

EP26: Proxy vs reverse proxy



ALEX XU

OCT 1, 2022



340



10



Share



In this issue, we will cover:

- Why is Nginx called a “reverse” proxy?
- CAP theorem
- How Does Live Streaming Platform Work?
- CDN

Postman the API platform for building and using APIs (sponsored)



POSTMAN

Postman simplifies each step of the API lifecycle and streamlines collaboration so you can create better APIs—faster. The platform is cloud-native and includes the comprehensive suite of features enterprises are looking for, including SSO, audit, platform security, and much more.

API repository

Postman can store and manage API specifications, documentation, workflow recipes, test cases and results, metrics, and everything else related to APIs.

Workspaces

Postman workspaces are collaborative places where teams gather and solve problems. Every person in a workspace sees the same API tools, collections, and environments, which are updated in real-time.

API lifecycle

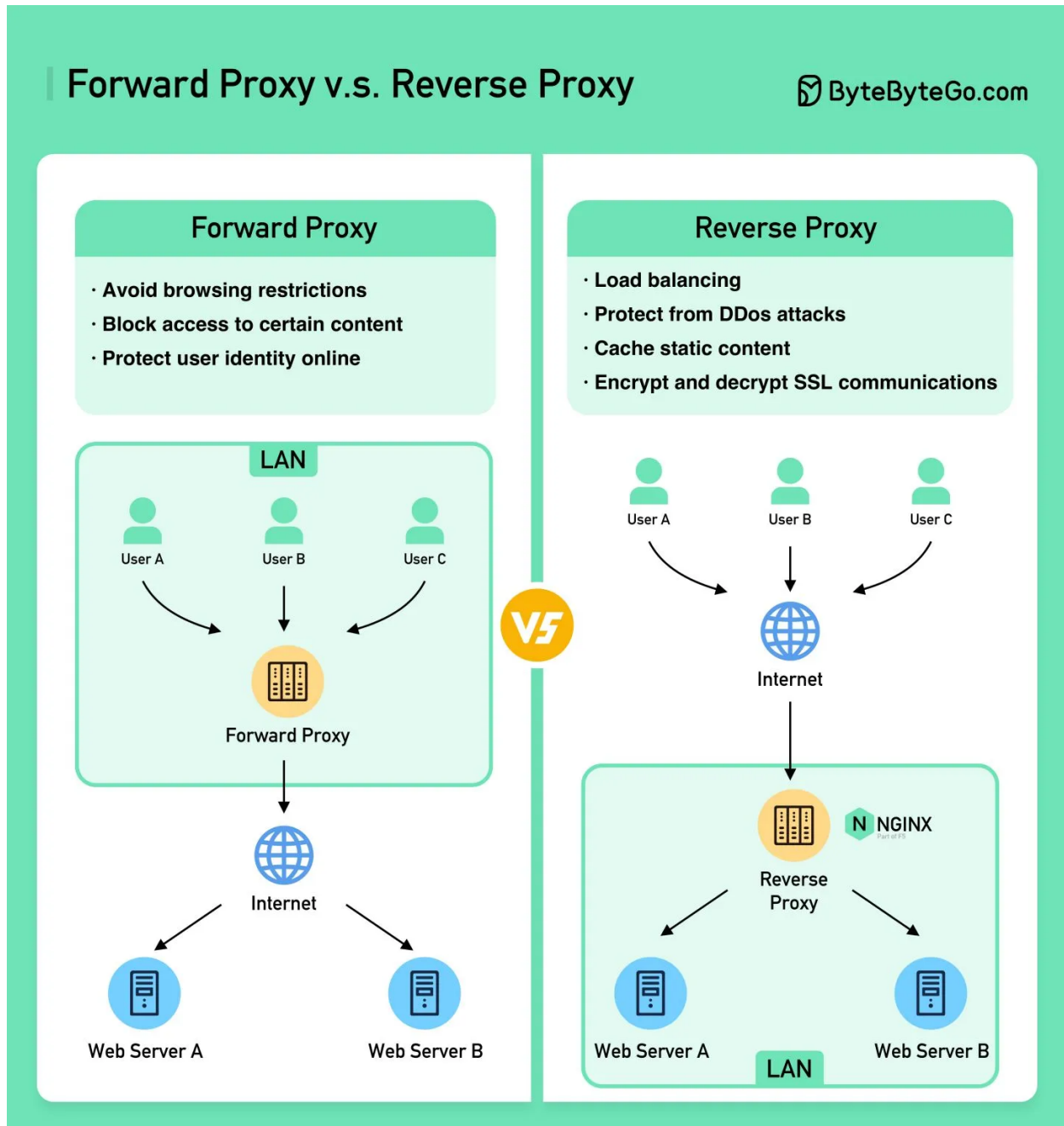
The Postman platform includes a comprehensive set of tools that help accelerate the API lifecycle—from design, testing, documentation, and mocking to the sharing and discoverability of your APIs.

Governance

Postman's full-lifecycle approach to governance lets adopters shift left in their development practices, producing better-quality APIs and fostering collaboration between developer and API design teams.

Why is Nginx called a “reverse” proxy?

The diagram below shows the differences between a **forward proxy** and a **reverse proxy**.



- A forward proxy is a server that sits between user devices and the internet.

A forward proxy is commonly used for:

1. Protect clients
2. Avoid browsing restrictions
3. Block access to certain content

- A reverse proxy is a server that accepts a request from the client, forwards the request to web servers, and returns the results to the client as if the proxy server had processed the request.

A reverse proxy is good for:

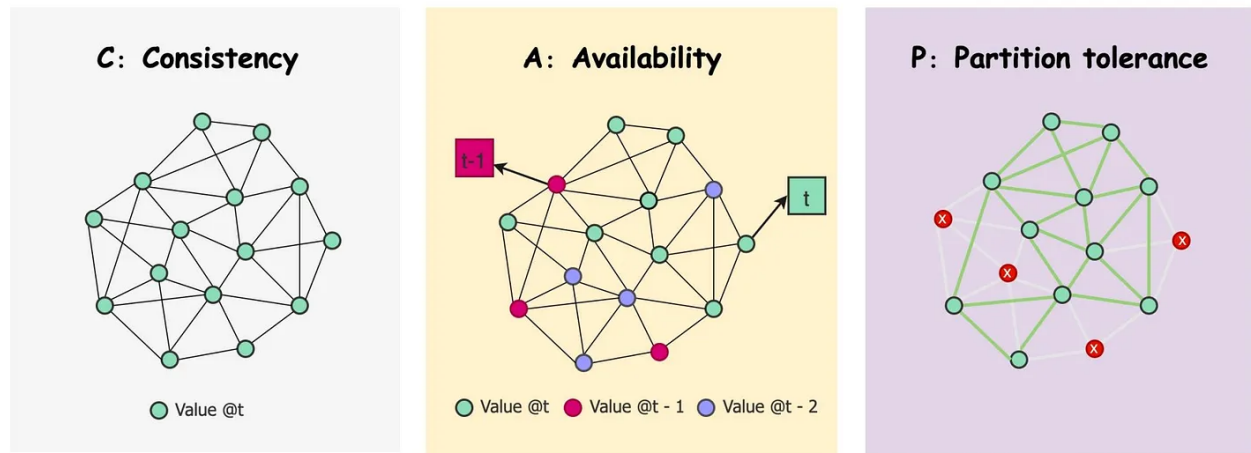
1. Protect servers
2. Load balancing
3. Cache static contents
4. Encrypt and decrypt SSL communications

CAP theorem: one of the most misunderstood terms

The CAP theorem is one of the most famous terms in computer science, but I bet different developers have different understandings. Let's examine what it is and why it can be confusing.

CAP theorem states that a distributed system can't provide more than two of these three guarantees simultaneously.

CAP Theorem



	CA systems	CP systems	AP systems
Sacrifice	No sacrifice	Availability	Consistency
Use Cases	Single-node only	Strong consistency. Banks, financial systems	Low latency. Consistency requirement is not high
Real-world examples	Single node RBMS (MySQL, Oracle)	Zookeeper, BigTable	Cassandra, CouchDB

Consistency: consistency means all clients see the same data at the same time no matter which node they connect to.

Availability: availability means any client which requests data gets a response even if some of the nodes are down.

Partition Tolerance: a partition indicates a communication break between two nodes. Partition tolerance means the system continues to operate despite network partitions.

The “2 of 3” formulation can be useful, **but this simplification could be misleading.**

1. Picking a database is not easy. Justifying our choice purely based on the CAP theorem is not enough. For example, companies don't choose Cassandra for chat applications simply because it is an AP system. There is a list of good characteristics that make Cassandra a desirable option for storing chat messages. We need to dig deeper.

2. “CAP prohibits only a tiny part of the design space: perfect availability and consistency in the presence of partitions, which are rare”. Quoted from the paper: CAP Twelve Years Later: How the “Rules” Have Changed.
3. The theorem is about 100% availability and consistency. A more realistic discussion would be the trade-offs between latency and consistency when there is no network partition. See PACELC theorem for more details.

Is the CAP theorem really useful?

I think it is still useful as it opens our minds to a set of tradeoff discussions, but it is only part of the story. We need to dig deeper when picking the right database.

How Does Live Streaming Platform Work?

(YouTube live, Twitch, TikTok Live)

How Does Live Streaming Platform Work? (YouTube live, Twitch, TikT...



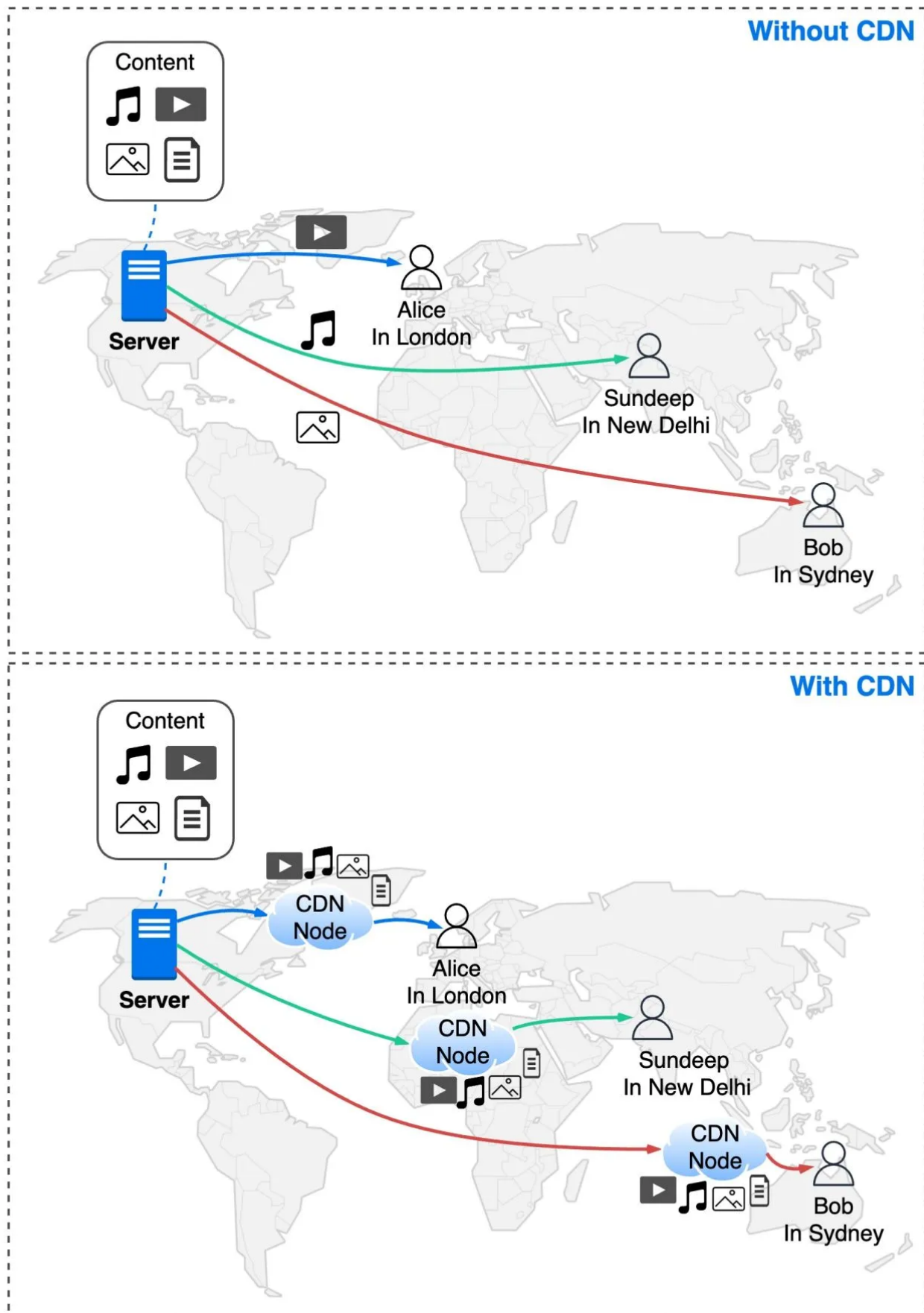
CDN

What is CDN (Content Delivery Network)? How does CDN make content delivery faster?

The diagram below shows why.

What is CDN?

 blog.bytebytego.com



A CDN refers to geographically distributed servers (also called edge servers) that provide fast delivery of **static and dynamic content**.

With CDN, users don't need to retrieve content (music, video, files, pictures, etc.) from the origin server. Instead, the content is cached at CDN nodes around the globe, and users can retrieve the content from nearby CDN nodes.

The benefits of CDN are:

1. Reducing latency
2. Reducing bandwidth
3. Improving website security, especially protecting against DDoS (Distributed Denial-of-Service) attack
4. Increasing content availability

Thanks for reading ByteByteGo Newsletter!
Subscribe for free to receive new posts and
support my work.

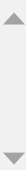


340 Likes

10 Comments



Write a comment...





David Jones Oct 1, 2022 Liked by Alex Xu

Great content. Keep it up! Im loving it

LIKE (7) REPLY SHARE



Satish May 13

Nice short post on forward and reverse proxy.

LIKE REPLY SHARE



8 more comments...