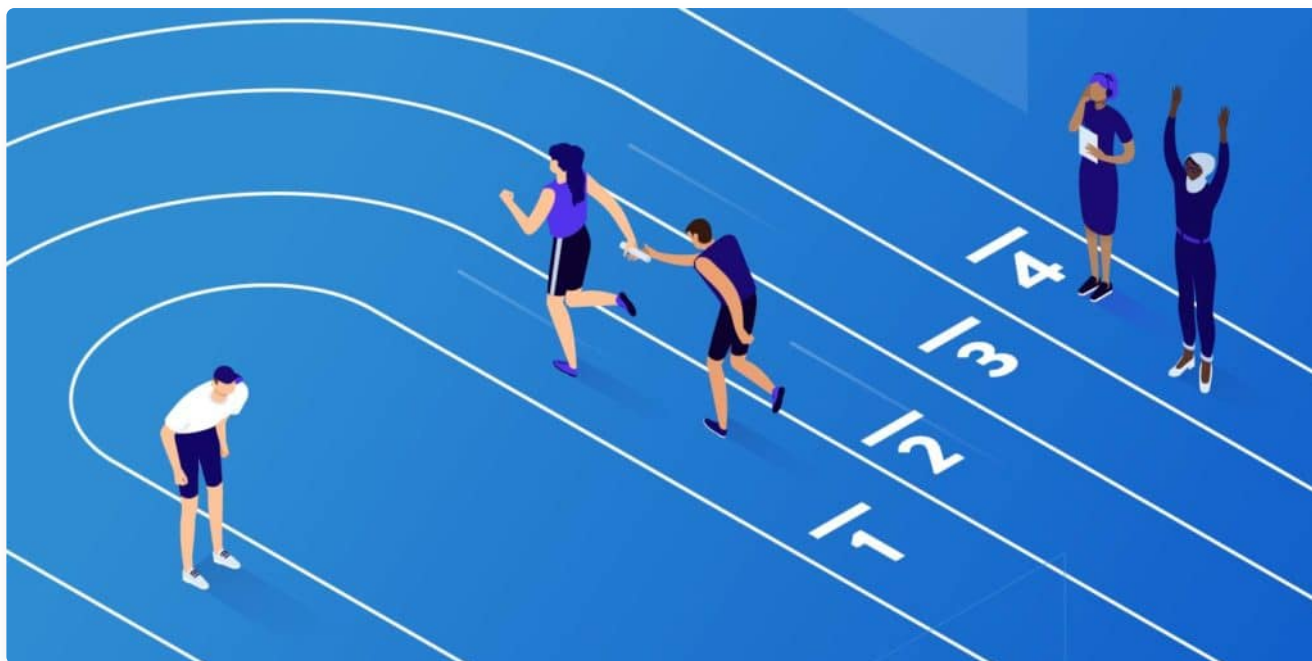


BLOG

Como Configurar um Proxy Reverso (Passo a Passo para Nginx e Apache)

Salman Ravoof, Setembro 27, 2022

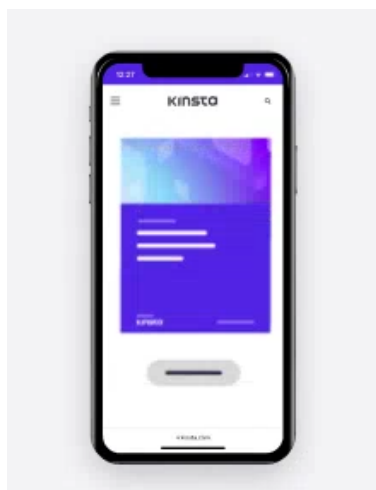


 Shares



Um proxy reverso fica na frente de um servidor web e recebe todas as solicitações antes que elas cheguem ao servidor de origem. Funciona de forma semelhante a um proxy forward, exceto neste caso, é o servidor web que utiliza o proxy e não o usuário ou cliente. Os proxy

Por exemplo, você pode ter um site nãoWordPress hospedado no domínio `example.com` no Servidor A e ter seu blog rodando no WordPress no `example.com/blog` URL hospedado no Servidor B. Você pode conseguir isso adicionando um proxy reverso para o servidor que hospeda seu site principal. Você pode configurar o proxy reverso para redirecionar as solicitações do blog para um servidor diferente (por exemplo, um [host WordPress gerenciado como Kinsta](#)).



Está com pressa? Salve este artigo como PDF.

Baixar

Neste artigo, você aprenderá o básico de servidores proxy reversíveis, como eles funcionam, quais são seus maiores benefícios e como você pode usá-los para acelerar e proteger seu site WordPress.

Entusiasmado? Vamos começar!

Tabela de Conteúdos

- [O que é uma Proxy Reverso?](#)
- [Benefícios da utilização de um Proxy Reverso](#)
- [Os mais populares proxy reverso](#)
- [Proxy Reverso: Casos de uso para sites WordPress](#)

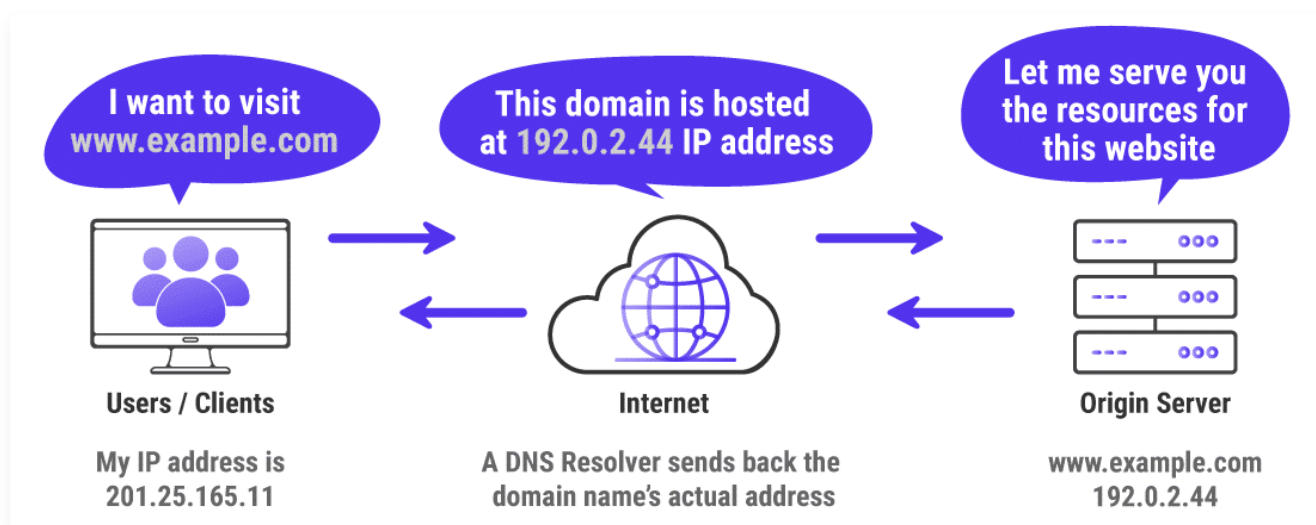
- [Como configurar o Apache como um Proxy Reverso](#)
- [Limitações de um Proxy Reverso](#)
- [Escolhendo entre um CDN e um Proxy Reverso](#)

Teste uma implantação em nossa moderna Hospedagem de Aplicativos. Por um tempo limitado, seus primeiros \$20 são por nossa conta.

O que é uma Proxy Reverso?

Para entender o que é um servidor proxy reverso, você precisa primeiro conhecer seu papel e se familiarizar com todos os seus termos relacionados.

Quando você navega na web normalmente inserindo um [nome de domínio](#) ou clicando em um link, seu navegador/dispositivo se conecta diretamente ao servidor do site e começa a baixar seus recursos.

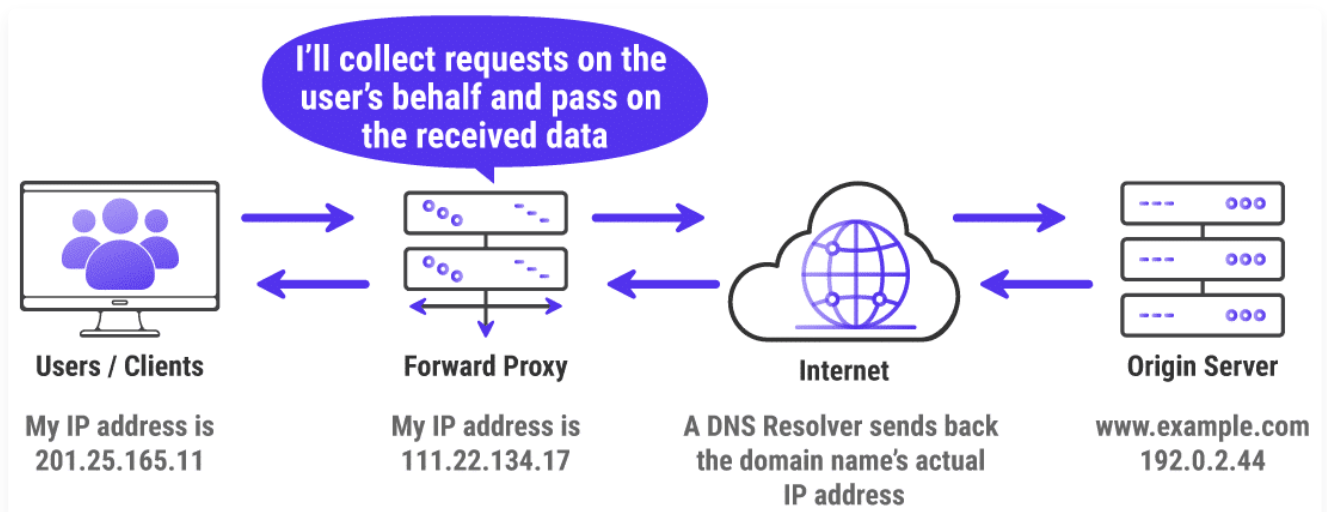


— Como a navegação na Internet funciona normalmente

Se você quiser anonimizar seu endereço IP dos sites que visitar, então você pode usar um servidor proxy para enviar todas as suas solicitações a ele primeiro. Ele encaminhará suas solicitações para o [resolvedor DNS](#) e depois fará o download dos recursos do site a partir de seu servidor de origem.

Ver planos

Depois disso, ele passará esses recursos para o seu dispositivo. Isto é chamado de forward proxy.



— Como funciona um servidor forward proxy

Você está completamente escondido do site, pois ele pensa que seu pedido é originário do procurador.

Info

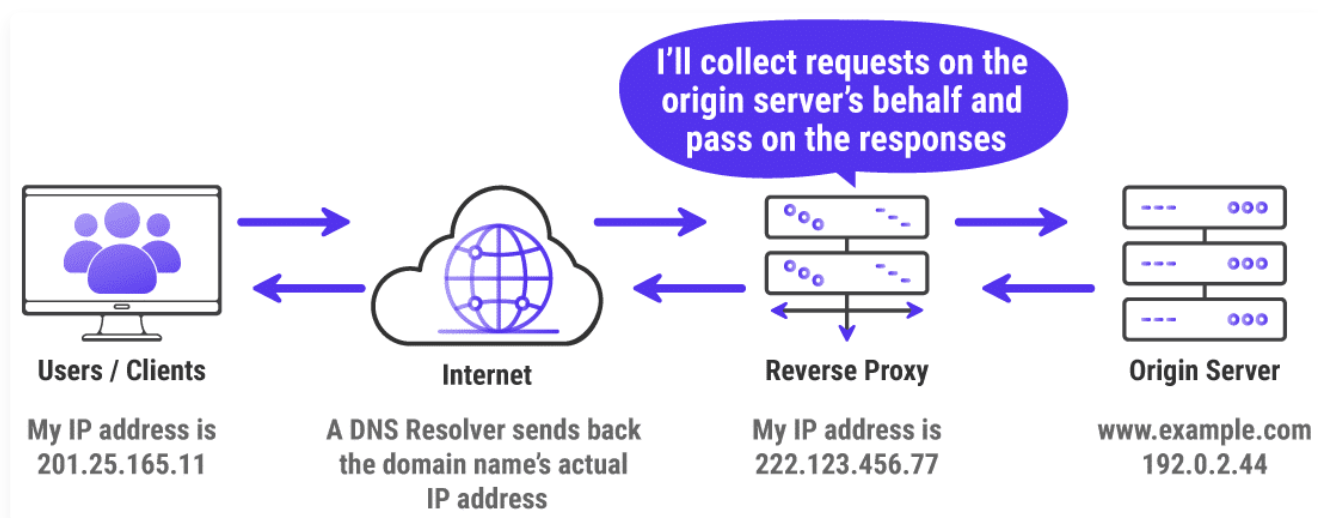
Devido à forma como alguns [provedores de hospedagem como Kinsta contam as visitas ao site](#), eles exigem que os usuários definam um cabeçalho para notificar seu endereço IP real ao servidor de origem. Portanto, os benefícios de privacidade não se aplicam em casos específicos como estes.

Além de aumentar a privacidade do usuário, um forward proxy é usado principalmente para contornar as restrições de conteúdo geográfico. Por exemplo, se você quiser assistir a um

Um forward proxy funciona quase da mesma forma que uma Rede Privada Virtual (VPN), mas [são tecnologias distintas](#) com casos de uso único (às vezes podem, no entanto, se sobrepor).

Proxy Reverso Server vs Forward Proxy Server

Um servidor proxy reverso atua como uma frente para que o servidor de origem mantenha o anonimato e [aumente a segurança](#), da mesma forma que um usuário/cliente pode usar um proxy avançado para conseguir o mesmo. Ele garante que nenhum usuário ou cliente se comunique diretamente com o servidor de origem.

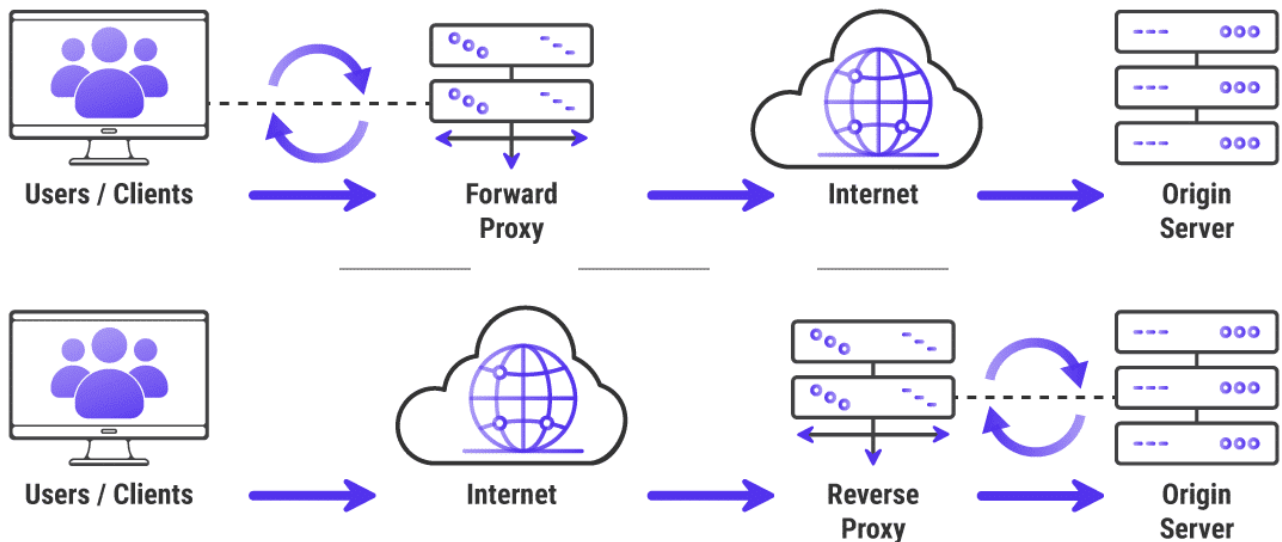


— Como funciona um servidor proxy reverso

A diferença entre um procurador avançado e um procurador reverso é pequena, mas eles funcionam de forma diferente.

Ambos podem trabalhar em conjunto, pois não há sobreposição entre seu funcionamento. Normalmente, os usuários/clientes utilizam um proxy forward, enquanto os servidores de origem utilizam um proxy reverse.

Forward Proxy vs Reverse Proxy



— Servidores de Proxy Frente vs. Proxy Reversa

Como um administrador do servidor pode controlar como funciona o proxy reverso, você pode usá-lo para habilitar muitas características úteis.

Listaremos todos os seus benefícios mais tarde, neste post.

Por que usar um Proxy Reverso?

Muitas empresas, especialmente as grandes empresas, utilizam websites feitos sob medida para suas necessidades únicas e que não estão rodando no WordPress. Alguns exemplos incluem sites de bancos e seguradoras.

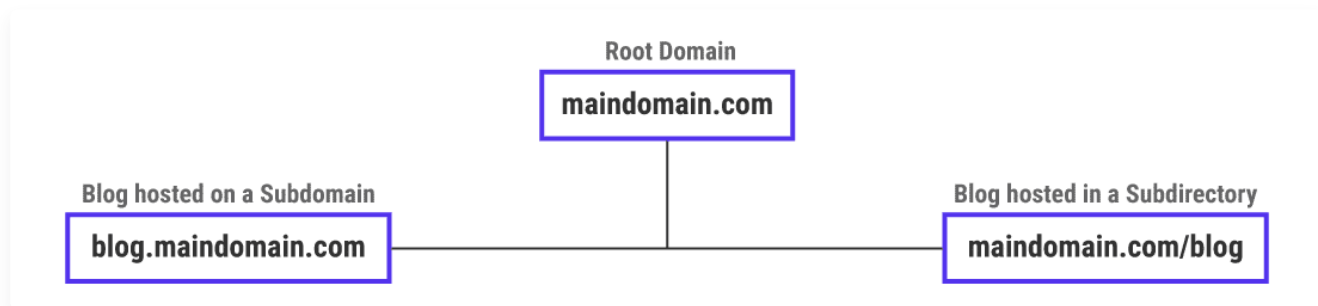
Em outros casos, uma empresa pode hospedar seu site em um serviço externo que não permite a instalação de nenhum software externo (por exemplo, [WordPress](#)). Normalmente, estes são pequenos e médios varejistas que utilizam uma [plataforma de comércio eletrônico](#) como o [Shopify](#).

Como o WordPress tem [características robustas de CMS](#), muitas empresas, incluindo grandes empresas com sites sob medida, podem [preferir hospedar seus blogs usando o WordPress](#).

Uma maneira de contornar este problema é [instalar o WordPress no subdomínio do site principal](#) e estruturar os [menus de navegação](#) de modo que os usuários possam alternar

Entre os outros pontos, o site principal é o site principal.

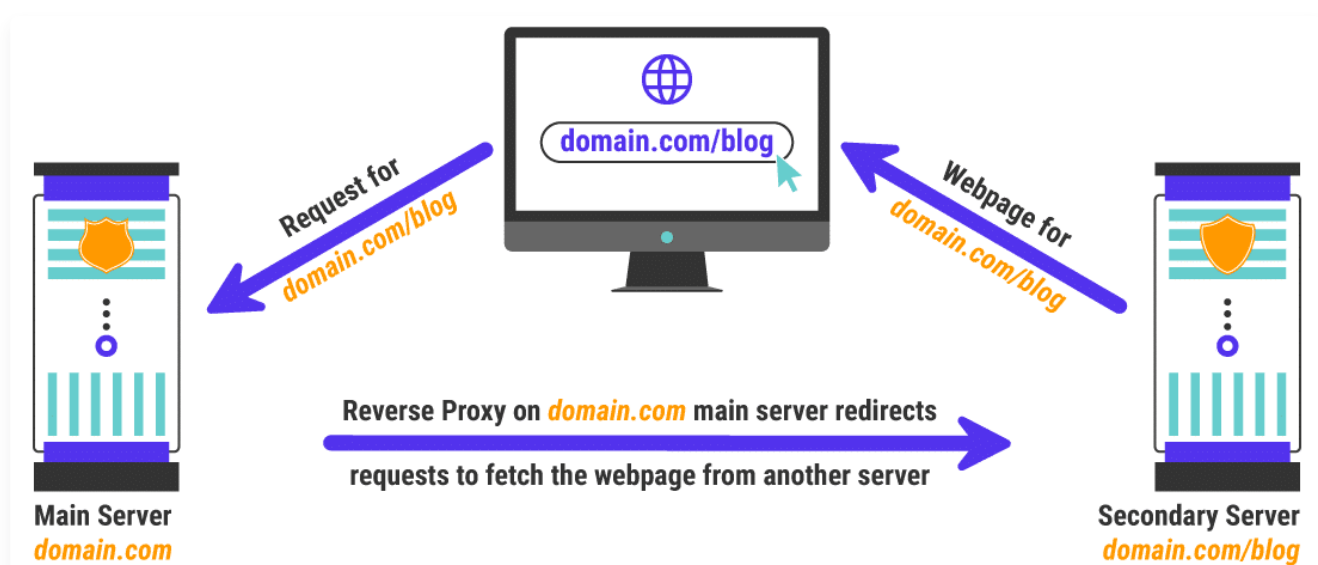
Como os subdomínios se comportam como um domínio único, isso pode afetar [a SEO de seu site](#). Mesmo que o Google trate ambos subdomínios e subdiretórios igualmente, é necessário mais esforço para otimizar um site para classificação em motores de busca se ele estiver [hospedado em um subdomínio do que se ele estiver hospedado em um subdiretório](#).



— Duas abordagens para hospedar blogs em um site

O [Google reafirmou](#) que trata os subdomínios e subdiretórios igualmente, mas alguns especialistas em SEO não concordam com isso. E mesmo que isso não afete o SEO do site, um site hospedado em um subdiretório é simplesmente mais fácil de manter.

É por isso que você pode usar um proxy reverso para redirecionar solicitações para o blog do site hospedado em um servidor separado. Por exemplo, um banco pode hospedar seu site principal em seus servidores com segurança, mas também pode hospedar seu [blog com WordPress](#) separadamente em um host WordPress gerenciado como o [Kinsta](#).



— Um exemplo de um caso de uso de proxy reverso

A unificação de dois sites diferentes sob um único nome de domínio é uma das principais vantagens do uso de um proxy reverso. Ela ajuda as marcas a manter seus sites organizados, profissionais e a manter a credibilidade.

“ Saiba como um proxy reverso pode melhorar o ⚡ desempenho, 🔒 segurança & 🧡 confiabilidade... tudo neste guia ⬇️

CLIQUE PARA TWEETAR

Benefícios da utilização de um Proxy Reverso

Além do caso de uso acima, os procuradores reversos também concedem muitos outros benefícios. A seção abaixo discute algumas de suas principais vantagens.

Balanceamento de carga

Um único servidor de origem não pode lidar com todo o [tráfego de entrada](#) para um website com milhões de visitantes únicos diários. Nesses casos, você pode distribuir o tráfego de forma inteligente entre um pool de muitos servidores. Normalmente, todos os servidores hospedarão o mesmo conteúdo para eliminar um único ponto de falha, tornando o site mais confiável.

Um proxy reverso é uma ótima maneira de configurar isso, pois pode receber o tráfego de entrada antes de chegar ao servidor de origem. Se o servidor de origem estiver sobrecarregado ou falhar completamente, ele pode distribuir o tráfego para outros servidores sem afetar a funcionalidade do site.

Os substitutos reversos também podem direcionar as solicitações recebidas para vários servidores, com cada servidor executando uma função específica para a qual está otimizado. O proxy reverso pode então reunir respostas de todos os servidores e entregá-las ao cliente.

Como usamos a maioria dos populares procuradores inversos principalmente para o balanceamento de carga, eles também são chamados de **Balanceadores de Carga**.

Balanceamento Global de Carga do Servidor (Global Server Load Balancing – GSLB)

através de [qualquer técnica de roteamento de transmissão](#), onde o proxy reverso escolhe o nó do servidor com base no tempo de viagem mais rápido entre o cliente e o servidor.

A GSLB não apenas aumenta consideravelmente a confiabilidade e segurança do site, mas também reduz a latência e os [tempos de carga](#), melhorando assim a experiência do usuário. Você pode usar o GSLB com outras técnicas de otimização de rede, tais como [Spoon Feeding](#) para liberar ainda mais os recursos computacionais dos servidores de origem.

Embora você possa configurar o Balanceamento de Carga do Servidor Global manualmente em seu servidor, ele é normalmente cuidado por CDNs dedicados, tais como [Cloudflare](#) e [KeyCDN \(que também alimenta o Kinsta CDN\)](#). Kinsta serve a todos os sites hospedados com ele através de um [Balanceador de Carga alimentado pela plataforma Google Cloud](#).

Segurança aprimorada

Os substitutos reversos podem camuflar o endereço IP e outras características dos servidores de origem. Assim, o servidor de origem de seu site pode manter melhor seu anonimato, aumentando significativamente sua segurança.

Como o proxy reverso receberá todo o tráfego antes de chegar ao servidor principal, qualquer [atacante ou hackers](#) terá mais dificuldade em atingir seu site com ameaças de segurança como [ataques DDoS](#).

Você pode [usar um firewall rígido](#) para endurecer o proxy reverso com uma segurança mais rígida contra ataques cibernéticos comuns. Sem um proxy reverso instalado, é difícil [remover malware](#) ou iniciar takedowns.

Info

Kinsta usa proxy reverso em sua arquitetura backend e oferece [correções gratuitas de hack do WordPress](#) para todos os sites que hospeda.

Um proxy reverso como o [HAProxy](#) pode adicionar autenticação de acesso HTTP básica a um servidor web que não o tenha habilitado. Você também pode usar um proxy reverso para adicionar autenticação centralizada para vários tipos de solicitações.

Você pode usar um proxy reverso para fins de aceleração da web, cachê tanto de conteúdo estático quanto dinâmico. Isto pode reduzir a carga no servidor de origem, resultando em um website mais rápido.

Por exemplo, se seu servidor de origem está nos EUA e um usuário da Europa visita seu site, então você pode servir uma versão em cache do seu site a partir de um servidor proxy reverso na Europa. Como o proxy reverso está mais próximo do usuário do que o servidor de origem, o site levará menos tempo para ser carregado, fazendo com que ele tenha um desempenho excelente.

O verniz e o Nginx FastCGI são exemplos proeminentes de proxies reversos que são usados para o conteúdo da web de caching. Se seu site é hospedado com Kinsta, [você não precisa se preocupar com o cache](#), pois Kinsta cuida de todo o trabalho de cache para você.

Compressão Superior

As respostas do servidor consomem muita largura de banda. A compressão das respostas do servidor (por exemplo, [com gzip](#)) antes de enviá-las ao cliente pode reduzir a quantidade de largura de banda necessária, acelerando as respostas do servidor através da rede.

Um proxy reverso é ideal para comprimir as respostas do servidor, já que se situa entre os servidores de origem e o cliente.

Criptografia SSL otimizada

Criptografar e decodificar pedidos SSL/TLS para cada cliente pode ser altamente tributário para o servidor de origem. Um proxy reverso pode assumir esta tarefa para liberar os recursos do servidor de origem para outras tarefas importantes, como servir conteúdo.

Outra vantagem de descarregar a [criptografia e descriptografia SSL/TLS](#) é reduzir a latência para clientes que estão geograficamente distantes do servidor de origem.

Você também pode optar por um proxy reverso com hardware especializado de aceleração SSL/TLS para otimizar ainda mais esta tarefa. Tal proxy reverso é chamado de [proxy de terminação SSL/TLS](#). Alguns servidores como o Varnish não suportam os protocolos SSL/TSL, portanto um proxy reverso de terminação SSL/TSL pode ajudar a proteger o tráfego que passa por eles.

A maioria das [ferramentas de teste A/B](#) exige que você use [bibliotecas JavaScript externas](#) para carregar suas funções. Entretanto, o carregamento de scripts de terceiros pode diminuir o tempo de carregamento de sua página e criar uma experiência agitada para os usuários.

Ao invés disso, você pode usar um proxy reverso para criar dois fluxos separados no próprio nível do servidor. Por exemplo, você pode usar os métodos `split_clients` ou `sticky route` do Nginx para controlar o redirecionamento do tráfego.

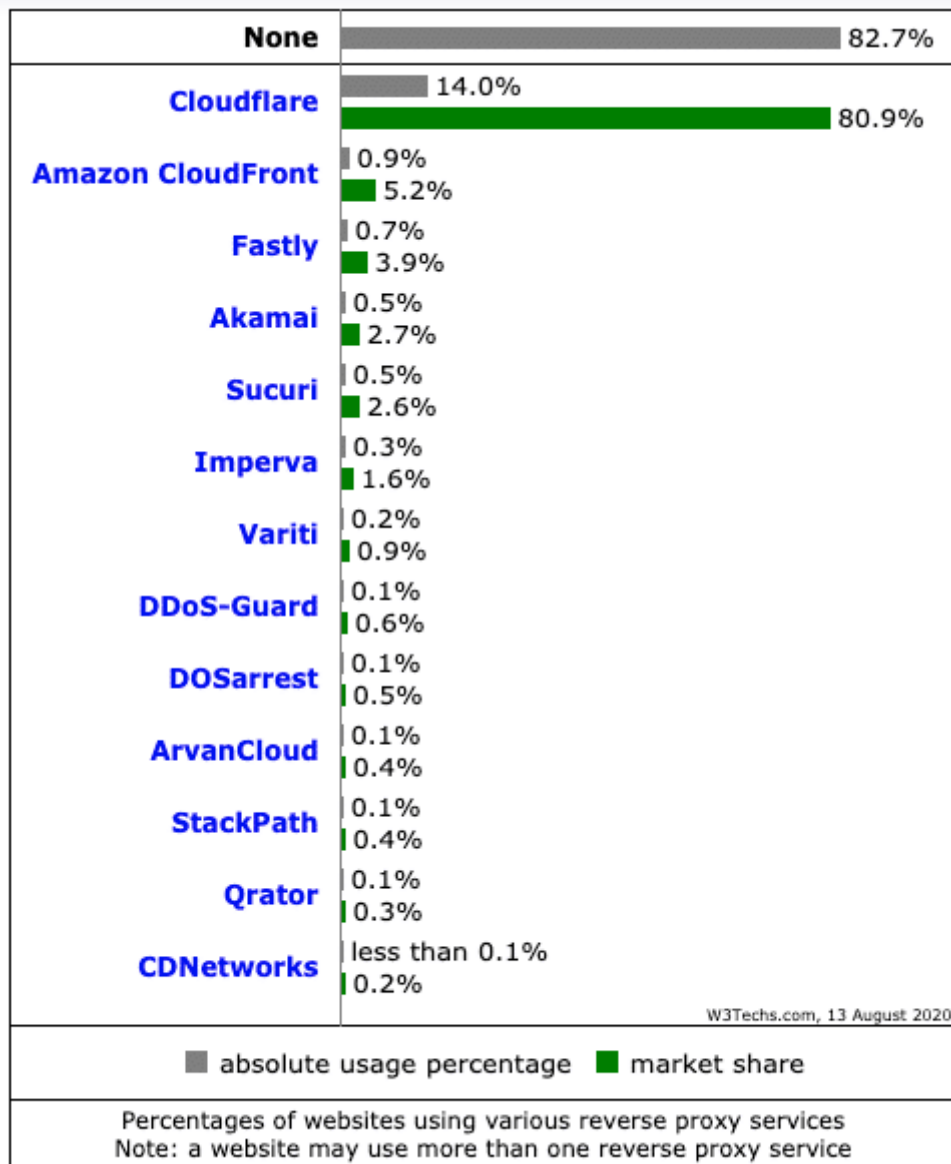
Você pode consultar os tutoriais sobre [Nginx](#) e [FreeCodeCamp](#) para saber mais sobre como realizar testes A/B com um proxy reverso.

Monitoramento e Registro de Tráfego

Um proxy reverso captura todas as solicitações que passam por ele. Assim, você pode usá-los como um hub central para monitorar e registrar o tráfego. Mesmo que você use vários servidores web para hospedar todos os componentes de seu site, o uso de um proxy reverso facilitará o monitoramento de todos os dados de entrada e saída de seu site.

Os mais populares proxy reverso

[Segundo a W3Techs](#), **quase 83%** dos websites não utilizam serviços de proxy reverso que eles monitoram.



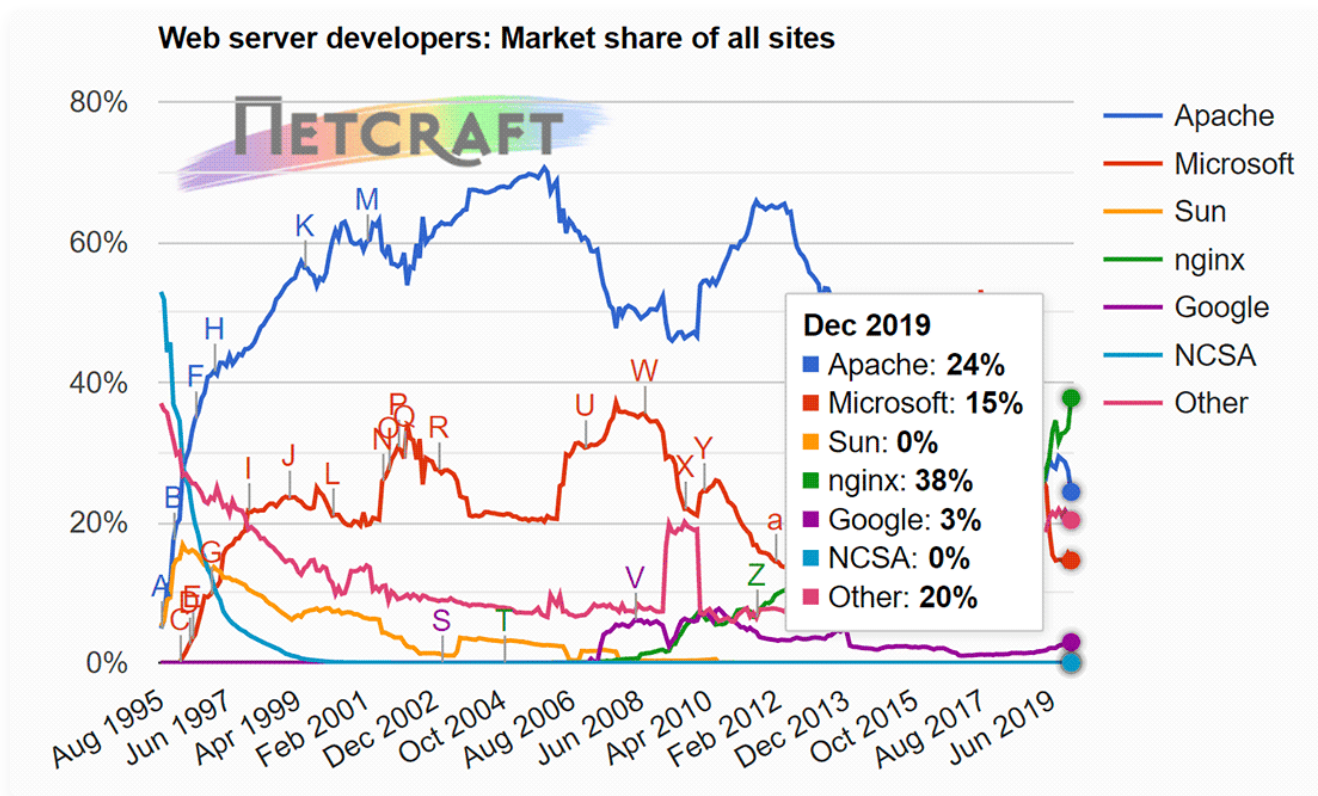
— Estatísticas de procurações invertidas utilizadas por websites (Fonte: W3Techs.com)

Dos **17%** de sites que utilizam um proxy reverso (listados acima), você notará que a maioria deles são **CDNs**. Isso porque a maioria dos procuradores reversos esconde sua existência por padrão como precaução de segurança. Portanto, você não pode confiar em serviços de monitoramento de websites como o W3Techs para descobrir quais proxies reversos são os mais populares.

A partir de nossa pesquisa e experiência, os mais populares procuradores inversos em uso atualmente são:

Nginx

[Nginx](#) é um servidor web de código aberto que também pode servir como proxy reverso. Além de ser utilizado para hospedar websites, é também uma das soluções de proxy reverso e balanceamento de carga mais utilizadas. [Segundo a Netcraft](#), mais de **479 milhões** de servidores web estavam usando Nginx em dezembro de 2019, tornando-o o líder no [mercado de servidores web](#).



— Participação no mercado de servidores Web de todos os sites (Fonte: Netcraft)

Nginx oferece todos os benefícios de proxy reverso discutidos acima, além de mais. Ele melhora o desempenho, a segurança, a confiabilidade e a escalabilidade da web. Você pode configurar o Nginx usando seu arquivo de configuração, que também é recarregável a quente. Na Kinsta, o proxy reverso Nginx é um dos [vários complementos premium que você pode usar](#).

Mas você também pode usar o Nginx Plus, uma oferta comercial, para obter acesso a opções de configuração baseadas em API e outras características adequadas para sites de grandes empresas.

A Kinsta alimenta todos os seus websites com Nginx. [Classificou em Review Signal o status de Top Tier Web hosting](#) em todas as categorias em que competiu. Algumas outras grandes empresas que usam Nginx são MaxCDN, [Cloudflare](#) e Netflix.

Configurar o Nginx como um proxy reverso básico é simples. O Nginx também fornece várias diretrizes para personalizar o proxy reverso de seu servidor de acordo com suas necessidades. Discutiremos como fazer isso em uma seção posterior. Se você for um cliente Kinsta, também aprenderá como usar um proxy reverso para sites hospedados com Kinsta na mesma seção.

Varnish

O [Varnish](#) é um proxy HTTP reverso de código aberto com um mecanismo de cache embutido. Ele é projetado principalmente para sites de alto tráfego que servem conteúdo dinâmico. Você também pode usar o Varnish como equilibrador de carga, um [firewall de aplicação web \(WAF\)](#) e um servidor de autenticação e autorização de borda.

Quer saber como aumentamos nosso tráfego em mais de 1000%?

Junte-se a mais de 20.000 outros que recebem nossa newsletter semanal com dicas privilegiadas do WordPress!

Inscreva-se agora

Funciona em todas as versões modernas do Linux e FreeBSD, sendo utilizado principalmente como fachada para servidores web Nginx ou [Apache](#). A poderosa e altamente flexível Varnish [Varnish Configuration Language \(VCL\)](#) permite definir várias características, tais como lidar com [solicitações HTTP](#), cache e conexão a um ou mais servidores web.

Por esta razão, muitos CDNs usam o Varnish como sua base principal para entregar conteúdo rapidamente.

O Varnish também suporta [Edge Side Includes \(ESI\)](#), uma linguagem que ajuda você a reutilizar seções de uma página da web em outras páginas da web. Se seu site usa muito conteúdo repetido em páginas diferentes, o ESI pode ajudá-lo a [acelerar o tempo de](#)

Você pode estender o Varnish com seus vários [módulos \(VMODs\)](#). Vá até o [tutorial oficial do Varnish](#) para aprender como configurar o Varnish como um proxy reverso para WordPress.

Apache Traffic Server

O [Apache Traffic Server](#) é um servidor proxy de cache de código aberto. É popular por suas características rápidas e escaláveis. Foi um produto comercial desenvolvido pelo Yahoo! há muito tempo, mas eles o tornaram open source e o doaram para a Fundação Apache para manutenção.

Várias grandes redes de conteúdo e CDNs como Comcast, Akamai, LinkedIn, Yahoo e Apple utilizam o Apache Traffic Server para alimentar sua tecnologia.

Você também pode usar o [Apache HTTP Server \(Apache httpd\)](#), um daemon servidor HTTP, para configurar um proxy reverso em seu servidor web. Além de atuar como um servidor web básico, ele também ajuda você a servir conteúdo estático e dinâmico aos usuários. Você aprenderá como configurar o Apache como um proxy reverso mais adiante neste artigo.

HAProxy

HAProxy é um proxy reverso de código aberto e equilibrador de carga. Foi projetado para se integrar com a maioria das arquiteturas de servidores web existentes, incluindo distribuições Linux e plataformas de nuvem. Similar ao Nginx, o HAProxy usa um modelo de E/S impulsionado por eventos e suporta a divisão de solicitações através de múltiplos processos de trabalhadores.

Para solicitações HTTP, o HAProxy tem um desempenho excepcionalmente bom mesmo sob cargas pesadas. Alguns dos [sites de maior tráfego](#) na Internet, como Airbnb, Reddit, Instagram, Stack Overflow, Tumblr, GitHub e Imgur usam HAProxy para entregar seus sites de forma eficiente.

Discutir como implementar o HAProxy está além do escopo deste artigo, mas você pode [consultar a documentação deles](#) para entender como funciona.

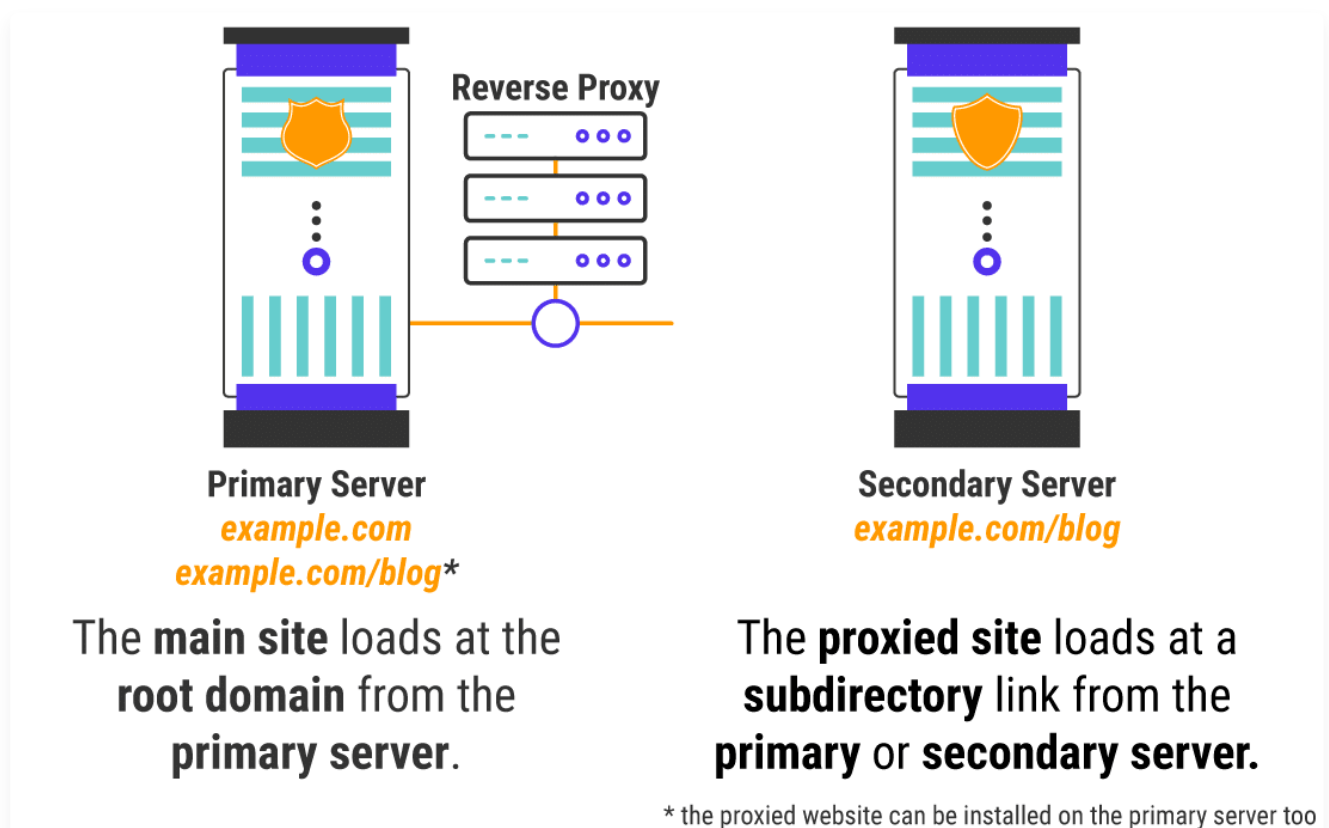
Nota: [Traefik](#) e [Envoy](#) são duas outras alternativas de código aberto ao HAProxy. Eles são ambos proxies reversíveis de alto desempenho e balanceadores de carga com muitas características avançadas.

Alguns outros procuradores populares são AWS Elastic Load Balancer, GLBC, [DigitalOcean](#)

substitutos inversos e balanceadores de carga em uso atualmente, você pode [verificar Stackshare.io](#).

Proxy Reverso: Casos de uso para sites WordPress

Há principalmente três casos de uso para empregar um proxy reverso para sites WordPress, incluindo sites hospedados em Kinsta.



— Carregando um 'Site principal' versus um 'Site procurado'.

Usaremos Nginx apenas para este exemplo, pois é o proxy reverso mais popular usado para sites WordPress atualmente. Mas os mesmos princípios básicos se aplicarão a outros proxy reverso.

A instalação, configuração e suporte de proxies reversíveis são muitas vezes desafiadores. Por este motivo, a Kinsta oferece [uma assinatura mensal adicional de \\$50](#) para cada proxv

1. Sites Principais e Sites Proxied Hospedados no Mesmo Servidor

Se tanto o site principal quanto o site proxy estiverem hospedados no mesmo servidor, o site principal pode ser executado em uma [instalação do WordPress](#), enquanto uma instalação separada do WordPress alimenta o site proxied.

Como você terá acesso a ambos os sites e seu servidor web compartilhado, você pode configurar as regras de proxy reverso para o site principal, e então configurar o site proxied para carregar a partir do proxy reverso.

Se você estiver hospedando estes dois sites em Kinsta, então você pode contactar a equipe de suporte de Kinsta e solicitar que eles configurem o proxy reverso para você. Aqui está o procedimento que você precisa seguir:

- Certifique-se de que tanto o site principal quanto o site procurado estejam hospedados em Kinsta. Se não estiverem, então você pode migrar ambos os sites para o ambiente de Kinsta, seja manualmente ou apresentando um [pedido de migração](#).
- Abra um ticket de suporte e forneça à equipe de suporte da Kinsta uma descrição clara da configuração do domínio. Levarei aproximadamente um dia útil para configurar o proxy reverso.
- Kinsta estabelecerá as regras de proxy reverso relevantes no site principal e configurará o site proxied para carregar sobre o proxy reverso.

Aqui estão as diretrizes padrão de proxy reverso Nginx usadas pela Kinsta para carregar um site de subdiretório sobre um proxy reverso:

```
location ^~ /subfolder/ {  
    proxy_pass http://subfolder.domain.com;  
    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;  
}
```

No código acima, é necessário substituir a **/subfolder/** localizador pelo nome real do

`http://subfolder.domain.com` deve corresponder à URL utilizada para apontar o proxy reverso em direção ao site procurado.

A diretiva de `location` inclui símbolos de carpete e til (^~) para dizer à Nginx que se ela encontrar a corda definida, deve parar de procurar outras correspondências e usar as diretivas listadas aqui. Saiba mais sobre as [diretivas de proxy reversa da Nginx em sua documentação](#). Em seguida, você precisa configurar o site proxy proxy para carregar sobre o proxy reverso. Aqui estão os passos padrão seguidos pela Kinsta para configurar o site procurado:

- Criar um subdiretório no caminho de onde o site procurado é carregado. Todos os arquivos do site procurado são movidos para este subdiretório.
- Atualizar os arquivos de configuração do servidor web para definir o novo subdiretório como diretório principal do site proxied. Além disso, é necessário adicionar uma regra de reescrita para remover o subdiretório do URI da solicitação para cada solicitação recebida.
- Atualizar todas as URLs no banco de dados do site proxy para corresponder às URLs do site ao vivo (por exemplo, `example.com/blog`).
- Edite o arquivo `wp-config.php` do site proxy com a definição `$_SERVER['HTTP_HOST']` , apontando-o para a URL do site principal.
- Se você estiver usando um certificado SSL, então você precisa definir regras estritas no arquivo `wp-config.php` para evitar loops de redirecionamento.

Nota: Um site proxied não pode criar URLs que duplicam o mesmo subdiretório sob o qual o site proxied carrega. Por exemplo, um site proxied em `example.com/blog` não pode criar uma página ou diretório em `example.com/blog/blog/blog` .

Precisa de uma hospedagem rápida, segura e amigável ao desenvolvedor para seus sites? Kinsta é construído com os desenvolvedores do WordPress em mente e fornece muitas ferramentas e um poderoso painel de controle. [Confira nossos planos](#)

2. Somente o Site Proxied Hospedado em Seu Servidor

Se você só tem acesso ao site proxy e seu servidor web, então você precisa entrar em contato com o administrador do servidor do site principal e pedir-lhes que estabeleçam as regras de proxy reverso para você.

Para fazer isso, você deve seguir os mesmos passos descritos acima, exceto neste caso, você deve configurar as regras em dois servidores diferentes.

`blog.example.com`) para carregar o site proxy sobre um link de subdiretório (por exemplo, `exemplo.com/blog`).

Após configurar seu site procurado em Kinsta, você pode [entrar em contato com a equipe de suporte de Kinsta](#) para configurar o site procurado para carregar por meio de um proxy reverso. Entretanto, a configuração do proxy reverso está fora do [escopo do suporte Kinsta](#), uma vez que somente o administrador do servidor pode cuidar dele.

3. Somente o site principal hospedado em seu servidor

Se você só tem acesso ao site principal e seu servidor web, então você deve configurar o proxy reverso e configurar suas regras para carregar o site proxied a partir de um host externo. A instalação e configuração do site proxied para carregar sobre o proxy reverso é de responsabilidade do administrador do servidor secundário.

Ter seu site principal hospedado na Kinsta lhe dará acesso à [equipe de suporte da Kinsta](#). Você pode levantar um ticket de suporte com eles para adicionar as regras padrão de proxy reverso listadas anteriormente neste artigo. Você também pode ter qualquer customização adicional a essas regras, se necessário.

Neste cenário, você é totalmente responsável pela configuração do site procurado para carregá-lo corretamente sobre o proxy reverso.

Como configurar o Nginx como um Proxy Reverso

Se a Kinsta não hospeda seu site e você administra seus servidores, então você mesmo deve configurar o proxy reverso e configurá-lo para apontar para o site proxied.

Dependendo do sistema operacional do seu servidor web, você pode instalar o Nginx de forma diferente. Para distribuições Linux, você pode [usar vários pacotes Nginx](#) com base na versão de sua distribuição Linux.

No exemplo abaixo, instalamos o site principal no domínio `example.com` , enquanto o site WordPress proxied está instalado no subdomínio `blog.domain.com` . Ambos são alimentados pelo Apache em um servidor web rodando no Ubuntu 18.04. Vamos instalar e configurar o Nginx como um proxy reverso no servidor principal.

Para começar, [acesse o terminal do seu servidor via SSH](#). Depois, use o comando **apt-get** para atualizar a lista de pacotes de sua distribuição e instalar o Nginx em seu servidor web.

```
sudo apt update
sudo apt install nginx
```

Em seguida, você precisa configurar o Nginx para proxy de solicitações de domínios hospedados no Apache. Para fazer isso, crie um novo arquivo host virtual. Aqui, eu estou usando o editor **nano** para adicionar o código, mas você pode usar qualquer [editor de código de sua escolha](#).

```
sudo nano /etc/nginx/sites-available/example.com.conf
```

Em seguida, defina as diretrizes Nginx para encaminhar solicitações ao Apache, adicionando os seguintes **server {...}** e blocos de **location** :

```
server {
    listen      80;
    server_name example.com www.example.com;
    index       index.php;
    root        /var/www/example.com/public    # fallback for index.php
    location / {
        try_files $uri $uri/ /index.php?$query_string;
    }location /blog {
        proxy_pass http://blog.domain.com:proxy http version 1.1;
```

```
# Proxy headers
proxy_set_header Upgrade          $http_upgrade;
proxy_set_header Connection       "upgrade";
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_set_header X-Forwarded-Host  $host;
proxy_set_header X-Forwarded-Port  $server_port;

# Proxy timeouts
proxy_connect_timeout             60s;
proxy_send_timeout                60s;
proxy_read_timeout               60s;
}
```

No código acima, estou definindo um subdiretório **exemplo.com/blog** link que será servido pelo servidor Apache. Certifique-se de usar o endereço IP público (ou URL) de seu website proxy na diretiva **proxy_pass**. No meu caso, meu website proxy está hospedado no subdomínio **blog.domain.com**.

Nota: Certifique-se de que o website proxy está instalado e pronto para ser servido antes de fazer qualquer mudança.

Você pode saber mais sobre todas as [diretrizes de proxy reversa usadas aqui](#) no índice detalhado de diretrizes da Nginx.

Salvar o arquivo hospedeiro virtual. Em seguida, ative o novo host virtual criando um link simbólico para os arquivos chamados **example.com.conf** tanto no diretório **/etc/nginx/sites-available** quanto nos diretórios **/etc/nginx/sites-enabled**.



Depois disso, teste o Nginx para verificar se há erros de configuração.

```
sudo nginx -t
```

Se não houver erros, recarregue o Nginx para impor as mudanças.

```
sudo systemctl reload nginx
```

Você criou com sucesso o Nginx para trabalhar agora como um proxy reverso. Para confirmar isto, você pode usar a função [phpinfo\(\)](#) para verificar as variáveis PHP carregadas quando visitar seu site proxy.

Sob as variáveis `SERVER_SOFTWARE` e `DOCUMENT_ROOT` PHP, você verá que o Apache serve a este domínio no backend. Mas as variáveis `HTTP_X_REAL_IP` e `HTTP_X_FORWARDED_FOR` PHP confirmam que o Nginx foi usado como proxy reverso para encaminhar as solicitações.

Você pode acelerar o serviço de seu site WordPress sobre Nginx usando os módulos `fastcgi_cache` e [ngx_cache_purge](#). Enquanto o primeiro módulo irá fazer o cache de seu site, o segundo módulo irá purgar automaticamente o cache com base em eventos específicos (por exemplo, publicação ou edição de um post/página do WordPress).

Você pode usar o plugin [Nginx Cache Controller](#) WordPress para controlar o cache do servidor proxy Nginx diretamente de seu painel de administração do WordPress. Se você estiver usando uma instalação do WordPress Multisite, então você pode usar o plugin [Nginx Helper](#) para fazer o mesmo.

Como configurar o Apache como um Proxy Reverso

Antes de começar, certifique-se de ter dois websites em funcionamento em `example.com` e `blog.domain.com`. O primeiro site pode ou não ser um site WordPress, mas o segundo deve ser um site WordPress, pois é usado principalmente para carregar o blog do domínio principal no link do subdiretório `example.com/blog`.

Comece a configurar o Apache [abrindo o terminal do seu servidor via SSH](#) e habilitando o módulo de proxy do Apache.

```
sudo a2enmod proxy proxy_http ssl
```

A execução do comando acima muito provavelmente reiniciará o Apache para recarregar as novas diretrizes definidas.

Em seguida, edite o arquivo host virtual do seu servidor principal para criar um proxy reverso. Aqui está o código que você precisa adicionar:

```
DocumentRoot /var/www/app/public
SSLProxyEngine On      ProxyRequests off
ProxyPass /blog http://blog.domain.com
ProxyPassReverse /blog http://blog.domain.com
```

A diretiva `ProxyPass` criará um proxy reverso para os caminhos especificados, enquanto a diretiva `ProxyPassReverse` interceptará os cabeçalhos de resposta HTTP enviados através deste proxy reverso e os reescreverá para combinar com o servidor Apache.

Após salvar o arquivo, você precisa editar seu arquivo `wp-config.php` adicionando o seguinte código imediatamente antes da linha que lhe pede para parar de editar.

```
# ProxyPass Settings
```

```
# overrides the variables below to ensure that any
```

```
$_SERVER['REQUEST_URI'] = '/blog' . $_SERVER['REQUEST_URI'];
$_SERVER['SCRIPT_NAME'] = '/blog' . $_SERVER['SCRIPT_NAME'];
$_SERVER['PHP_SELF'] = '/blog' . $_SERVER['PHP_SELF'];
```

Finalmente, você precisa atualizar [banco de dados do seu site WordPress](#) para adicionar os valores de configuração para o link do subdiretório `/blog`. Você pode fazer isso executando a seguinte consulta SQL:

```
UPDATE wp_options SET option_value = 'https://www.example.com/blog' WHERE
```

Agora você deve poder visitar a URL `https://www.example.com/blog` e ter seu site WordPress hospedado em `http://blog.domain.com` sem alterar sua URL. Você pode continuar usando o WordPress como de costume para navegar, escrever, editar e gerenciar seu site.

Limitações de um Proxy Reverso

- Um proxy reverso representa um risco de segurança significativo, pois pode ler e alterar todo o tráfego que passa por ele. Se você estiver passando o [tráfego HTTPS](#) através do proxy reverso, então ele precisa decodificar e reencriptar os dados de passagem. Isto significa que ele deve possuir as chaves privadas do [certificado SSL/TLS](#). Assim, se alguma parte maliciosa puder comprometer seu proxy reverso, ela pode registrar senhas e injetar malware em seus sites.
- Se você ou seus usuários não puderem acessar seu servidor principal diretamente, a utilização de um proxy reverso pode levar a um único ponto de falha. Por exemplo, se você estiver usando um proxy reverso como frente para servir múltiplos domínios, então sua interrupção pode levar a que todos os domínios fiquem offline simultaneamente.

- [Restaurar backups](#) ou [empurrar sites de encenação ao vivo](#) em sites que carregam por meio de proxy reverso pode fazer com que o site procurado pare de carregar corretamente.

Escolhendo entre um CDN e um Proxy Reverso

Os CDNs são uma forma avançada de proxy reverso com a maior parte da configuração e manutenção cuidada por um terceiro. Eles podem fornecer [benefícios incríveis de desempenho](#) ao seu site WordPress com um pequeno esforço de sua parte.

Não apenas o conteúdo do cache CDN e servi-lo rapidamente aos usuários, mas também reduzem a carga em seus servidores de origem, reduzem os custos de largura de banda, fornecem uma camada adicional de segurança, aumentam [o SEO de seu site](#) e o ajudam a escalar melhor seu site.

Info

Kinsta CDN tem um TTFB muito baixo e melhora significativamente o desempenho de seu site. Cada conta Kinsta vem com um nível gratuito de Kinsta CDN e você pode configurá-lo facilmente em segundos. Para mais informações, você pode consultar a [comparação da Kinsta CDN com uma CDN tradicional](#).

Você notará que a maioria dos benefícios fornecidos pelos CDNs são os mesmos que os fornecidos pelos procuradores reversos. Então, você deve escolher um CDN ao invés de um proxy reverso, ou vice-versa?

Não há razão para que você tenha que se contentar com apenas um. Se você já tiver um proxy reverso instalado, você ainda verá ganhos de velocidade e desempenho com o uso de um CDN. Ambas as suas caches estão bem posicionadas, e se você tiver alguma necessidade única de tratamento de pedidos (por exemplo, [conteúdo dinâmico](#), [comércio eletrônico](#)), então você pode configurá-lo facilmente com alguns cabeçalhos personalizados passados pelo CDN ou pelo proxy reverso.

para acelerar e proteger seu site WordPress.

CLIQUE PARA TWEETAR

Resumo

O WordPress é altamente flexível. Você pode usá-lo como um [blog](#), um [site de comércio eletrônico](#), ou mesmo um [Sistema de Gerenciamento de Aprendizagem](#). Na maioria dos casos, você pode personalizar o WordPress para atender às suas exigências exclusivas.

Entretanto, às vezes você pode ter que usar um domínio separado ou um servidor secundário para hospedar um site adicional. Como discutido anteriormente, pode ser devido ao uso de diferentes pilhas de tecnologia para um [grande site empresarial](#) ou ao [lançamento de um blog WordPress](#) para um site pré-existente nãoWordPress.

Um proxy reverso pode ajudar em ambos os casos, ajudando você a obter o máximo do WordPress sem abrir mão do site principal e começar de novo.

Economize tempo, custos e otimize o desempenho do seu site com:

- Ajuda instantânea de especialistas em hospedagem do WordPress, 24/7.
- Integração do Cloudflare Enterprise.
- Alcance global com 35 centros de dados em todo o mundo.
- Otimização com nosso monitoramento integrado de desempenho de aplicativos.

Tudo isso e muito mais em um plano sem contratos de longo prazo, migrações assistidas e uma garantia de 30 dias de devolução do dinheiro. [Confira nossos planos](#) ou [entre em contato com as vendas](#) com as vendas para encontrar o plano certo para você.

Artigos relacionados