

Referência Rápida de Nginx

Blocos de servidor, proxy, SSL, balanceamento de carga, logs

Instalação

Instalar por Sistema Operacional

Ubuntu / Debian	sudo apt install nginx
RHEL / CentOS	sudo dnf install nginx
macOS	brew install nginx
Alpine	apk add nginx
Docker	docker run -p 80:80 nginx

Gerenciamento de Serviço

sudo systemctl start nginx	Iniciar o Nginx
sudo systemctl stop nginx	Parar o Nginx
sudo systemctl reload nginx	Recarregar configuração (sem downtime)
sudo systemctl enable nginx	Habilitar na inicialização
nginx -t	Testar sintaxe da configuração
nginx -T	Testar e exibir configuração completa
nginx -s reload	Sinalizar processo em execução para recarregar

Configuração Básica

Localização dos Arquivos

/etc/nginx/nginx.conf	Arquivo de configuração principal
/etc/nginx/conf.d/	Configurações adicionais de sites (*.conf)
/etc/nginx/sites-available/	Configurações de sites disponíveis (Debian)
/etc/nginx/sites-enabled/	Links simbólicos para configurações ativas
/var/log/nginx/	Logs de acesso e de erro
/var/www/html/	Raiz de documentos padrão

Configuração Mínima

```
server {
    listen 80;
    server_name example.com;
    root /var/www/mysite;
    index index.html;
}
```

Estrutura de Configuração

http { }	Configurações do servidor HTTP (nível superior)
server { }	Definição de host virtual
location { }	Bloco de correspondência de URI
upstream { }	Grupo de servidores back-end
events { }	Configurações de tratamento de conexões

Blocos de Servidor

Hosts Virtuais Baseados em Nome

```
server {
    listen 80;
    server_name site-a.com;
    root /var/www/site-a;
}
server {
    listen 80;
    server_name site-b.com;
    root /var/www/site-b;
}
```

Padrão e Curinga

```
server {
    listen 80 default_server;
    server_name _;
    return 444; # drop connection
}
```

Redirecionamento HTTPS

```
server {
    listen 80;
    server_name example.com;
    return 301 https://$host$request_uri;
}
```

Blocos de Location

Prioridade de Correspondência (maior para menor)

= /path	Correspondência exata (maior prioridade)
^~ /path	Correspondência de prefixo, ignora regex
~ regex	Regex sensível a maiúsculas
~* regex	Regex insensível a maiúsculas
/path	Correspondência de prefixo (menor prioridade)

Exemplos de Location

```
location = / {
    # exact root only
}
location /api/ {
    proxy_pass http://backend;
}
location ~* \.(jpg|png|gif)$ {
    expires 30d;
}
```

try_files

```
location / {
    try_files $uri $uri/ /index.html;
}
```

Tenta arquivo, depois diretório, depois fallback -- essencial para SPAs

Proxy Reverso

Proxy Básico

```
location /api/ {
    proxy_pass http://localhost:3000/;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
}
```

Proxy para WebSocket

```
location /ws/ {
    proxy_pass http://localhost:3000;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "upgrade";
}
```

Diretivas de Proxy

proxy_pass	URL do back-end
proxy_set_header	Passar cabeçalhos personalizados ao back-end
proxy_read_timeout	Timeout para resposta do back-end (padrão 60s)
proxy_buffering off	Desabilitar buffer de resposta
proxy_redirect	Reescrever cabeçalhos Location do back-end

SSL / TLS

Servidor HTTPS

```
server {
    listen 443 ssl;
    server_name example.com;

    ssl_certificate /etc/ssl/certs/example.crt;
    ssl_certificate_key /etc/ssl/private/example.key;
    ssl_protocols TLSv1.2 TLSv1.3;
    ssl_ciphers HIGH:!aNULL:!MD5;
}
```

Let's Encrypt com Certbot

```
sudo certbot --nginx -d example.com
sudo certbot renew --dry-run
```

Boas Práticas de SSL

ssl_protocols TLSv1.2 TLSv1.3	Desabilitar versões antigas do TLS
ssl_prefer_server_ciphers on	Servidor escolhe a cifra
ssl_session_cache shared:SSL:10m	Reutilização de sessão para desempenho
add_header Strict-Transport-Security	Cabeçalho HSTS
ssl_stapling on	OCSP stapling para handshake mais rápido

Balanceamento de Carga

Bloco Upstream

```
upstream backend {
    server 10.0.0.1:3000;
    server 10.0.0.2:3000;
    server 10.0.0.3:3000;
}
server {
    location / {
        proxy_pass http://backend;
    }
}
```

Métodos de Balanceamento de Carga

(padrão)	Round-robin
least_conn	Menor número de conexões ativas
ip_hash	Sessões fixas por IP do cliente
hash \$request_uri	Hash consistente por URI

Opções de Servidor

weight=3	Enviar 3x mais tráfego
max_fails=3	Falhas antes de marcar como inativo
fail_timeout=30s	Tempo para marcar servidor como inativo
backup	Usar apenas quando os outros estiverem fora
down	Marcar servidor como permanentemente offline

Arquivos Estáticos e Cache

Servir Arquivos Estáticos

```
location /static/ {
    alias /var/www/assets/;
    expires 30d;
    add_header Cache-Control "public, immutable";
}
```

Referência Rápida de Nginx

Compressão Gzip

```
gzip on;
gzip_types text/plain text/css
        application/json application/javascript;
gzip_min_length 1000;
gzip_comp_level 5;
```

Diretivas de Cache

expires 30d	Definir Expires e Cache-Control max-age
expires off	Desabilitar cabeçalho expires
etag on	Habilitar cabeçalho ETag (padrão)
sendfile on	Servir arquivos eficientemente via kernel
tcp_nopush on	Otimizar envio de pacotes

Logs

Configuração de Log

```
access_log /var/log/nginx/access.log;
error_log /var/log/nginx/error.log warn;

# Custom log format
log_format main '$remote_addr - $status '
                '"$request" $body_bytes_sent';
access_log /var/log/nginx/access.log main;
```

Níveis do Log de Erros

debug	Detalhado (requer --with-debug)
info	Informativo
notice	Normal mas relevante
warn	Avisos
error	Erros (padrão)
crit	Problemas críticos

Log Condicional

```
map $status $loggable {
    ~^[23] 0;
    default 1;
}
access_log /var/log/nginx/access.log combined if=$loggable;
```

Ignorar respostas 2xx/3xx nos logs para reduzir volume

Segurança

Limitação de Requisições

```
limit_req_zone $binary_remote_addr
zone=api:10m rate=10r/s;

location /api/ {
    limit_req zone=api burst=20 nodelay;
}
```

Controle de Acesso

```
location /admin/ {
    allow 192.168.1.0/24;
    deny all;
}
```

Cabeçalhos de Segurança

X-Frame-Options DENY	Prevenir clickjacking
X-Content-Type-Options nosniff	Prevenir sniffing de MIME
X-XSS-Protection "1; mode=block"	Filtro XSS (navegadores legados)
Content-Security-Policy	Controlar fontes de carregamento de recursos
Referrer-Policy no-referrer	Controlar informações de referência

Padrões Comuns

SPA (Aplicação de Página Única)

```
location / {
    root /var/www/app;
    try_files $uri $uri/ /index.html;
}
```

Cabeçalhos CORS

```
location /api/ {
    add_header Access-Control-Allow-Origin *;
    add_header Access-Control-Allow-Methods
        "GET, POST, PUT, DELETE, OPTIONS";
    if ($request_method = OPTIONS) {
        return 204;
    }
    proxy_pass http://backend;
}
```

Variáveis Úteis

\$host	Cabeçalho Host da requisição
\$uri	URI atual (normalizado)
\$request_uri	URI original com query string
\$remote_addr	Endereço IP do cliente
\$scheme	http ou https
\$args	Parâmetros da query string
\$status	Código de status da resposta