KPLABS Course

HashiCorp Certified: Consul Associate

Getting Started with Consul

ISSUED BY
Zeal

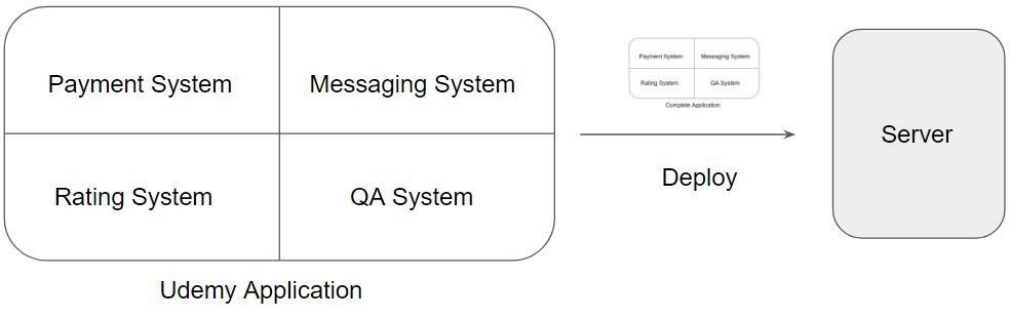
REPRESENTATIVE
instructors@kplabs.in

Module 1: Introduction to Consul

1.1 Monolithic Approach

In the Monolithic approach, there is a single application that gets deployed.

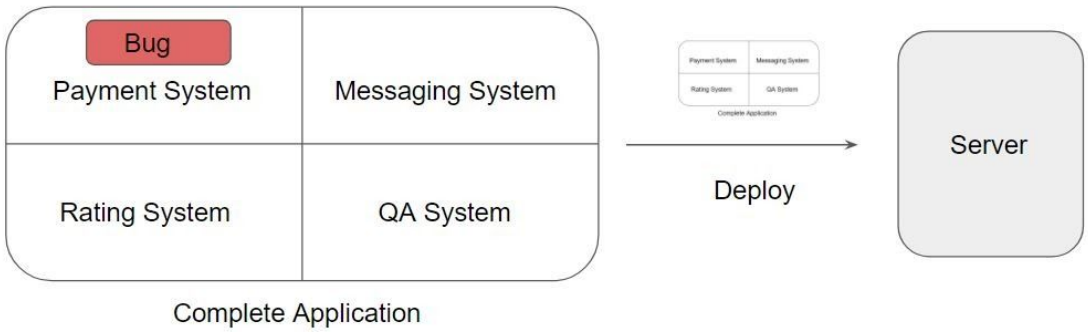
This single application in-turn has multiple sub-components.



1.2 Challenges with Monolithic Approach

If there is a bug in any one of the sub-components, we have to re-deploy the entire application.

We cannot just fix the bug in the “Payment system” and deploy it.



1.3 Disadvantage of Monolithic Based Architecture

The application and codebase size increases over time and it becomes difficult to manage.

The whole application needs to be deployed as a package even for small changes.

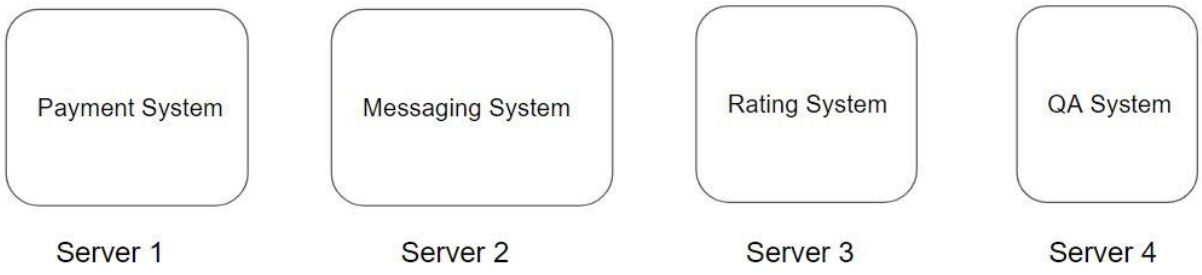
Even if a single part of the application is receiving huge traffic, we need to deploy the whole application.

A single bug can affect the whole system.

1.4 Microservice-based Architecture

In this architecture, “complete application” is divided into set of individual services.

Each service can independently be deployed as well as scaled.



1.5 Overview of Consul

Consul is a service mesh solution providing a full-featured control plane with service discovery, configuration, and segmentation functionality

1.6 Feature 1 - Service Discovery

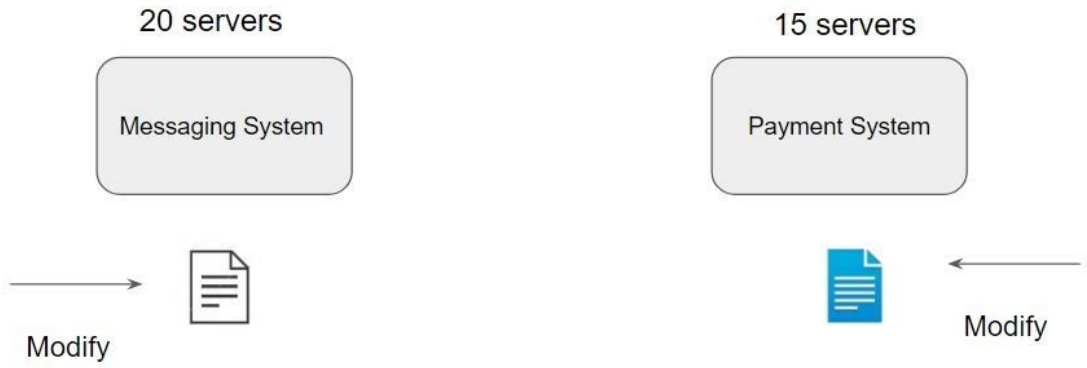
Service Discovery is the way in which microservices can locate each other on the network.

1.7 Feature 2 - Health Checks

Consul with health check functionality will only keep the healthy instances related data within its database.

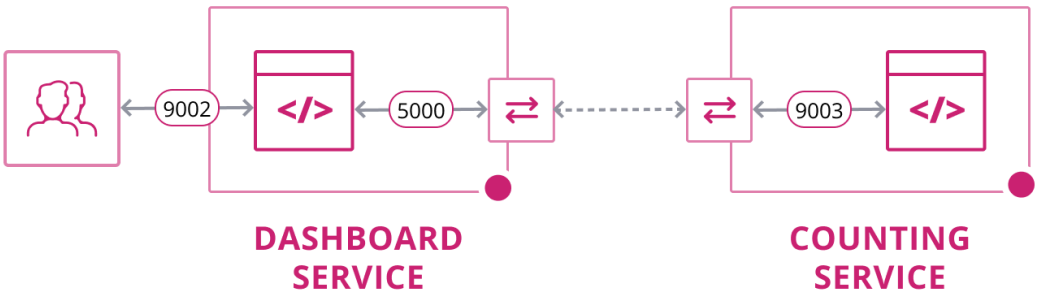
1.8 Feature 3 - KV Store

Key-Value store is generally used for storing the service configuration and other meta-data.



1.9 Feature 4 - Secure Service Communication

Sidecar proxies can be used to automatically establish TLS connections for inbound and outbound connections.



Module 2: Our Lab Architecture

2.1 Overview of Installation Process

Consul installation is very simple.

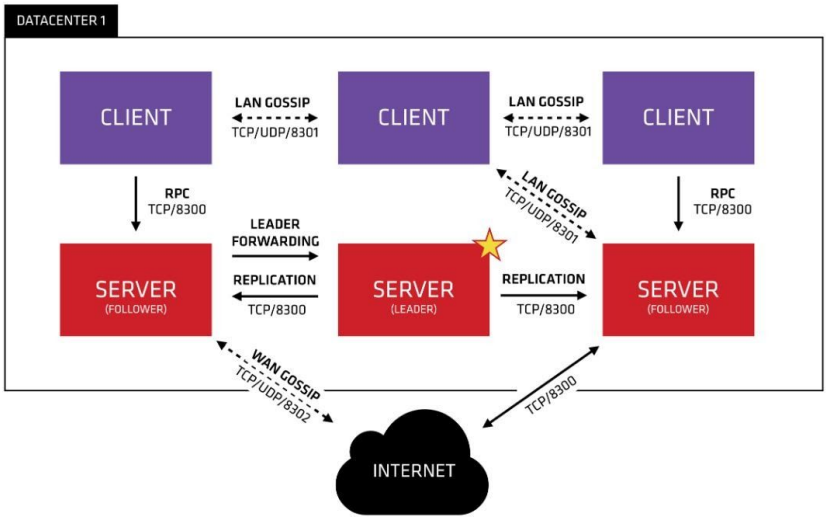
You have a single binary file, download and use it.

There are many amazing features that we need to practically try it out.

Some of these includes:

- Service Discovery
- Health Checks
- Service Mesh
- Secure Service Communication (Mutual TLS)

For these, we will need multiple instances of consul running.



2.2 Selecting a Provider

For our labs, we will be making use of the following configuration:

Operating System	Cloud Provider
Linux (CentOS 8)	Digital Ocean

You can use any Cloud provider as you like.

2.3 Why Digital Ocean for our Labs?

Following ar some of the benefits of the Digital Ocean:

Reasonably priced Servers ($5/month) - pay per hour

Good Amount of Credits for New Users (Referral) - $100

Keep it simple approach.

Module 3: Dev Agent Mode

3.1 Learning Approach

Whenever you learn any technology, you start from the very basics and keep it simple initially.

Once you are comfortable with the basics, you go more in-depth about aspects like security, high-availability, and others.

3.2 Overview of Consul Dev mode

The Dev agent mode in Consul is useful for local development, testing, and exploration.

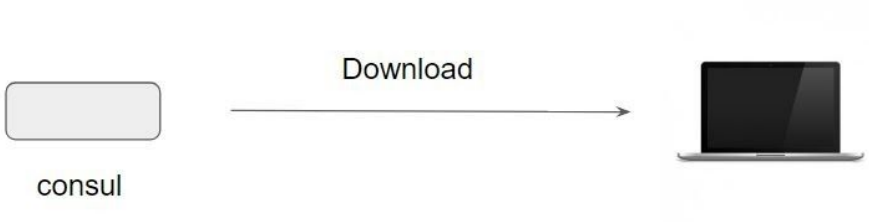
Not very secure.

In-memory mode

Module 4: Installing Consul In Linux

Consul installation is very simple.

You have a single binary file, download and use it.



Consul works on multiple platforms, primary ones include:

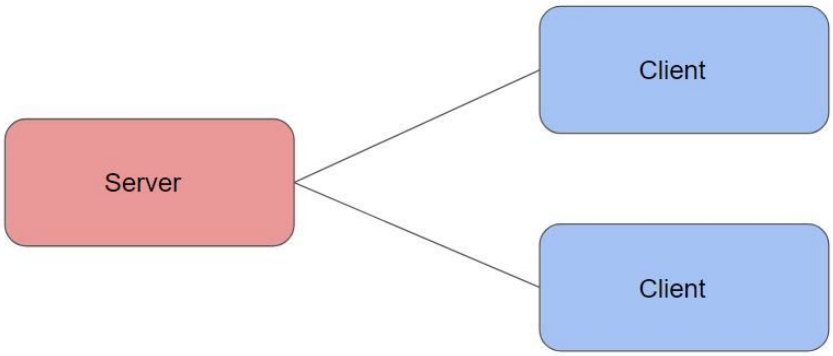
-
- Windows
- macOS
- Linux

Module 5: Consul Architecture

5.1 10,000 Feet Overview

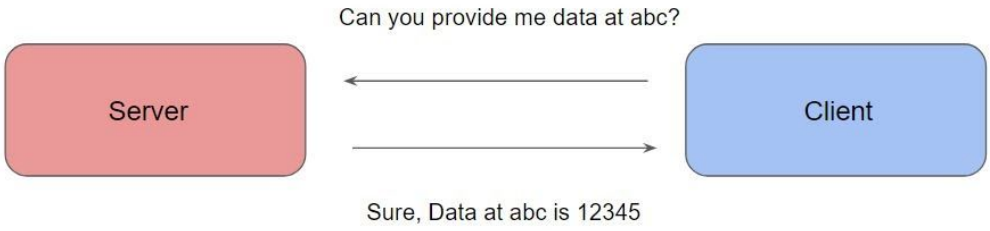
There are two primary components:

- Consul Server
- Consul Client



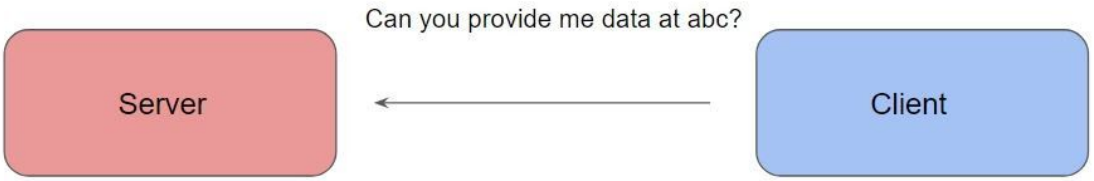
5.2 Consul Server

Consul Server is primarily responsible for maintaining the cluster state, as well as responding to queries received from clients.



5.3 Overview of Consul Client

The client is primarily responsible for making requests to the server and are also used for performing health check.



5.4 Overview of Consul Agent

An agent is a long-running daemon on every member of the Consul cluster.

The agent is able to run in either client or server mode.

Since all nodes must be running an agent, it is simpler to refer to the node as being either a client or server

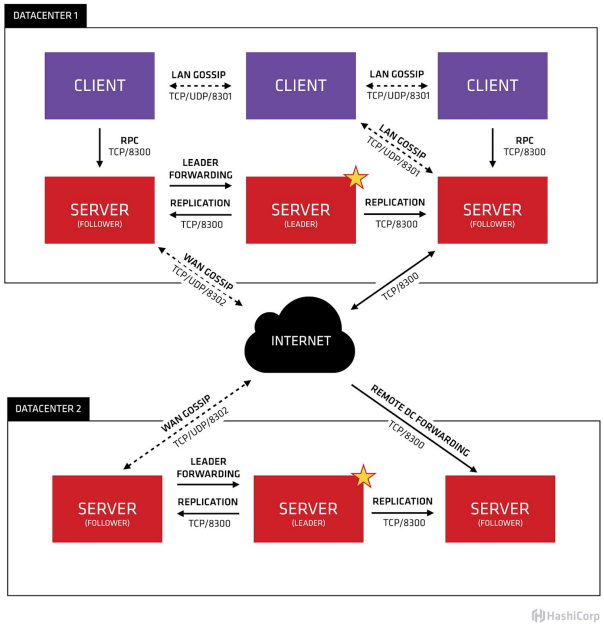
```
[root@consul-01 ~]# consul agent -dev
==> Starting Consul agent...
           Version: '1.8.5'
           Node ID: '71571abc-9474-9a9b-397c-1ff5acdfca72'
         Node name: 'consul-01'
        Datacenter: 'dc1' (Segment: '<all>')
            Server: true (Bootstrap: false)
       Client Addr: [127.0.0.1] (HTTP: 8500, HTTPS: -1, gRPC: 8502, DNS: 8600)
      Cluster Addr: 127.0.0.1 (LAN: 8301, WAN: 8302)
           Encrypt: Gossip: false, TLS-Outgoing: false, TLS-Incoming: false, Auto-Encrypt-TLS: false

==> Log data will now stream in as it occurs:
```

5.5 Consul Datacenter

A datacenter is a networking environment that is private, low latency, and high bandwidth.

This excludes communication that would traverse the public internet.

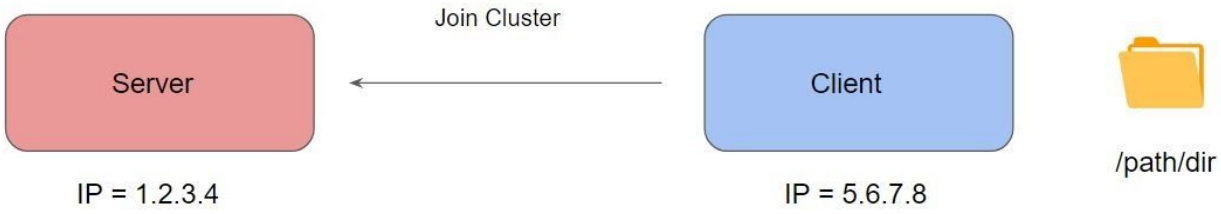
Multiple Availability Zone within a single AWS region would be considered part of a single datacenter.



Module 6: Joining Consul Clients

To join clients to consul cluster, the following command needs to be used:

consul agent -join 1.2.3.4 -bind 5.6.7.8 -data-dir /path/dir

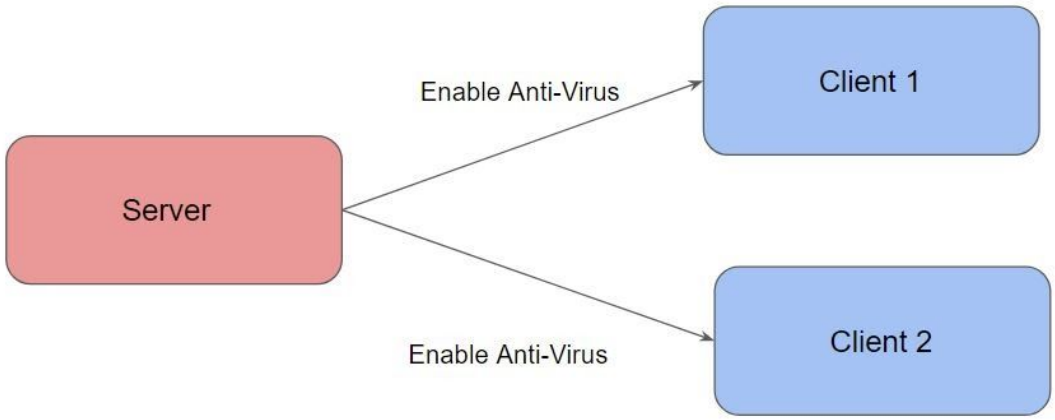


The following table shows some of the important flags along with their description.

Flag	Description
-join 1.2.3.4	Address of another agent to join upon starting up.
-bind 5.6.7.8	This is an IP address that should be reachable by all other nodes in the cluster
-data-dir /path/dir	This is required for all agents. The directory should be durable across reboots. All used for storing cluster state.

Module 7: Remote Execution Functionality

Remote Execution can be used to run a certain set of commands to perform the desired action.



The feature of remote execution is achieved with the consul exec command.

consul exec ping google.com

```
[root@consul-01 ~]# consul exec ping -c1 google.com
    consul-02: PING google.com (216.58.196.174) 56(84) bytes of data.
    consul-02: 64 bytes from maa03s31-in-f14.1e100.net (216.58.196.174):
    consul-02:
    consul-02: --- google.com ping statistics ---
    consul-02: 1 packets transmitted, 1 received, 0% packet loss, time 0m
    consul-02: rtt min/avg/max/mdev = 9.753/9.753/9.753/0.000 ms
    consul-02:
==> consul-02: finished with exit code 0
1 / 1 node(s) completed / acknowledged
```

Important Note:

Remote Execution is disabled by default.

You will need to explicitly enable it at the node level to make use of it.

Following is the command to Enable Remote Execution:

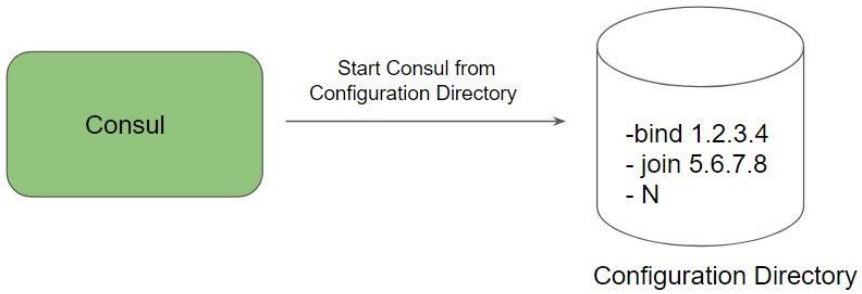
consul agent -hcl 'disable_remote_exec=false'

Module 8: Configuration Directory

8.1 Overview of Configuration Directory:

Running Consul from CLI with multiple command-line options is not a very scalable and efficient way of doing things.

consul agent -join 134.209.155.89 -bind 165.22.222.190 -data-dir /root/consul -hcl 'disable_remote_exec=false'



8.2 Using a Configuration Directory Approach

The agent has various configuration options that can be specified via the command-line or via configuration files.

Configuration can either be in JSON or HCL format.

```
[root@consul-02 consul-config]# consul agent -config-dir=/root/consul-config/
==> Starting Consul agent...
           Version: '1.8.5'
           Node ID: '952d3940-f447-3215-be8e-f20bf9b5e275'
         Node name: 'consul-02'
        Datacenter: 'dc1' (Segment: '')
            Server: false (Bootstrap: false)
       Client Addr: [127.0.0.1] (HTTP: 8500, HTTPS: -1, gRPC: -1, DNS: 8600)
      Cluster Addr: 165.22.222.190 (LAN: 8301, WAN: 8302)
           Encrypt: Gossip: false, TLS-Outgoing: false, TLS-Incoming: false, Auto-Encrypt-TLS: false

==> Log data will now stream in as it occurs:
```

8.3 Important Note - Configuration Directory

The CLI and Configuration File options are not always named the same.

CLI Command	Configuration File
join	start_join
bind	bind_addr
data-dir	data_dir

Module 9: Leave Behavior for Agents

Whenever we go somewhere, we generally inform parents/spouse about it.

If you do not inform, be prepared to sit on the couch for an hour or two listening to what could have been improved.

There are two primary leave behavior for an agent in Consul:

Options	Descriptions
Graceful Exit	It is used to ensure other nodes see the agent as "left" instead of "failed" When gracefully exiting, the agent first notifies the cluster it intends to leave the cluster.
Force Removal	When server simply fails (power/network cut). Datacenter will detect the failure and replication will continuously retry.

To gracefully halt an agent, send the process an interrupt signal (usually Ctrl-C from a terminal or running killall -s 2 consul

Forceful removal can be achieved by sending SIGKILL signal

killall -s 9 consul

Module 10: Consul Server Mode

Running consul in server mode (non-dev) allows customers to have flexibility related to the option sets that can be used.

Important Flags	Description
server	Providing this flag specifies that you want the agent to start in server mode.
-bootstrap-expect	This tells the Consul server how many servers the datacenter should have in total
-node	Each node in a datacenter must have a unique name. By default, Consul uses the hostname of the machine, but you'll manually override it, and set it to agent-one.
-bind and -data-dir	Address Agent will Listen & Storing state data.
config-dir	This flag tells consul where to look for its configuration Standard location is /etc/consul.d

Module 11: Systemd and Consul

11.1 Overview of Challenges

As of now, we have been deploying Consul either via CLI flags or via Configuration Directory.

Although this is a good approach for an initial start, but for production, it is not an ideal approach.

```
[root@consul-02 ~]# consul agent -join 134.209.155.89 -bind 165.22.222.190 -data-dir /root/consul
==> Starting Consul agent...
           Version: '1.8.5'
           Node ID: '952d3940-f447-3215-be8e-f20bf9b5e275'
         Node name: 'consul-02'
        Datacenter: 'dc1' (Segment: '')
            Server: false (Bootstrap: false)
       Client Addr: [127.0.0.1] (HTTP: 8500, HTTPS: -1, gRPC: -1, DNS: 8600)
      Cluster Addr: 165.22.222.190 (LAN: 8301, WAN: 8302)
           Encrypt: Gossip: false, TLS-Outgoing: false, TLS-Incoming: false, Auto-Encrypt-TLS: false
```

11.2 Systemd Based Approach

Systemd provides a system and service manager that runs as PID 1 and starts the rest of the system

You can specify a service file with all the configuration and settings and systemd will manage the consul process accordingly.

```
[root@consul-02 ~]# cat /usr/lib/systemd/system/consul.service
[Unit]
Description="HashiCorp Consul - A service mesh solution"
Documentation=https://www.consul.io/
Requires=network-online.target
After=network-online.target
ConditionFileNotEmpty=/etc/consul.d/consul.hcl

[Service]
User=consul
Group=consul
ExecStart=/usr/bin/consul agent -config-dir=/etc/consul.d/
ExecReload=/usr/bin/consul reload
ExecStop=/usr/bin/consul leave
KillMode=process
Restart=on-failure
LimitNOFILE=65536
```