Пользовательские статьи

В данной книге зарегистрированные пользователи могут создавать свои собственные статьи. После прохождения модерации такие статьи могут быть перенесены в существующие разделы базы знаний.

- Балансировка UDP/TCP трафика (L3-L4) средствами службы Nginx
- Массовое обновление KUMA агентов
- Включение IPv6 Oracle\CentOS\RedHat

Балансировка UDP/TCP трафика (L3-L4) средствами службы 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

Массовое обновление KUMA агентов

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

> Данный механизм работает с версии 3.4 KUMA, т.к именно в ней добавилась поддержка ключа --accept-eula

> С сервера, на котором будем запускать скрипт понадобятся сетевые доступы к серверам по 5985/TCP для запуска команд и 445/TCP для передачи файлов, также права локального администратора на сервере.

За основу берем скрипт

И заполняем в нём ключевые поля, а именно:

```
# Список серверов
$servers = @("server1.domain.local", "server2.domain.local")

# Задаем переменную версии
$version = "3.4.1.53"

# Локальный путь к файлу, который нужно скопировать
$localFilePath = "C:\Temp\kuma.exe"

# Путь на удаленном сервере, куда будет скопирован файл (если используется нестандартный путь для установки KUMA агента)
$remoteFilePath = "C$\Program Files\Kaspersky Lab\KUMA\kuma.exe"
```

Принцип работы скрипта:

1. Подключается к серверам из списка (поочередно)
2. Проверяем, есть ли служба KUMA на сервере
3. Остановка службы KUMA

4. Копируем обновленный файл kuma.exe в директорию KUMA
5. Правим файл с лицензионным соглашением (обновляем значение версии в файле .license-version
6. Добавляем в реестре, в службе KUMA ключ --accept-eula
7. Запускаем службу KUMA агента

Включение IPv6 Oracle\CentOS\RedHat

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

Для того, чтобы включить IPv6 нам понадобится:

1. Проверить наличие трех полей по части IPv6 с правильным атрибутом (no\yes), в файле `/etc/sysconfig/network-scripts/ifcfg-` `ens192` (имя интерфейса может меняться, точный необходимо уточнять командой ip a) как в примере, если их нет - добавляем их (при необходимости правим значения.):

   ```
   IPV6INIT=yes
   IPV6_DEFROUTE=yes
   IPV6_FAILURE_FATAL=no
   ```

2. Проверяем папку `/etc/sysctl.d/*` на наличие файлов, если они есть - ищем в них следующие строки конфигурации. Обычно используются:

   ```
   net.ipv6.conf.lo.disable_ipv6 = 1
   net.ipv6.conf.all.disable_ipv6 = 1
   net.ipv6.conf.default.disable_ipv6 = 1
   ```

   Если файлов **.conf в папке* /etc/sysctl.d/ нет, или отсутствуют строки, касающиеся работы IPv6 - идем в файл /etc/sysctl.conf и ищем в нем данные строки, если нашли - комментируем их (#перед строкой).
3. Вводим команду `sysctl -p`, IPv6 локальный должен появится. Проверяем командой ip -6 addr
4. Если адрес не появился, проверяем файл `/etc/default/grub`
   в строке GRUB_CMDLINE_LINUX добавляем `ipv6.disable=0`
   Должно получиться: `GRUB_CMDLINE_LINUX="ipv6.disable=0"`, также в этой строке могут быть другие параметры, не стираем их, а добавляем. Запятую ставить не нужно.
5. Запускаем команду `sudo grub2-mkconfig` и `sudo init 6`. Сервер перезагрузится.
6. Проверяем командой `ip -6 addr`

Особый случай с RedOS 8:

Все вышеуказанные рекомендации были применены, но grub не обновлялся.
Пришлось править руками файл /boot/grub2/grub.cfg так как оставались записи (выделены желтым)
После правки файла не забываем выполнить grub2-mkconfig