Edge Cloud Keystone Federation Demo

- An OPNFV & OpenStack Cooperation Project on Edge

Edge Cloud Keystone Federation Demo

➢ An OPNFV & OpenStack Cooperation Project

➢ An Edge Cloud authentication solution: OpenStack Keystone Federation

➢ Edge Cloud features: great in numbers & widely distributed

➢ Keystone Federation makes it possible for Edge:
  ➢ Centralized Management + Single Place for Authentication
  ➢ Local Operation and Management Capability
Where is Edge Cloud

Distance From UE

1km: 10us
10km: 50us
50km: 150-300us
200-300km: 1-2ms
500-1500km: 15-20ms
2000-3000km: 40-80ms

Number of Servers per DC

2018: 0
2020: 34
2025: 60

2018: 0
2020: 77
2025: 197

2018: 410
2020: 993
2025: 2109

2018: 274
2020: 1366
2025: 4636
What is Keystone Federation

Key component:
• Identity Provider: a central identification service/component, could be Keystone and other identification services
• User, a service requester
• Service Provider, local keystone or other OpenStack services

Authentication Step:
• User requires a token through command line
• After keystone federation process 1 and 2, user login to the edge cloud
Why Keystone Federation for Edge

- **Edge Cloud Features:**
  - Great in numbers
  - Widely distributed

- **Why Keystone Federation for Edge:**
  - **Centralized Management + Single Place for Authentication**
    - Identified with central Identity Provider and get access to resources everywhere
  - **Local Operation and Management Capability**
    - Edge operate autonomously if network between Edge and Central Office goes down
  - **Neat Edge Account Management**
    - One account for same services deployed in multiple edge cloud
What we did and will do

Our current work:
- Keystone to Keystone Federation: environment setup & function test
  - Installed two OpenStack with OPNFV XCI, one as Identity Provider, the other one as Service Provider
  - User at Edge Cloud asks token from keystone A in central cloud, and use the token to get access to resources in edge cloud

Next plan: edge performance test

TC1: inject packet delay. See if delay will influence the user authentication process (counting delay for authentication), and eventually the authentication fails
TC2: inject packet loss. See if packet loss will influence the user authentication process (counting delay for authentication), and eventually the authentication fails
TC3: increasing user number with a certain amount of network delay, see if the failure rate of users will change
TC4: increasing user number with a certain amount of packet loss, see if the failure rate of users will change
Summary

• Value of this project:
  • OPNFV & OpenStack Cooperation on Edge
  • OPNFV XCI can create multiple edge cloud environment
  • Verified keystone federation provides centralized O&M for multiple edge cloud

• Future Plan:
  • Continue cross-community cooperation
  • Simulate real edge environment by adding delay and packet loss between different cloud
  • Output edge-based keystone federation test cases to relevant project/s
Thank you!