?????? ?
??????????????????

- Подключение агента KUMA через HA Proxy
- VIP адрес для использования с балансировками (отказоустойчивость)
- Балансировка UDP/TCP трафика (L3-L4) средствами службы Nginx

?????????? ?????? KUMA ????? HA Proxy

Схема работы агента KUMA через промежуточный узел с использованием haproxy

Для обращения агента KUMA к серверу Core используются порты 7210/tcp и 8429/tcp

Правила МЭ для работы

№п/п	Наименование источника	ip-адрес источника	Наименование получателя	ip-адрес получателя	Порты	Примечание
1	Агент KUMA	10.10.15.15	Хост с HA Proxy	10.10.15.15	7210/tcp 8429/tcp	
2	Хост с HA Proxy	10.10.15.15	Ядро KUMA	192.168.110.5	7210/tcp 8429/tcp	

Для работы необходимо:

1. В файле hosts на хосте, где планируется установить агент kuma прописать fqdn-имя core kuma и ip-адрес с установленным haproxy (какое-то другое имя не подойдет, так как между компонентами kuma аутентификация происходит по сертификатам)
2. На промежуточном хосте с коллекторами установить haproxy
3. Установить агент kuma на машину, как рекомендует вендор, с указанием fqdn имени core kuma

Для нашего примера

1. В файл hosts прописываем `10.10.15.15  core.kuma.example`
2. В конфигурации с haproxy в файле `/etc/haproxy/haproxy.cfg`

Добавить следующий конфиг

```
frontend core.kuma.example:7210
        bind            *:7210
        mode            tcp
        log             global
        default_backend core_kuma_api

frontend core.kuma.example:8429
        bind            *:8429
        mode            tcp
        log             global
        default_backend core_kuma_agent

backend core_kuma_api
        mode                    tcp
        log                         global
        balance                 roundrobin
        option          tcp-check
        server          core   192.168.110.5:7210

backend core_kuma_agent
        mode                    tcp
        log                         global
        balance                 roundrobin
        option          tcp-check
        server          core   192.168.110.5:8429
```

3. Установить агента kuma

```
kuma.exe agent --core https://core.kuma.example:7210 --id 334b673f-b9a0-4dde-8398-45a7648ef767 --user имя_машины\имя_пользователя --install
```

VIP ????? ??? ????????????? ? ?????????????? (??????????????????)

> Информация, приведенная на данной странице, является разработкой community KUMA и **НЕ** является официальной рекомендацией вендора.

Чтобы настроить балансировку трафика между коллекторами KUMA:

1. Установите nginx на сервере, предназначенном для управления потоком событий (предпочтительно выделенные сервера, не менее двух)

- Команда для установки в Oracle Linux 8+:
  `$sudo dnf install keepalived`
- Команда для установки в Ubuntu 20.4:
  `$sudo apt-get install keepalived`

2. Подготавливаем конфигурационный файл **/etc/keepalived/keepalived.conf** под свою задачу. **Обратите внимание, конфига два! Нужно раскидать конфиг по серверам ACTIVE\BACKUP**

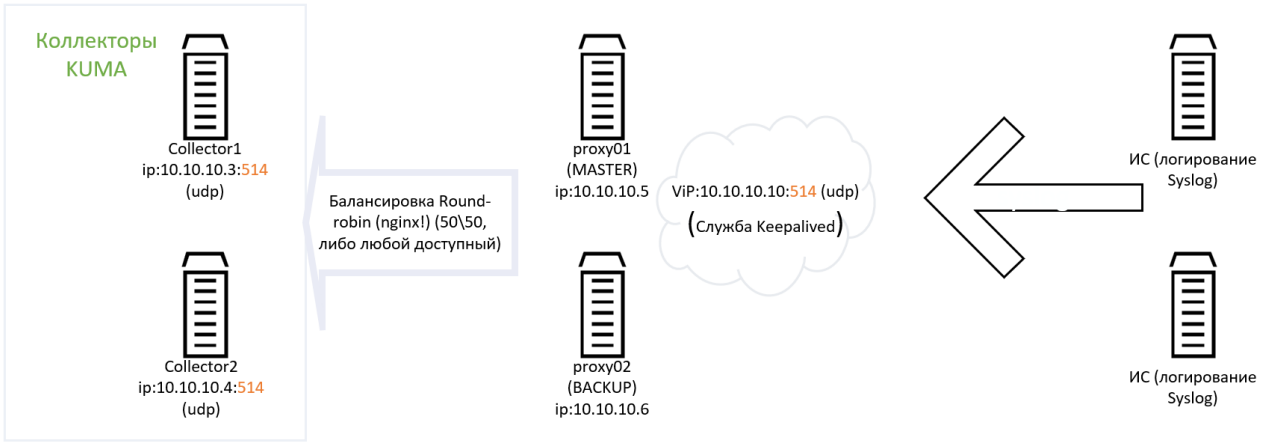
```
#CONFIG FOR MASTER SERVER

! Barebones conf File for keepalived

  global_defs {
     notification_email {
       your_mail@testmailcompany.ru
     }
     notification_email_from keepalived@testmailcompany.ru
     smtp_server mail.testmailcompany.ru
     smtp_connect_timeout 60
  }

  vrrp_instance VI_1 {
      state MASTER
```

```
      interface ens192 #меняем под свой интерфейс
      virtual_router_id 100
      priority 100
      advert_int 1
      authentication {
          auth_type PASS
          auth_pass 12345678 #меняем пароль!
      }
      virtual_ipaddress {
          10.10.10.10
      }
  }



#CONFIG FOR BACKUP SERVER

! Barebones conf File for keepalived

  global_defs {
     notification_email {
       your_mail@testmailcompany.ru
     }
     notification_email_from keepalived@testmailcompany.ru
     smtp_server mail.testmailcompany.ru
     smtp_connect_timeout 60
  }

  vrrp_instance VI_1 {
      state BACKUP
      interface ens192 #меняем под свой интерфейс
      virtual_router_id 100
      priority 100
      advert_int 1
      authentication {
          auth_type PASS
          auth_pass 12345678 #меняем пароль!
      }
      virtual_ipaddress {
          10.10.10.10
```

```
    }
}
```

3. Запускаем службу командой `sudo systemctl start keepalived` на двух серверах, при выводе ip -a можем наблюдать на MASTER сервере - дополнительный адрес - 10.10.10.10, для проверки "переезда" адреса можем остановить службу на MASTER сервере командой `sudo systemctl stop keepalived`, виртуальный адрес поднимется на BACKUP сервере.

4. Настраиваем по статье балансировку средствами nginx и получается следующая отказоустойчивая схема приёма логов на коллекторах:



?????????? UDP/TCP ??????? (L3-L4) ?????????? ?????? Nginx

> Информация, приведенная на данной странице, является разработкой команды pre-sales и/или community KUMA и **НЕ** является официальной рекомендацией вендора.

> Официальная документация по данному разделу приведена в Онлайн-справке на продукт: Управление потоком событий с помощью nginx (kaspersky.com)

Чтобы настроить балансировку трафика между коллекторами KUMA:

1. Установите nginx на сервере, предназначенном для управления потоком событий (предпочтительно выделенные сервера, не менее двух)

- Команда для установки в Oracle Linux 8.6:
  `$sudo dnf install nginx`
- Команда для установки в Ubuntu 20.4:
  `$sudo apt-get install nginx`
  При установке из sources, необходимо собрать с параметром `-with-stream`:
  `$sudo ./configure -with-stream -without-http_rewrite_module -without-http_gzip_module`

2. Подготавливаем конфигурационный файл nginx.conf, где блоки выделенные красным меняем (название\ip адреса\порт) под свою задачу.

```
{
      upstream back_FW_ASA {
      server 10.11.17.145:514;
      server 10.11.18.145:514;
      }
```

и

```
server {
            listen 514 udp;
            proxy_pass back_FW_ASA;
            proxy_bind $remote_addr transparent;
      }
```

При помощи данного файла nginx будет "прозрачно" для коллекторов пробрасывать оригинальный сетевой пакет трафика, позволяя передать реальный адрес\имя устройства, которое передало лог. При необходимости прослушивания TCP убираем в 16 строке udp.

```
user nginx;
worker_processes auto;
error_log /var/log/nginx/error.log;
pid /run/nginx.pid;
# Load dynamic modules. See /usr/share/doc/nginx/README.dynamic.
include /usr/share/nginx/modules/*.conf;
events {
    worker_connections 1024;
}
stream {
        upstream back_FW_ASA {
        server 10.11.17.145:514;
        server 10.11.18.145:514;
        }
        server {
                listen 514 udp;
                proxy_pass back_FW_ASA;
                proxy_bind $remote_addr transparent;
        }
}
```

3. Укажите адреса коллекторов KUMA и порт, в примере их адреса\порты - 10.11.17.145:514 ? 10.11.18.145:514, ???????????? ???????? ????????? ????? ??????? ? ???????????? 50:50 (??? ???????????? ???? ????????????).

4. Служба балансировщика будет "прослушивать" 514 порт со всех IP адресов сервера, для большей отказоустойчивости служб предлагается использовать службу keepalived на двух серверах. Настройка keepalived