



Configuring GEO in Active/Active High Availability

- [Configuring GEO in Active/Active High Availability, on page 1](#)
- [Verifying GEO Services, on page 3](#)
- [Active/Active GEO HA Failure Injection Limitations, on page 4](#)

Configuring GEO in Active/Active High Availability

ESC Active/Active HA has three VMs as a cluster in one datacenter. The second datacenter consists of GEO-HA.

Following are the 6 pre-defined roles in GEO:

1. `init`: initial role of geo service
2. `pre_primary`
3. `primary`
4. `pre_secondary`
5. `secondary`
6. `unknown`: used when consul is not reachable

GEO can change one role to another. Transitions are defined in `esc-config.yaml`. Each transition is divided into the following 3 parts:

- `from`: current role
- `goto`: destination role
- `condition`: when GEO changes the role

Transition Conditions

When A/A HA Geo comes up, the primary datacenter has to go through `init`, `pre_primary`, and `primary` states; meanwhile, the secondary datacenter has to go through `init`, `pre-secondary`, and `secondary` states. When

all ESC VMs' health check pass on both primary and secondary datacenters, the ESC A/A HA GEO is up and running. It is ready for use.

Condition Functions

The following are all supported condition functions:

1. `return`: do nothing but return the argument
2. `and`: return true if all arguments are true
3. `or`: return true if any of the argument is true
4. `len`: return the length of the argument
5. `equals`: return true if all arguments are equal
6. `true`: return true if args can be tested for truth value in python
7. `false`: opposite to 'true'

The following are samples for GEO configurations on primary datacenter:

```

on_init: consul start
on_primary: start
on_secondary: stop
on_stop: consul stop
startup: manual
transitions:
- condition:
  return:
    and:
      - equals:
        - len: service1
        - 3
      - equals:
        - len: service2
        - 3
  rise: 3
service1:
  dc: dcl
  name: consul_agent
  passing: true
  type: service
service2:
  dc: dc2
  name: geo
  passing: true
  type: service
from: init
goto: primary
- condition:
  fall: 2
  return:
    equals:
      - len: service
      - 3
  service:
    dc: dcl
    name: consul_agent
  from: primary
  goto: secondary

```

The following are samples for GEO configurations on secondary datacenter:

```

on_init: consul start
on_primary: start
on_secondary: stop
on_stop: consul stop
startup: manual
transitions:
- condition:
  return:
    and:
      - equals:
        - len: service1
        - 3
      - equals:
        - len: service2
        - 3
    rise: 3
  service1:
    dc: dc1
    name: consul_agent
    passing: true
    type: service
  service2:
    dc: dc2
    name: geo
    passing: true
    type: service
  from: init
  goto: secondary
- condition:
  fall: 2
  return:
    equals:
      - len: service
      - 3
  service:
    dc: dc1
    name: consul_agent
  from: secondary
  goto: primary

```

Verifying GEO Services

To start Active/Active GEO-HA, run the following command:

```
escadm geo start
```

To verify the GEO status, use the following command:

```
[root@test-geo3-ha-1 esc-scripts]# escadm geo status
geo (pgid 3745) is primary
```

To verify the GEO services in current datacenter, use the following command:

```
[root@test-geo3-ha-1 esc-scripts]# escadm geo dump
{
  "37410@test-geo3-ha-2.novalocal:44793": {
    "role": "primary",
    "location": "37410@test-geo3-ha-2.novalocal:44793",
    "service": "geo"
  },
  "43391@test-geo3-ha-3.novalocal:52459": {
    "role": "primary",
    "location": "43391@test-geo3-ha-3.novalocal:52459",

```

```

        "service": "geo"
    },
    "37898@test-geo3-ha-1.novalocal:38841": {
        "role": "primary",
        "location": "37898@test-geo3-ha-1.novalocal:38841",
        "service": "geo"
    }
}

```

To verify all the GEO services in the datacenters, use the following command:

```

[root@test-geo4-ha-1 admin]# escadm geo dump --all
{
  "3745@test-geo4-ha-1.novalocal:36760": {
    "role": "primary",
    "location": "3745@test-geo4-ha-1.novalocal:36760",
    "service": "geo"
  },
  "3742@test-geo4-ha-6.novalocal:42362": {
    "role": "secondary",
    "location": "3742@test-geo4-ha-6.novalocal:42362",
    "service": "geo"
  },
  "3738@test-geo4-ha-3.novalocal:51936": {
    "role": "primary",
    "location": "3738@test-geo4-ha-3.novalocal:51936",
    "service": "geo"
  },
  "3713@test-geo4-ha-4.novalocal:37604": {
    "role": "secondary",
    "location": "3713@test-geo4-ha-4.novalocal:37604",
    "service": "geo"
  },
  "3710@test-geo4-ha-2.novalocal:44450": {
    "role": "primary",
    "location": "3710@test-geo4-ha-2.novalocal:44450",
    "service": "geo"
  },
  "3714@test-geo4-ha-5.novalocal:34875": {
    "role": "secondary",
    "location": "3714@test-geo4-ha-5.novalocal:34875",
    "service": "geo"
  }
}

```

Active/Active GEO HA Failure Injection Limitations

The ESC Active/Active GEO HA is enhanced to support a one-way GEO HA failover with a maintenance window to move it back to healthy state.

If a GEO failover happens and the ESC VMs move to the unhealthy state, use the following steps to bring ESC A/A GEO HA back to the healthy state through manual intervention:

Procedure

-
- Step 1** Resolve any issues in Datacenter1 (DC1), that may have caused the failures and enabled the GEO switch to Datacenter2 (DC2).

- Step 2** Ensure that the consul is running in at least 2 nodes in DC1 and DC2.
- Step 3** Run the `sudo escadm geo replicate -all` command on DC2 when DC1 has at least two nodes with consul in the running state.
- Step 4** Run the `sudo escadm stop` command on all the 6 ESC VMs.
- Step 5** Run the `sudo escadm geo restart` command on all the 6 ESC VMs.
-

What to do next



Note ESC does not support any operations on ESC VMs in DC1 after the GEO HA fails over to DC2.
