

Laboratório de Desenvolvimento de Aplicações Móveis e Distribuídas Balanceamento de carga e Proxy reverso com nginx

Prof. Cleiton Silva Tavares

Prof. Cristiano de Macedo Neto

Prof. Hugo Bastos de Paula

Proxy Reverso



Proxy reverso é um servidor posicionado na frente de um ou mais servidores web.



Intercepta solicitações dos clientes e as encaminha ao servidor de origem apropriado.



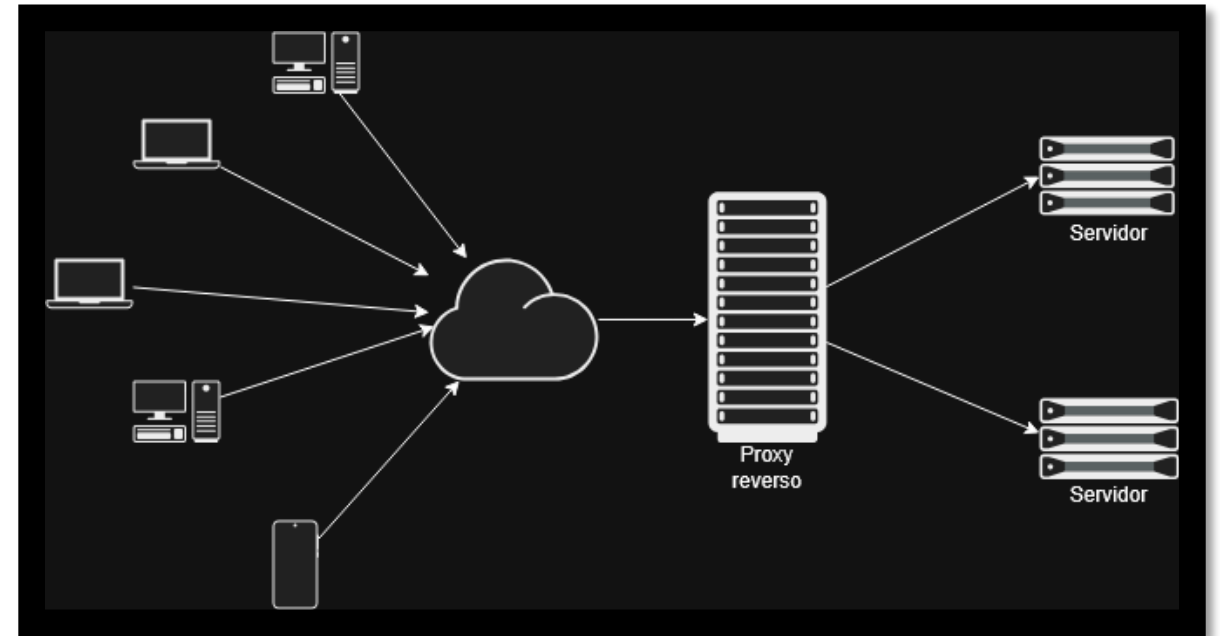
O cliente nunca se comunica diretamente com o servidor de origem, apenas com o proxy reverso.



Pode se tornar um ponto único de falha se não houver redundância.

Funcionamento do Proxy Reverso

- O cliente faz uma solicitação (por exemplo, acessar um site).
- O proxy reverso recebe a solicitação e decide para qual servidor backend encaminhá-la.
- Após processar, o servidor backend responde ao proxy reverso, que então retorna a resposta ao cliente.
- O proxy reverso atua como intermediário, ocultando o servidor de origem do cliente.



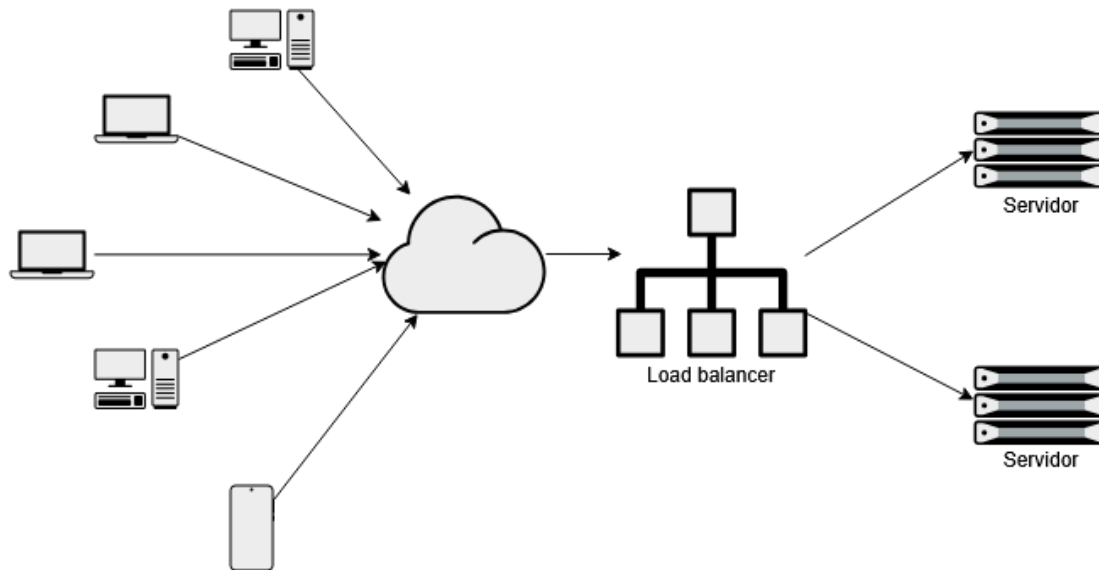
Benefícios do Proxy Reverso

- Segurança: oculta o endereço IP do servidor de origem, dificultando ataques diretos (ex: DDoS).
- Desempenho: permite cache de conteúdo, reduzindo latência e carga nos servidores.
- Balanceamento de carga: distribui solicitações entre múltiplos servidores backend.
- Gerenciamento centralizado de SSL/TLS: descarrega o processamento criptográfico dos servidores de aplicação.

O que é Balanceamento de Carga?

- Processo de distribuir o tráfego de rede entre múltiplos servidores.
- Objetivo: otimizar disponibilidade, desempenho e experiência do usuário.
- Evita sobrecarga em um único servidor e melhora a resiliência do sistema.

Funcionamento do Balanceador de Carga



- Recebe todas as solicitações de clientes.
- Decide, com base em algoritmos, para qual servidor backend encaminhar cada solicitação.
- Monitora a saúde dos servidores e evita enviar tráfego para servidores indisponíveis.

Algoritmos de Balanceamento

Round Robin/Weighted round robin:

- distribui solicitações sequencialmente entre os servidores.

Least connections:

- direciona para o servidor com menos conexões ativas.

Least response time:

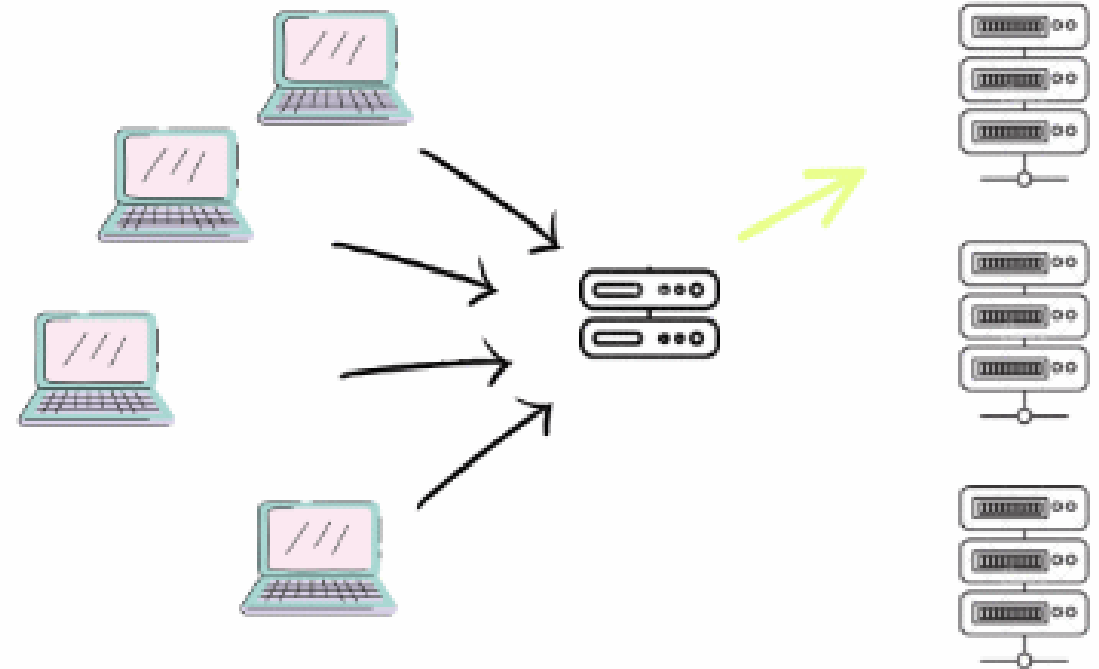
- direciona para o servidor com menor tempo de resposta.

IP Hash:

- utiliza o endereço IP do cliente para decidir o servidor.

Customizados:

- baseados em métricas de desempenho ou políticas específicas.



Balanceamento de Carga

Vantagens

Alta disponibilidade: mantém o serviço operacional mesmo com falhas em servidores individuais.

Escalabilidade: permite adicionar/remover servidores conforme a demanda.

Desempenho: reduz latência e melhora o tempo de resposta ao usuário.

Limitações

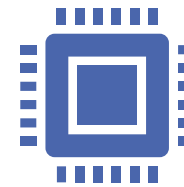
Complexidade: implementação e manutenção exigem conhecimento técnico.

Custo: soluções avançadas podem ser onerosas.

Sobrecarga: pode introduzir pequena latência adicional.

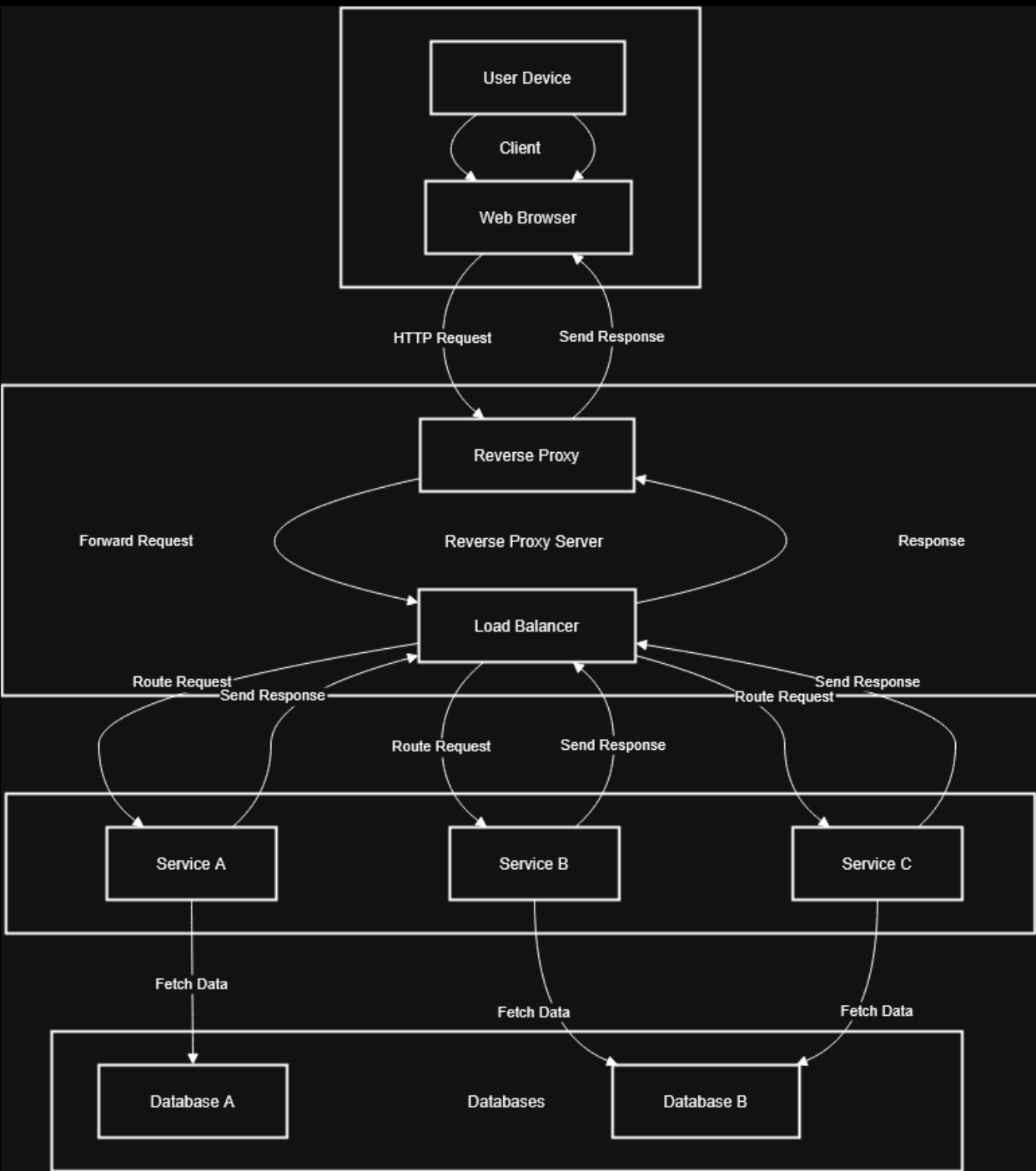
Erros de configuração: podem comprometer desempenho ou segurança.

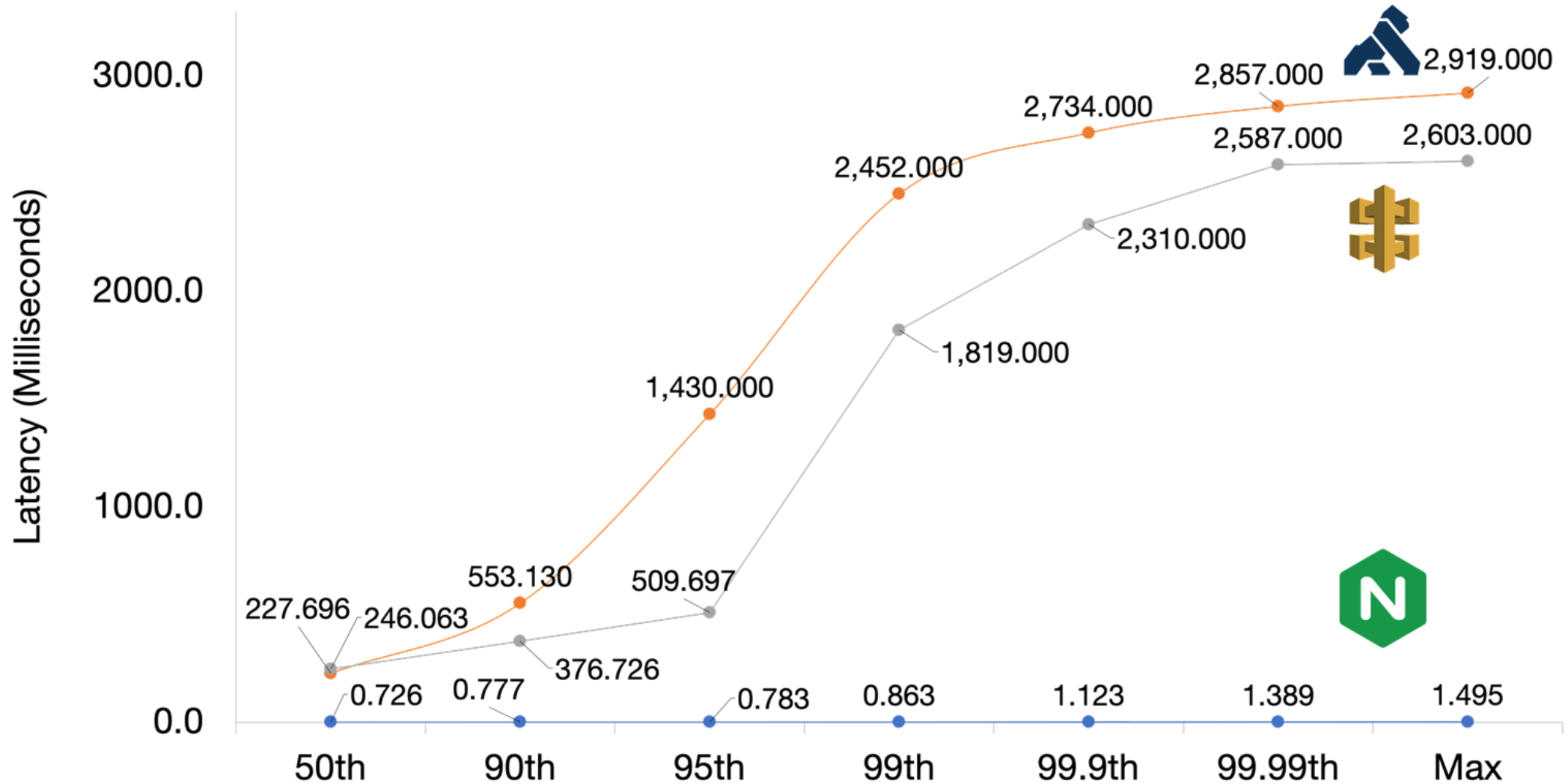
Exemplo Prático



Um site de e-commerce com milhares de acessos simultâneos utiliza um balanceador de carga para distribuir as solicitações entre diversos servidores de aplicação.

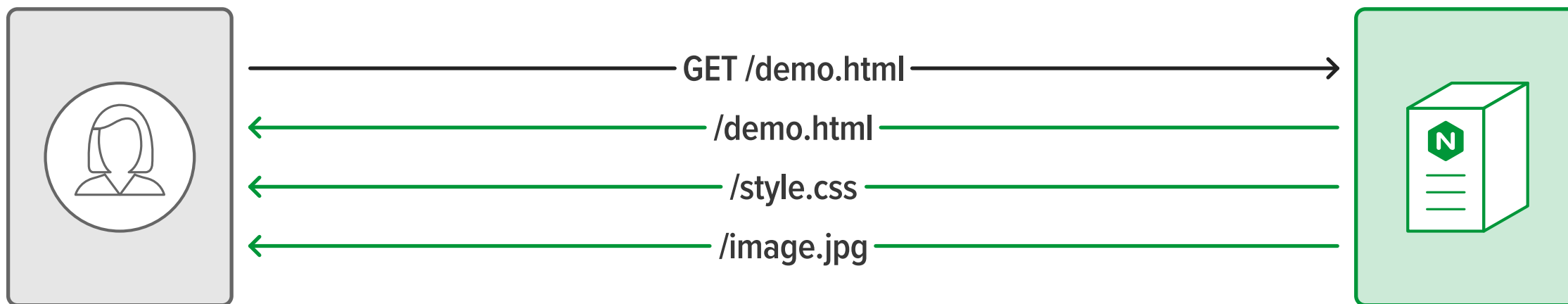
Um proxy reverso protege esses servidores, gerencia certificados SSL e faz cache de conteúdo estático, melhorando a performance e a segurança do sistema.





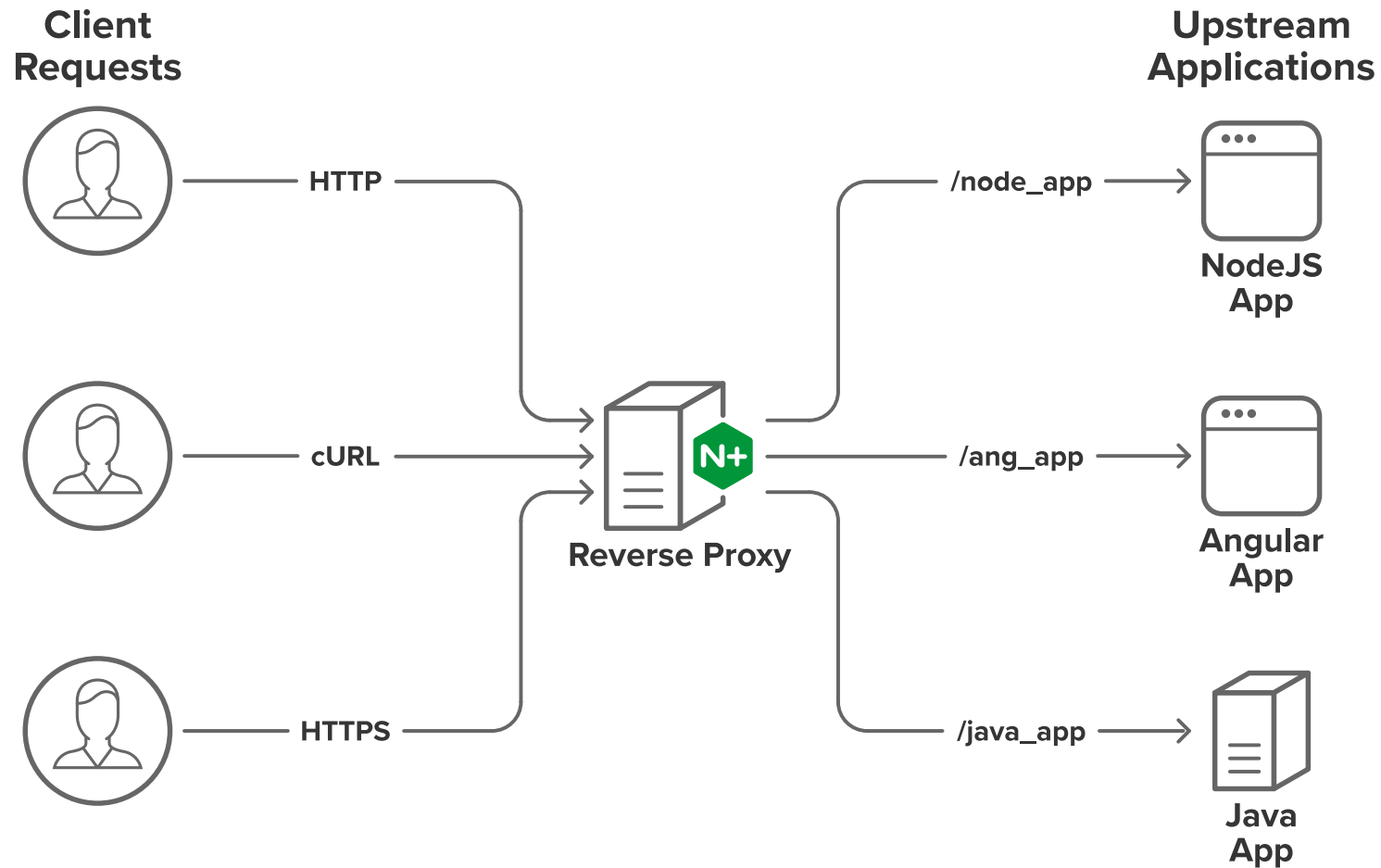
Web Server

Serve conteúdo estático e dinâmico



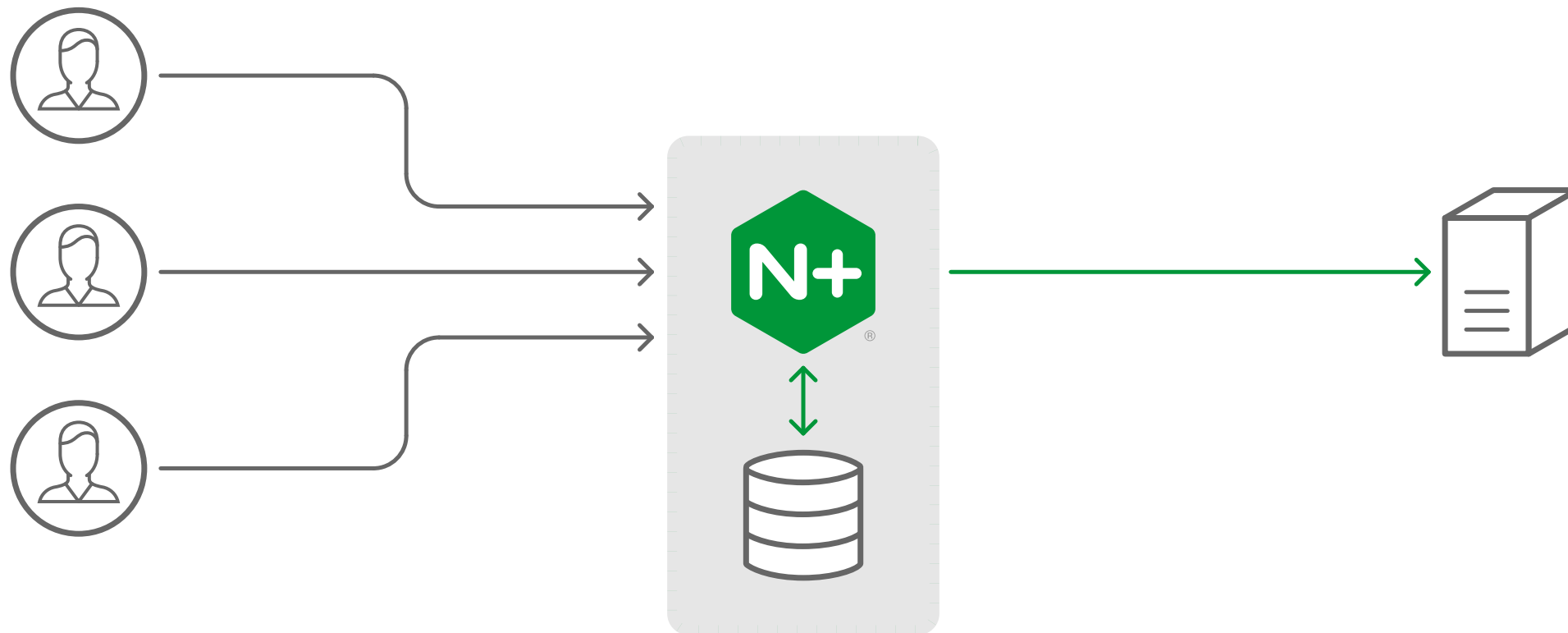
Reverse Proxy

Encaminhar solicitações para o endpoint correto



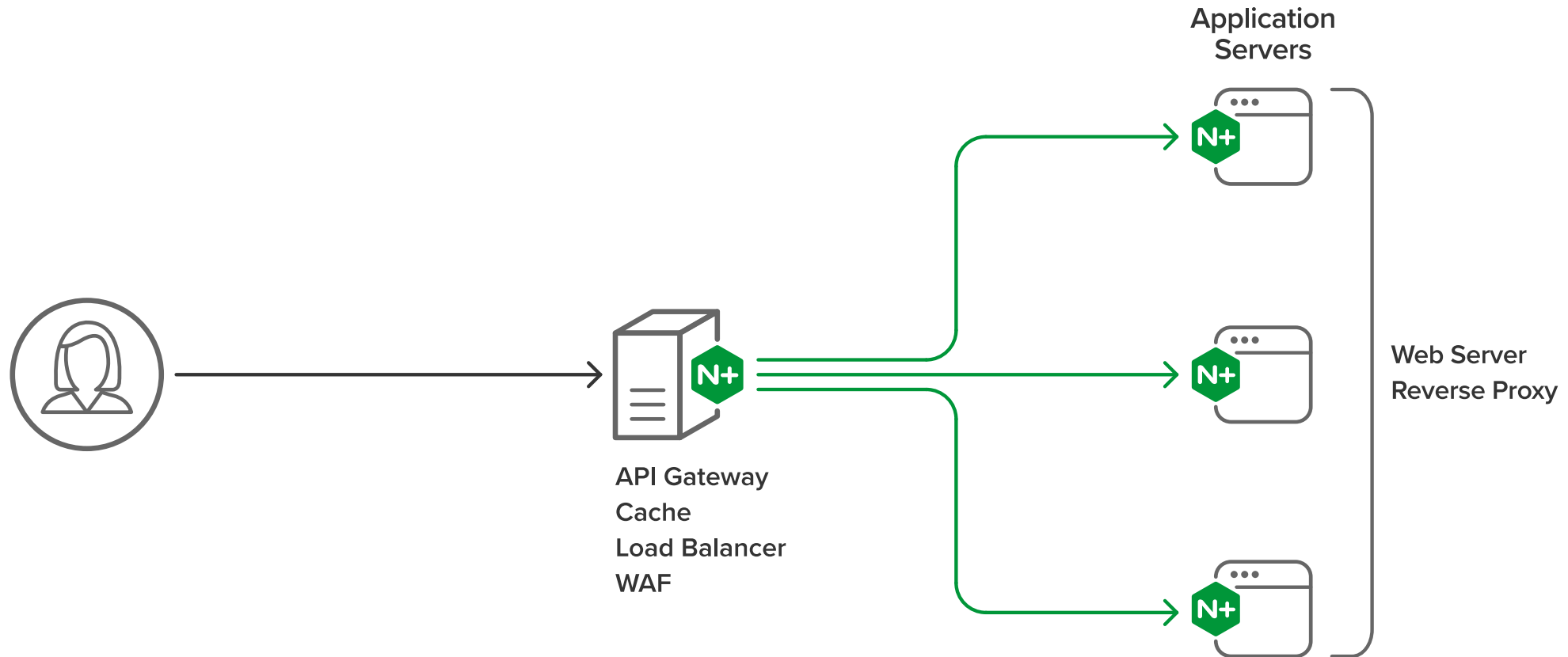
Caching

Armazene em cache todo o seu conteúdo

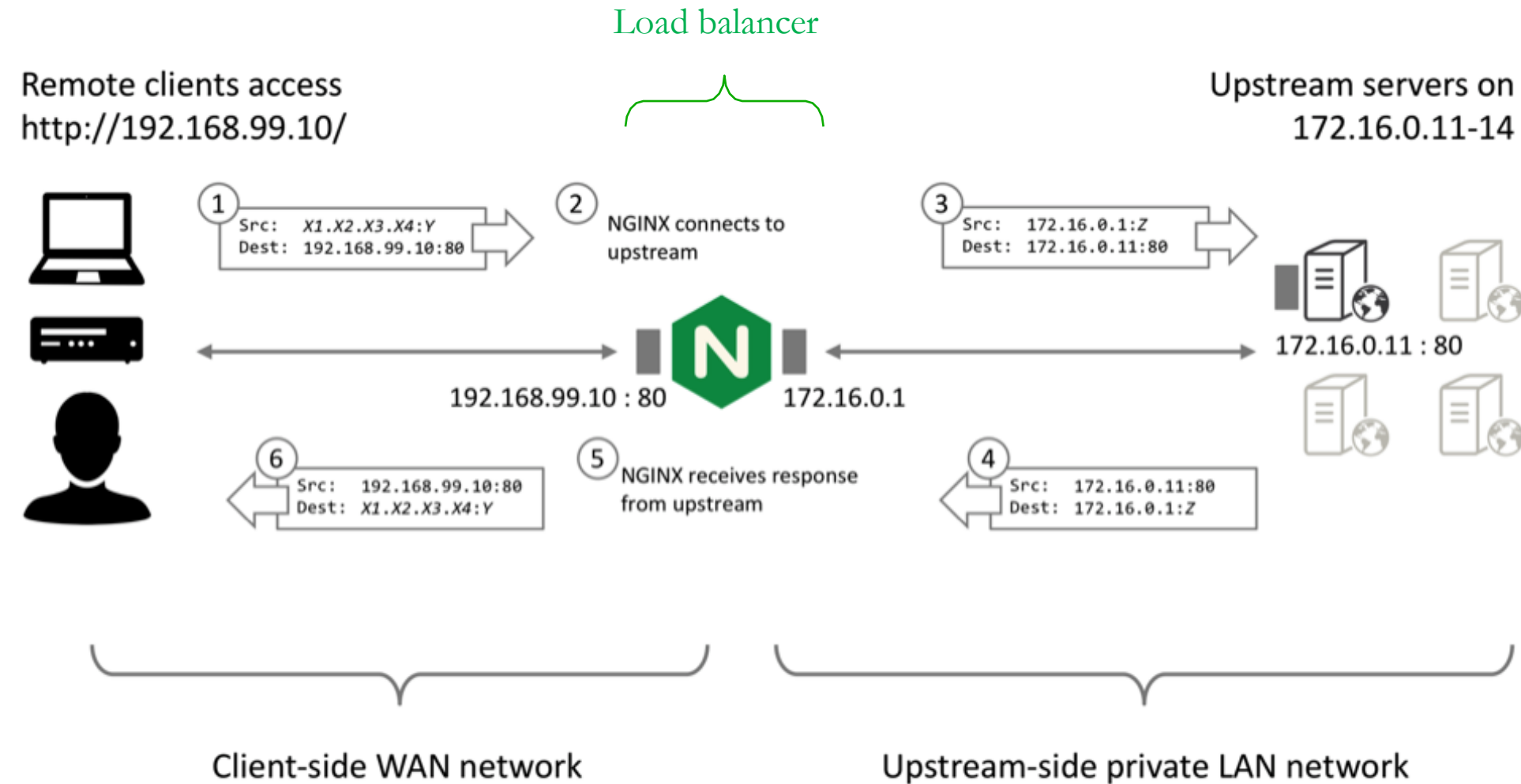


Combinando todos os três juntos

Reverse Proxy -> Content Cache -> Web Server



Nginx como um Load Balancer



1. O cliente envia solicitação ao LB.
2. LB escolhe um servidor "upstream".
3. LB retransmite a solicitação para o servidor upstream.
4. O servidor upstream envia resposta ao LB.
5. LB aceita resposta e procura o cliente associado com esta conexão.
6. LB retransmite a resposta ao cliente.