

# Курс «Администратор Linux»

## VRRP и HAproxy

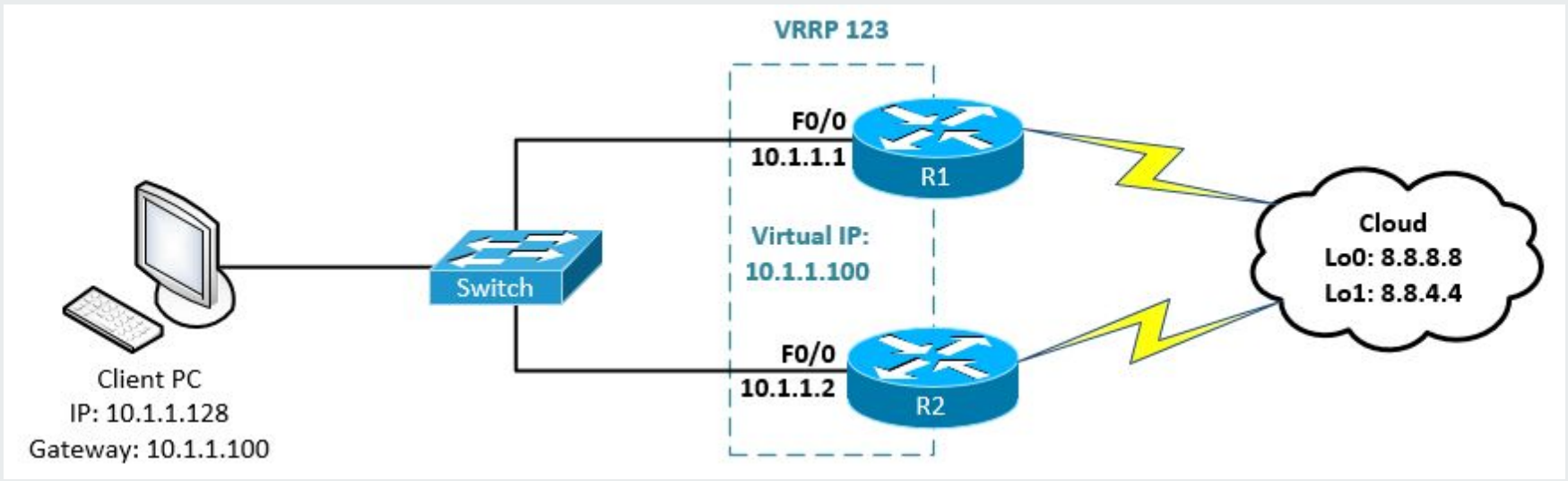
Занятие # 33

Алексей Цыкунов



- **VRRP**
- **HAproxy**

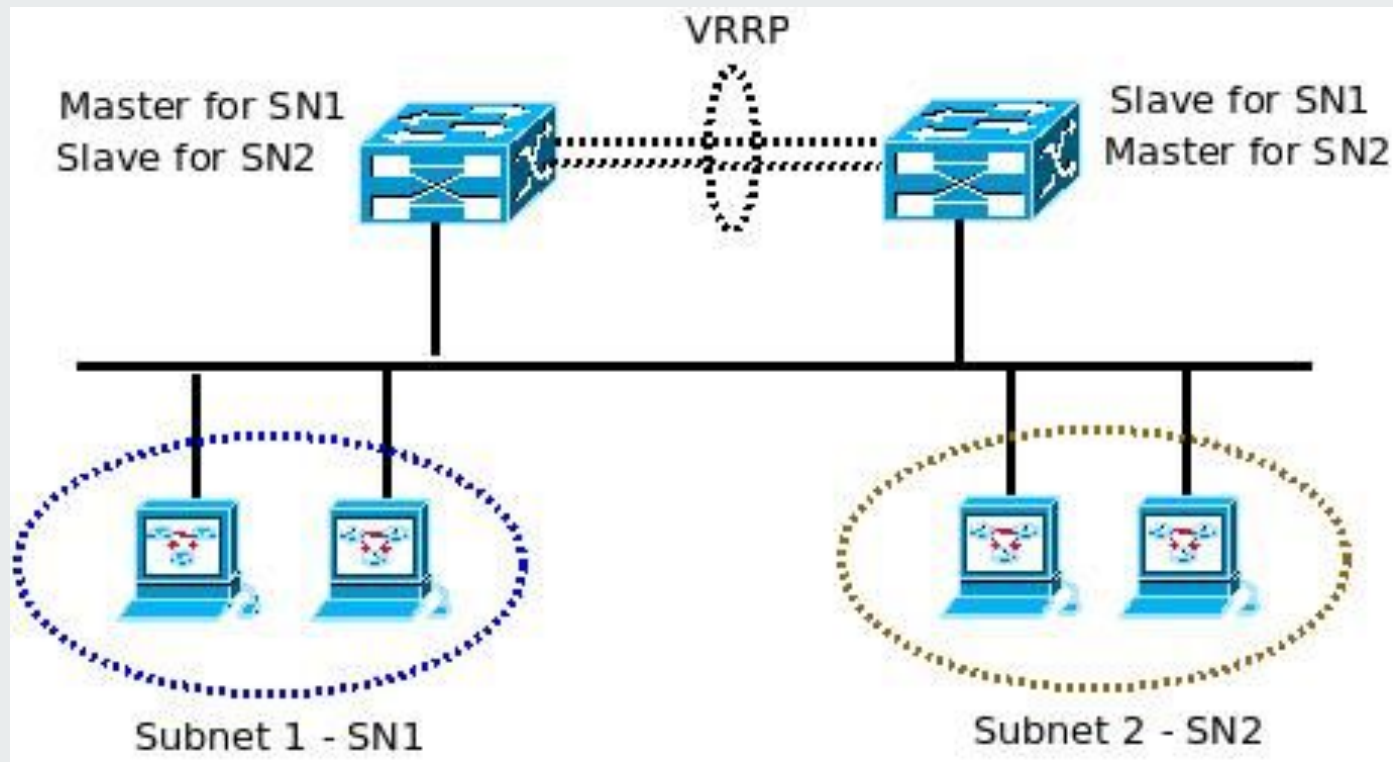
- **VRRP** (Virtual Router Redundancy Protocol) — сетевой протокол, предназначенный для увеличения доступности маршрутизаторов выполняющих роль шлюза по умолчанию. Это достигается путём объединения группы маршрутизаторов в один **виртуальный маршрутизатор** и назначения им **общего IP-адреса**, который и будет использоваться как шлюз по умолчанию для компьютеров в сети.



- **VRRP-маршрутизатор (VRRP Router)** – маршрутизатор, на котором работает протокол VRRP. Он может участвовать в одном или более виртуальных маршрутизаторах.
- **Виртуальный маршрутизатор (Virtual Router, VR)** – группа маршрутизаторов в одной сети с одним **VIP** и **VRID**
- **Владелец IP-адреса (IP Address Owner)** – VRRP-маршрутизатор, который использует IP-адрес, назначенный виртуальному маршрутизатору, как реальный IP-адрес, присвоенный интерфейсу.
- **VRRP-объявление (ADVERTISEMENT)** – сообщения, которые отправляет Master-маршрутизатор.

- **Виртуальный IP-адрес (VIP)** — это IP-адрес, присвоенный интерфейсу одного из маршрутизаторов, которые составляют Virtual Router.
- **Virtual Router Master или VRRP Master router** — VRRP-маршрутизатор, который отвечает за отправку пакетов, отправленных на IP-адрес, который ассоциирован с виртуальным маршрутизатором, и за ответы на ARP-запросы, отправленные на этот адрес. Если владелец IP-адреса доступен, то он всегда становится Master.
- **Virtual Router Backup или VRRP Backup router** — один или несколько маршрутизаторов, готовых взять на себя роль VRRP Master router, если текущий VRRP Master router станет недоступным.
- **Виртуальный MAC-адрес (Virtual MAC)** — **0000:5E00:01xx**, где xx — номер группы VRRP.

- FailOver маршрутизатора
- Быстрое переключение
- Возможность режиме Active/Active
- IETF Open Standard Protocol - поддержка многими вендорами
- Возможность нескольких устройств как бэкапных

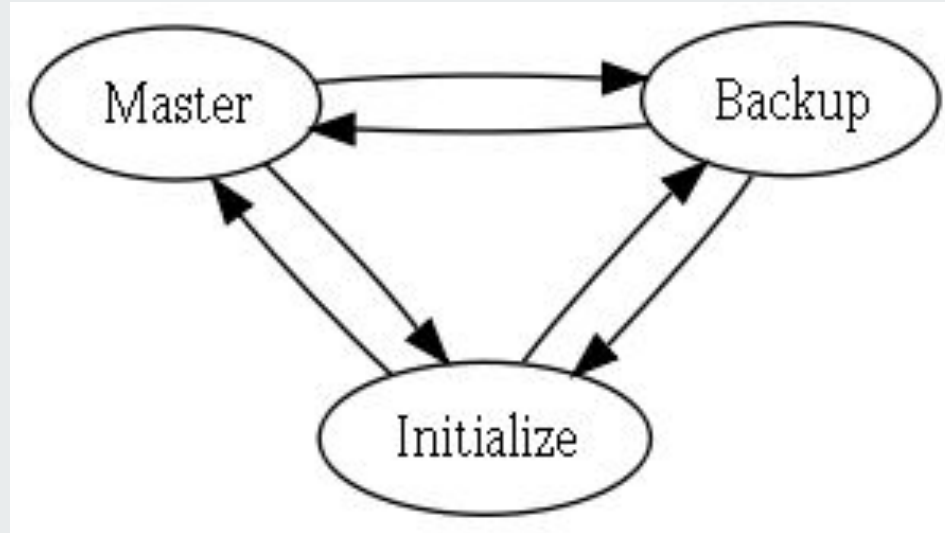


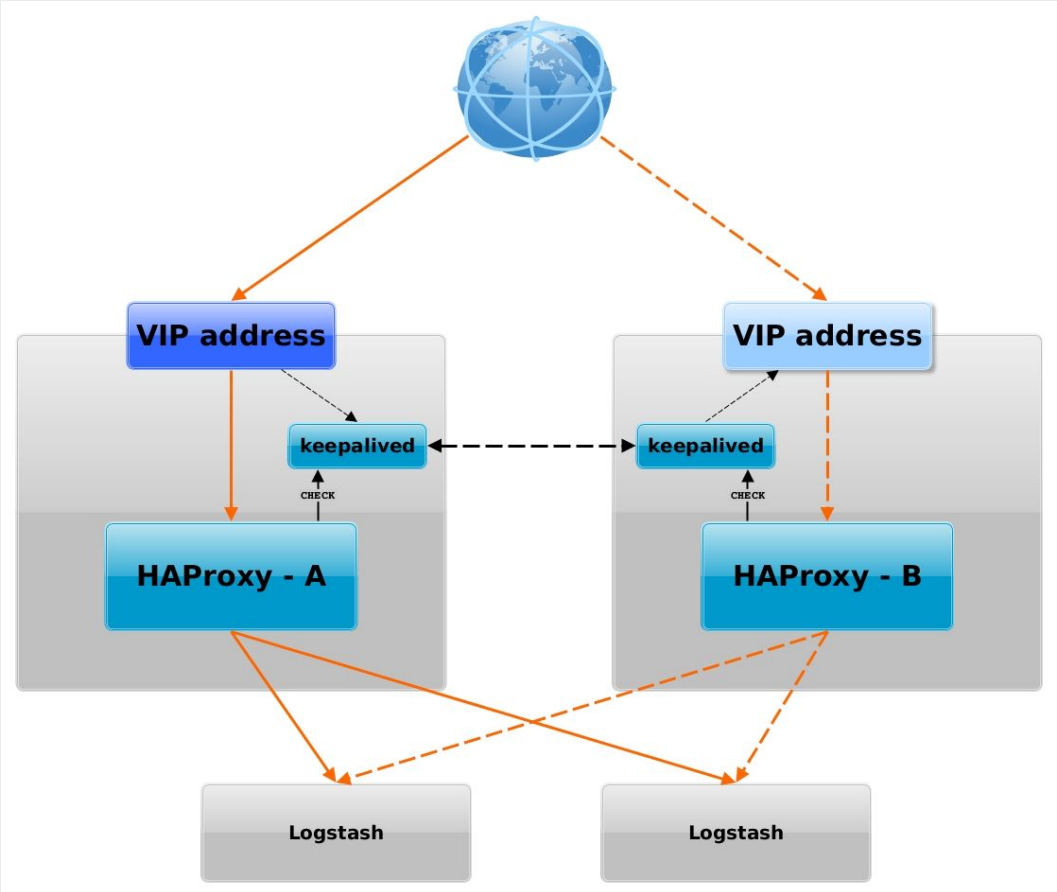
- Роутер с максимальным приоритетом становится **master router**.
- При включение роутера с более высоким приоритетом - он перехватит роль (можно отключить)
- Все остальные - **backup routers**.
- Виртуальный MAC - **0000.5e00.01xx**, где **xx** - номер группы в 16-м формате
- Hello пакеты шлются каждую секунду на мультикаст адрес 224.0.0.18.
- VRRP-маршрутизатор может находиться в нескольких виртуальных маршрутизаторах, каждый с уникальной комбинацией **VRID/IP-адрес**. Соответствия между VRID и IP-адресом должны быть одинаковыми на всех маршрутизаторах в одной сети.

- **VRRP object tracking** - обеспечивает метод проверки что **master-роутер** является наилучшим из всех VRRP-роутеров группы путем изменения приоритетов VRRP и опроса статуса tracked objects, таких как интерфейс или состояние маршрутизации IP.
- VRRP использует специальный Object Tracking - независимый процесс, создающий, опрашивающий и удаляющий специальные объекты, отражающие состояние роутера.

- **Идентификатор виртуального маршрутизатора (VRID)** — настраиваемое значение в диапазоне от 1 до 255. Нет значения по умолчанию
- **Приоритет** — значение приоритета (1 - 254)
  - a. 255 - владелец адреса
  - b. 0 - Мастер
- **IP-адрес** — один или более IP-адресов, присвоенные виртуальному маршрутизатору. Нет значения по умолчанию
- **Advertisement\_Interval** — временной интервал между отправкой объявлений. По умолчанию 1 секунда

- **Skew Time** – время (в секундах), которое используется для отклонения Master Down Interval.
  - а. Рассчитывается по формуле:  $(256 - \text{Priority}) / 256$
- **Master Down Interval** – временной интервал, после которого Backup Router станет Master Router.
  - а. Рассчитывается по формуле:  $(3 * \text{Advertisement Interval}) + \text{Skew time}$
- **Режим preempt** – контролирует то, будет ли Backup-маршрутизатор с более высоким приоритетом пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.
- **Тип аутентификации**
- **Данные аутентификации**





- `net.ipv4.ip_nonlocal_bind = 1`
- `yum install keepalived -y`

- Однопоточный неблокирующий event-driven демон
- Open-source
- TCP прокси/балансировщик с проверкой состояний
- Http/Https reverse-proxy
- SSL terminator/initiator/offloader
- L4/L7 балансировщик
- Контентный коммутатор
- Защита от Ddos и точка диагностирования проблем в сети

- **Global** определяет общую конфигурацию (logging, user, group...)
- **Defaults** определяет настройки по-умолчанию (mode, maxconn...)
- **Listen** объединяет в себе описание для фронтенда и бэкенда и содержит полный список прокси.
- **Frontend** определяет, каким образом перенаправлять запросы к бэкенду в зависимости от того, что за запрос поступил от клиента.
- **Backend** содержит список серверов и отвечает за балансировку нагрузки между ними в зависимости от выбранного алгоритма

**ACL** - свойство HAproxy для принятия решения о направлении трафика в зависимости от контента в реквесте

```
acl <имя правила> <критерий> [flags] [operator] [<value>] ...
```

```
frontend http-in
  bind *:80
  acl url_appX path_beg -i /appX/
  use_backend appX-backend if url_appX
  default_backend appZ-backend
```

Выбор бэкенда происходит на основе:

- Веса weight
- Алгорима Round-Robin
- Алгоритма выбора наименее загруженного сервера
- Алгоритма определения по хэшу IP клиента

Закрепления сервера за клиентом с помощью:

- Куки

```
cookie SRVNAME insert
```

```
server lamp1 10.0.0.1:80 cookie S1 check
```

```
server lamp2 10.0.0.2:80 cookie S2 check
```

- Префикса куки

## global

```
log 127.0.0.1 local0 notice  
user haproxy  
group haproxy
```

## defaults

```
log global  
retries 2  
timeout connect 3000  
timeout server 5000  
timeout client 5000
```

```
listen mysql-cluster
  bind 10.140.0.4:3306
  mode tcp
  option mysql-check user haproxy_check
  balance roundrobin
  server mysql-1 10.140.0.2:3306 check
  server mysql-2 10.140.0.3:3306 check
```

# Спасибо за внимание

Алексей Цыкунов

