



# Getting Started with HashiCorp Consul

Created by Bryan Krausen

- schedule the frequency of health checks
- updating health checks
  - API
  - update config file & reload
- differences between ID and NAME
- health check port

- types of health checks
  - system level (host) health check
  - application (service) health check
- types of checks
  - define health checks in configuration
  - define multiple health checks
  - script check
  - HTTP check
  - tcp check
  - TTL check
  - Docker check
  - gRPC check
  - alias check

- create
  - using the API
- purpose of prepared query
- failover policies
  - static policy
  - dynamic policy
  - hybrid policy
- tags

- default namespace for PQ `<name>.query.consul`
- actions taken based on prepared query results
- order of results based on local and federated services



HashiCorp  
**Consul**

## Objective 3: Register Services and Use Service Discovery

### Objective 3a: Interpret a service registration

- command: `consul services register`
  - works with local agent
- default namespace for service `<name>.service.consul`
- default behaviour of registered services
  - when are they returned by Consul?
  - when they are not returned by Consul?
    - service health check failure
    - node health check failure
    - service was deregistered
- determine what IP and port for service
  - determine required fields
  - determine defaults

### Objective 3b: Differentiate ways to register a single service

- registered via API
- service definition as file
  - `.hcl`
  - `.json`
  - place inside `-config-dir`
  - create new definition
    - use `consul reload`
- reloadable configurations

### Objective 3c: Interpret a service configuration with health check

### Objective 3d: Check the service catalog status from the output of the DNS/API interface or Consul UI

- use DNS to determine healthy nodes
- use the API to determine healthy nodes
- using the Consul UI to determine healthy nodes

### Objective 3e: Interpret a prepared query

### Objective 3f: Use a Prepared Query