FC-VPN

Cloud-Native Elastic VPN Solution

Liu Yafei

Qin Sihan

Zhang Runze

Zhang Zeshen

July 2025

Problem Statement

Traditional Solutions	Cloud-based Solutions
- Fixed, inflexible capacity - Billing: fixed number of nodes - Poor at handling traffic spikes - Designed for large enterprises only	- Elastic scaling on demand - Instances created as needed - Pay-per-use billing - Flexible deployment for small businesses/teams - auto-scaling

Implementation Principle

(Internal technical discussion only)

- **VPN prototype:** Proxy forwards traffic (HTTPS over SOCKS5)

- **Elastic scaling:** Managed by Kubernetes (k8s)

- **User scaling:** Each proxy handles a fixed number of users; k8s auto-scales pods as amount of users changes

Flowchart

(Internal technical discussion only)

```
A[User] --> B[Cloud VPN Proxy]
B --> C[k8s Operator (Manages Pods)]
C --> D[Each Pod Connects to Target Server]
[draw on the broad, maybe]
```

Assumptions

(Internal technical discussion only)

- Pool of public IPs; each pod can be assigned a public IP to connect the Internet

- Target users (two directions):
  -1. small businesses/teams (NAT traverselNetwork Address Translation traversal)
  -2. Bypassing the firewall

- Complete VPN component is available (instead of a prototype)

Demo

Client (Frontend):

- Multiple clients, each with a personal user profile (StatefulSet)
- Display outgoing public IP (optional, or as an element to show different IP by proxy)
- data display

Demo

Proxy Monitor:

- Real-time traffic monitoring
- Dynamic pod/node/objects count&details (via k8s dashboard, e.g. AWS)

Discussion Questions

- The key technical challenges?

- Dividing ofthe work?

- Regarding "direction": good to see two or just focus on one?