Internal Load Balancing in 5 mins

Deliver scalable and resilient internal-only services on GCP
# Google Cloud Load Balancing

<table>
<thead>
<tr>
<th>Global Load Balancing</th>
<th>Regional Load Balancing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HTTP(S) Load Balancing</strong></td>
<td><strong>Internal TCP/UDP Load Balancing</strong></td>
</tr>
<tr>
<td><strong>SSL proxy</strong></td>
<td><strong>Network TCP/UDP Load Balancing</strong></td>
</tr>
</tbody>
</table>

**Request Routing**
- Hash-based LB algorithm
- Capacity-based LB algorithm

**Health Checks**
- Connection Draining
- Autoscaling

**Affinity**
- Logging
- Monitoring

**Cloud CDN**
- Cross-region failover
Internal Load Balancing

- Internal (RFC 1918) Load Balancing
- Client IP preserved
- Health checks (TCP, HTTP, HTTPS)
- Autoscaling (no prewarming)
- Session Affinity

REST API  gcloud CLI  Console
Internal Load Balancing Example

Network: shopnet

Client instances

Zone: us-central1-a
192.168.1.5
192.168.1.1

Subnet 1

Internal LB IP 10.240.0.200

Subnet 2

10.240.0.1
10.240.0.2
10.240.0.3

Zone: us-central1-b

Zone: us-central1-c

172.16.1.1

Subnet 3

Google Cloud Platform US Central region

Logical representation of the ILB, there is no LB instance between client and backend instances

Backends running shopping cart application
Under the hood: Software-defined load balancing

Load Balancers can become choke points

There is no load balancer in the path between your client and backend instances

Andromeda

Internal Load Balancing is delivered using Andromeda, GCP’s software-defined network virtualization stack

Typical Instance-based Internal Load Balancing

Software-defined Internal Load Balancing on GCP
Configuring Internal Load Balancing (console)
Internal Load Balancing Configuration

- Forwarding Rule
- Regional Backend Service
- Backend Configuration
  - Internal LB IP, port configuration

Region X

- Subnet A
  - Instance Group 1
    - Instance 1
    - Instance 2
  - Instance Group 2
    - Instance 3
    - Instance 4

- Subnet B
  - Instance Group 3
    - Instance 5
    - Instance 6
1. Click “create load balancer” to get started

Click Create
2. Select the type of Internal Load Balancer- either TCP or UDP

Click here for TCP ILB
3. Specify you want to load balance traffic between your instances (internal)

Specify you want internal load balancing
4. Configure backends in your region of choice

- Configure backends
- Remember to open up firewall for ILB health checks
5. Configure your RFC 1918 Internal LB IP (specify or let ILB auto-allocate)
6. Click “create” and your ILB is ready to distribute traffic!
Internal Load Balancing Use Cases
ILB Use Case 1: Scaling and HA for internal (micro)services

Load balance across multiple instances of your internal service

Google Cloud Platform Application composed of ILB-scaled services
ILB Use Case 2: Multi-tier apps

- **Web Tier with external load balancing**
- **Internal Tier with Internal Load Balancing**
- **User in San Francisco**
  - External Load Balancing:
    - Global: HTTP(S) LB, SSL Proxy
    - Regional: Network TCP/UDP LB
  - IP: 100.1.1.1, Port: 80
- **User in Iowa**
  - Internal Tier with Internal Load Balancing
  - IP: 10.10.10.10, Port: 80
- **User in Singapore**
  - External Load Balancing:
    - Global: HTTP(S) LB, SSL Proxy
    - Regional: Network TCP/UDP LB
  - IP: 10.20.1.1, Port: 80

- **Database Tier**

- **Web Tier**
- **Internal Tier**
- **Database Tier**

- **User in San Francisco**
- **User in Iowa**
- **User in Singapore**
ILB Use Case 3: Scale-out and HA for virtual appliances