейс

```
nmcli con add type team-slave ifname eth0 master team0
```

```
nmcli con add type team-slave ifname eth1 master team0
```

Последовательно поднимаем интерфейсы

```
nmcli con up team-slave-eth0
```

```
nmcli con up team-slave-eth1
```

```
nmcli connection up team0
```


Teaming

Настройка с помощью конфигов:

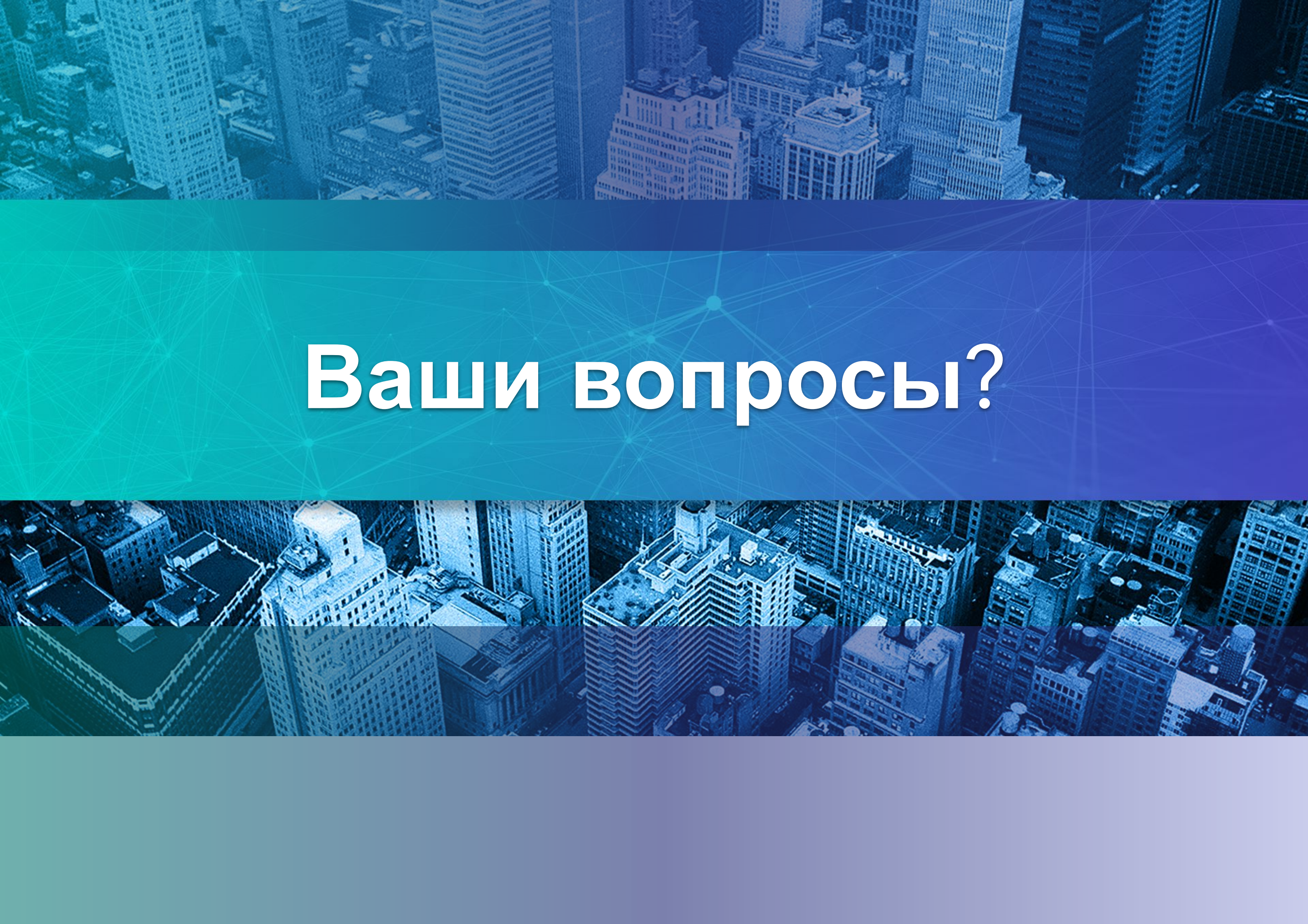
```
cat /etc/sysconfig/network-scripts/ifcfg-team0
DEVICE=team0
IPADDR=10.20.0.2
NETMASK=255.255.255.0
ONBOOT=yes
NM_CONTROLLED=no
USERCTL=no
BOOTPROTO=none
DEVICETYPE="Team"
TEAM_CONFIG='{ "runner" : { "name" : "activebackup", "hwaddr_policy" : "by_active" }, "link_watch" : { "name" : "ethtool" } }'
```


Teaming

Настройка с помощью конфигов:

```
cat /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE=eth1
ONBOOT=yes
BOOTPROTO=none
NM_CONTROLLED=no
USERCTL=no
DEVICETYPE="TeamPort"
TEAM_MASTER="team0"
TEAM_PORT_CONFIG='{ "prio" : -100 }
```

```
cat /etc/sysconfig/network-scripts/ifcfg-eth2
DEVICE=eth2
ONBOOT=yes
BOOTPROTO=none
NM_CONTROLLED=no
USERCTL=no
DEVICETYPE="TeamPort"
TEAM_MASTER="team0"
TEAM_PORT_CONFIG='{ "prio" : -100 }
```


The background of the slide features an aerial view of a dense city skyline, likely New York City, with numerous skyscrapers. The image is tinted with a blue and teal color scheme. A network of white lines and dots is overlaid on the image, creating a digital or technological feel. The text "Ваши вопросы?" is centered in the middle of the slide in a large, white, sans-serif font.

Ваши вопросы?

Домашнее задание

- 1 В Office1 в тестовой подсети появляется сервер с доп. интерфейсами и адресами в internal сети testLAN
- 2 Изолировать с помощью vlan:
testClient1 <-> testServer1
testClient2 <-> testServer2
- 3 Между centralRouter и inetRouter создать 2 линка и объединить их с помощью bond-интерфейса, проверить работу с отключением сетевых интерфейсов
- 4 Результат ДЗ: vagrant файл с требуемой конфигурацией
Конфигурация должна разворачиваться с помощью ansible

Рефлексия



Назовите 3 момента, которые вам запомнились в процессе занятия



Что вы будете применять в работе из сегодняшнего вебинара?



Заполните, пожалуйста,
опрос о занятии по ссылке в чате



Спасибо за внимание!
Приходите на следующие вебинары



Викирюк Павел

Системный инженер