Балансировка нагрузки в Nginx и Angie

Лавлинский Николай

Технический директор «Метод Лаб»

Более 15 лет в веб-разработке

Преподавал в ВУЗе более 10 лет
Более 4 лет в онлайн-образовании

Автор YouTube-каналов “Ускорение сайтов” и “Поддержка сайтов”
https://t.me/methodlab_tg
https://www.youtube.com/c/NickLavlinsky
https://www.youtube.com/@site_support

Специализация: оптимизация производительности, ускорение сайтов и веб-приложений

План

- Nginx
- Angie
- Типы балансировки
- Round robin
- Least connections
- IP hash / Hash
- Random
- Параметры директивы `server`
- Серверное кэширование

Ссылка на презентацию

Nginx

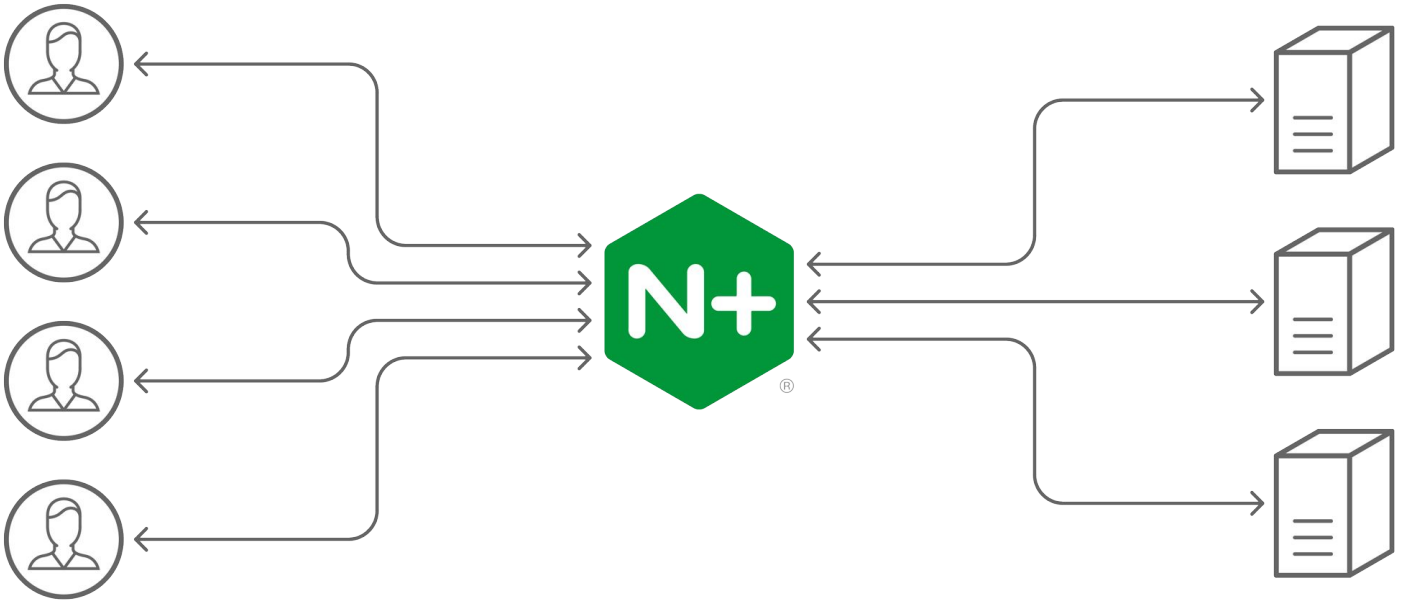
- Один из самых популярных в мире
- Разработан Игорем Сысоевым (первые версии)
- Легковесный
- Высокопроизводительный
- Может работать как веб-сервер, прокси, балансировщик, кэш
- `HTTP/1.1, HTTP/2, HTTP/3, TLS 1.3, Gzip, Brotli...`

NGINX®
Part of F5

Angie

- Форк Nginx с отдельной командой
- Вся функциональность Nginx
- Встроенные средства мониторинга
- Консоль с веб-интерфейсом
- `slow_start` при вводе апстрима
- `resolve` для списка апстримов
- Готовые бинарные пакеты в репозитории
- https://angie.software/

Общая схема балансировки



Виды балансировки

- HTTP-балансировка (ngx_http_upstream_module)
    - Round Robin
    - Least Connection
    - IP Hash
    - Hash
    - Random Least Connection
    - Least Time (random) (NGINX Plus)
- TCP и UDP (ngx_stream_upstream_module)
    - Round Robin
    - Least Connection
    - Hash
    - Random Least Connection
    - Least Time (random) (NGINX Plus)

NGINX®
Part of F5

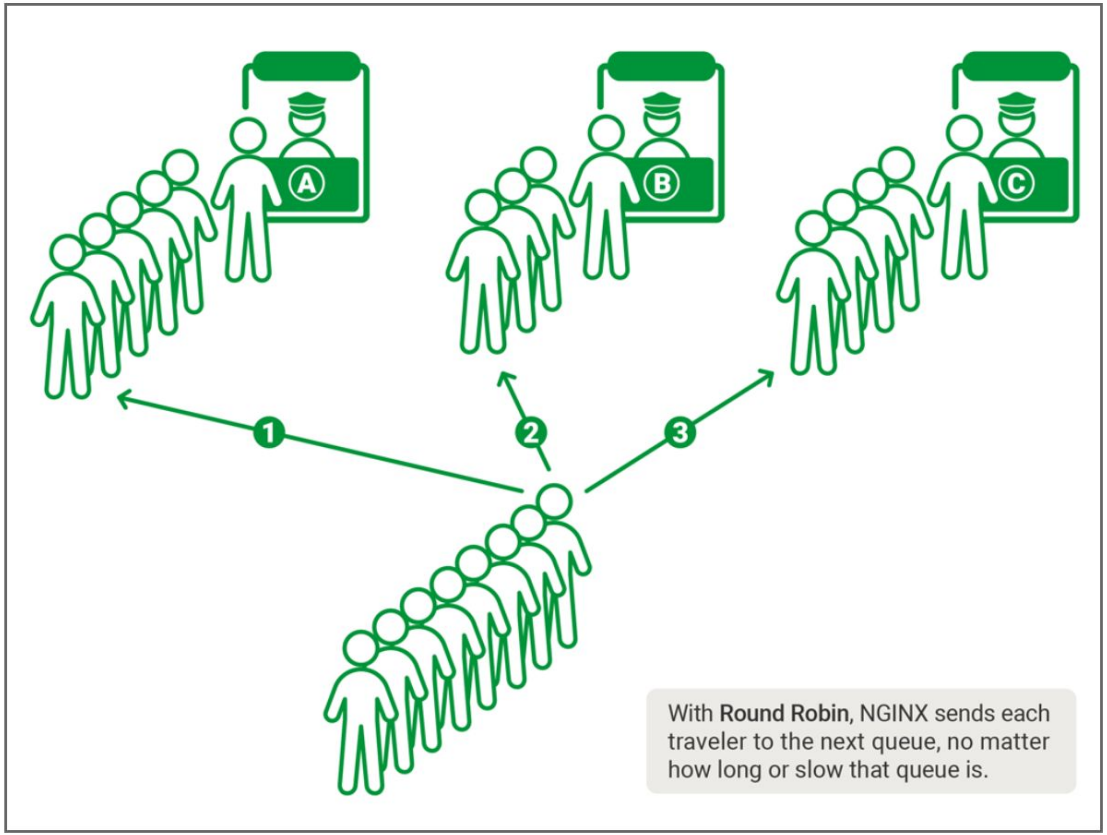
Отличия Angie

- Поддержка HTTP/3 при соединении с бэкендами
- Возможность автоматически обновлять списки проксируемых серверов, соответствующих доменному имени, и получать эти списки из DNS-записей SRV
- Режим привязки сессий (`sticky`), при котором все запросы в рамках одной сессии будут направляться на один и тот же проксируемый сервер
- Механизм плавного ввода проксируемого сервера в работу после сбоя с помощью опции `slow_start` директивы server

https://angie.software/http_upstream/ https://angie.software/stream_upstream/

Round Robin

- Перебор серверов по порядку
- Метод балансировки по умолчанию
- Поддержка весов



Round Robin

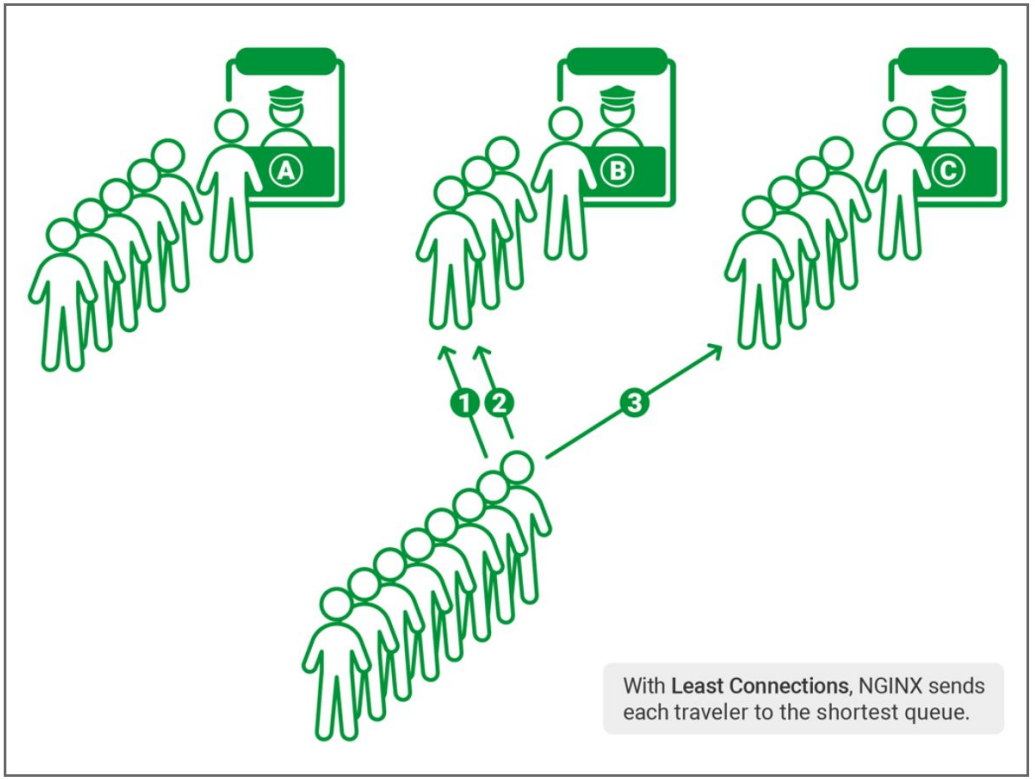
Round robin

```
upstream backend {
    server backend1.example.com;
    server backend2.example.com;
    server backend3.example.com;
}

server {
    listen 80;
    server_name example.com;
    location / {
        proxy_pass http://backend;
    }
}
```

Least Connections

- Перебор серверов по количеству соединений
- Директива: `least_conn`
- Посылает запрос на сервер с меньшей очередью
- Поддерживает веса
- Удобно для серверов разной мощности



Least Connections

Least Connections

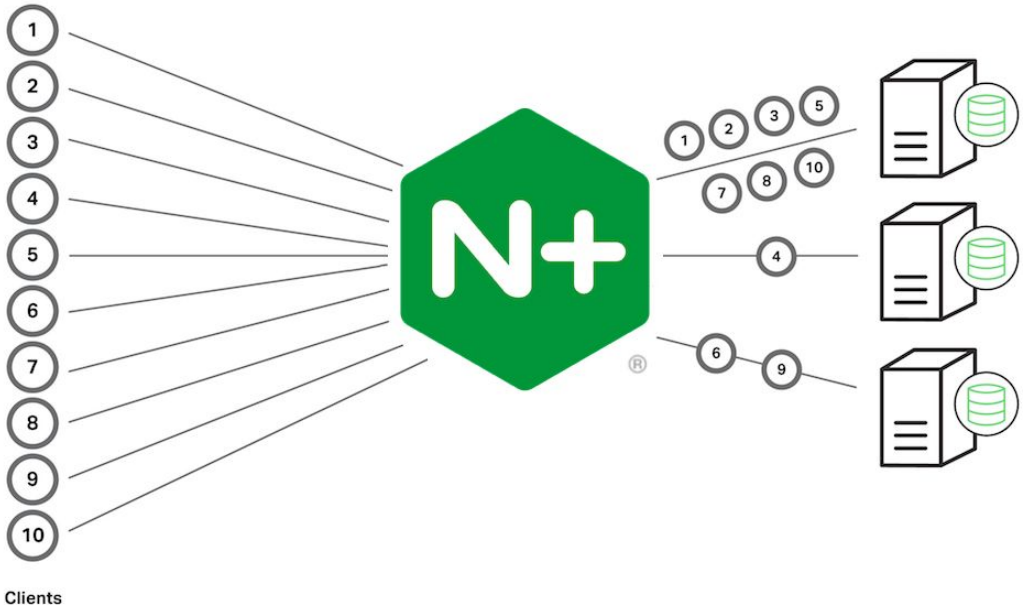
```
upstream backend {
    least_conn;
    server backend1.example.com;
    server backend2.example.com;
    server backend3.example.com;
}

server {
    listen 80;
    server_name example.com;

    location / {
        proxy_pass http://backend;
    }
}
```

IP Hash

- Распределение серверов по значению хеша по IP (весь для IPv6 и первые три октета для IPv4)
- Директива: `ip_hash`
- Нет гарантии равномерности
- Обеспечивается постоянство связи клиента с бэкендом
- Поддерживаются веса



IP Hash

IP Hash

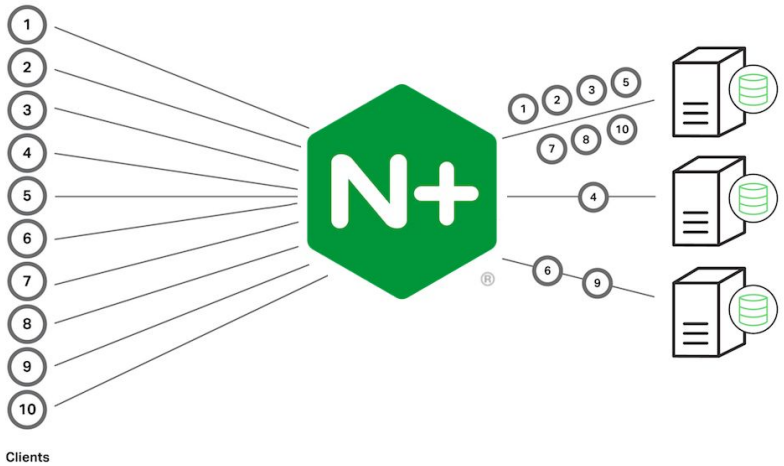
```
upstream backend {
    ip_hash;
    server backend1.example.com;
    server backend2.example.com;
    server backend3.example.com;
}

server {
    listen 80;
    server_name example.com;

    location / {
        proxy_pass http://backend;
    }
}
```

Hash

- Распределение серверов по значению хеша от переменных
- Директива: `hash`
- Обеспечивается постоянство связи клиента с бэкендом (точнее, чем `ip_hash`)
- `consistent` – постоянство выбора бэкенда при отключении или добавлении новых серверов. Использует метод хэширования Ketama
- Нет гарантии равномерности
- Поддерживаются веса



Hash

Hash

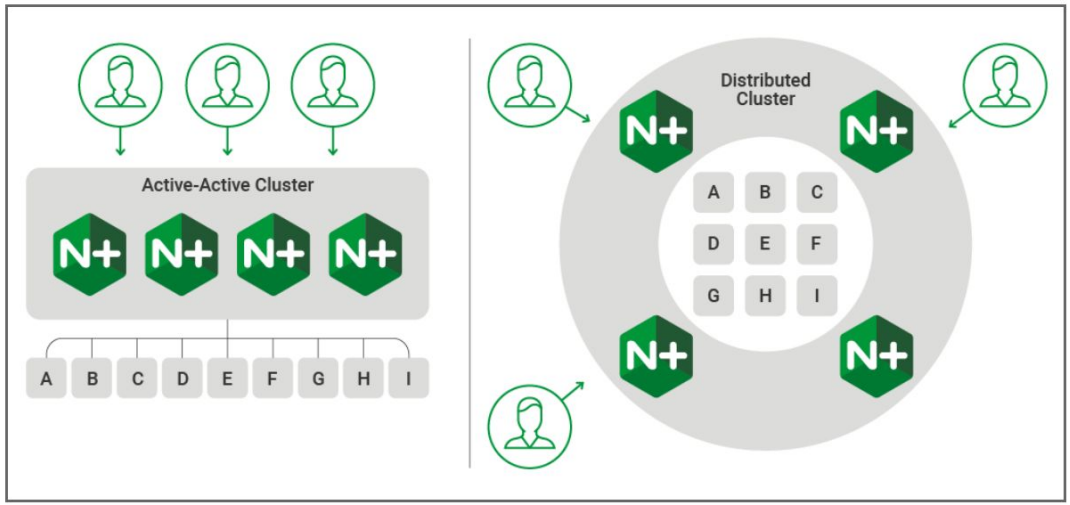
```
upstream backend {
    hash $scheme$request_uri;
    server backend1.example.com;
    server backend2.example.com;
    server backend3.example.com;
}

server {
    listen 80;
    server_name example.com;

    location / {
        proxy_pass http://backend;
    }
}
```

Random

- Произвольный выбор сервера
- Директива: `random`
- Можно выбирать два произвольных сервера (`two`) и из них по `least_conn`
- Поддержка весов
- Полезно при нескольких балансировщиках в системе



Random

Random

```
upstream backend {
    random; #two least_conn;
    server backend1.example.com;
    server backend2.example.com;
    server backend3.example.com;
}

server {
    listen 80;
    server_name example.com;

    location / {
        proxy_pass http://backend;
    }
}
```

Параметры директивы server

- `weight` – вес сервера
- `max_conns` – максимальное количество одновременных подключений
- `backup` – запасной сервер
- `down` – недоступный сервер
- `max_fails` – максимальное количество ошибок
- `fail_timeout` – время для определения недоступности сервера (`max_fails`)

https://nginx.org/ru/docs/http/ngx_http_upstream_module.html#server

Постоянные соединения к бэкендам

```
upstream http_backend {
    server 127.0.0.1:8080;
    keepalive 16;
    keepalive_requests 1000;
    keepalive_timeout 60s;
}
server {
    location /http/ {
        proxy_pass http://http_backend;
        proxy_http_version 1.1;
        proxy_set_header Connection "";
    }
    location /fastcgi/ {
        fastcgi_pass fastcgi_backend;
        fastcgi_keep_conn on;
    }
}
```

http://nginx.org/en/docs/http/ngx_http_upstream_module.html#keepalive

Серверное кэширование

nginx.conf:

```
proxy_cache_valid 1m;
proxy_cache_key $scheme$host$request_uri;
proxy_cache_path /cache levels=1:2 keys_zone=one:10m inactive=48h max_size=800m;
```

server.conf:

```
location / {
    proxy_cache one;
    proxy_cache_valid 200 1h;
    proxy_cache_lock on;
    proxy_cache_use_stale updating error timeout invalid_header http_500 http_502 http_504;
    proxy_cache_background_update on;
}
```

http://nginx.org/en/docs/http/ngx_http_proxy_module.html#proxy_cache
http://nginx.org/en/docs/http/ngx_http_fastcgi_module.html#fastcgi_cache

Спасибо за внимание!

Ссылка на презентацию