Master Thesis Proposal

Constantin Gaul - TU-Berlin, 315687

May 30, 2014

1 Proposal Summary

Based on a Cloud-Federation scenario, the scope of the Master Thesis will be to connect two or more Cloud Networks in order to form a federation, having the ability to route data from one network to the other, with a network encapsulation in place.
Today's Cloud Networks are commonly set up in a SDN fashion, enabling the possibility of changing routing paths in a flexible manner, mostly using OpenFlow. This gives a Cloud-Provider the possibility to use the virtualized network as a resource to be leased to his customers, as it is already the case in CPU virtualization, where customers could rent computing time on a specified "Compute-Unit".
As of that, a Cloud federation scenario does not only have to care about Cloud discovery and connection establishment, but more importantly it also needs to take care of the interconnection of the virtualized network environment in each federated Cloud. In order to form a federated virtual network, a federation management will be proposed that is able to route traffic through different virtual network scopes, enabling tenant networks to cross the border of one single physical network.

2 Research & Development Tasks

- **Cloud Discovery**
(Technical Implementation on an Agent based Platform)
- **Cloud Federation on the Data Level**
(User- & VM Image-Data, Theoretical Approach at first, could be made technical optionally)
- **Cloud Federation on the Network-Level**
(Core Area, Technical Focus of Implementation as well as Theoretical discourse)
- ***Cloud Federation Security - Authentication and Authorization***
(Will be approached on a Theoretical Level)

3 Technologies of Interest

- **Software Defined Networking:**
  - OpenFlow
  - OpenVirteX (OpenFlow Proxy-Controller)
  - Mininet (OpenFlow enabled SDN simulation)
  - VirtualBox (As a Virtualization Hypervisor for Simulation purposes)
  - OpenStack & KVM (optional) (for a "real" Cloud environment setup)
  - Python & Java (as Mininet & OpenVirteX are using these languages)
- **Agent based Cloud Federation:**
  - Scala + Akka (Java based language + Actor model for distributed, parallel tasks)