Security Threats in Cloud Computing

Cloud computing is one of the most promising technologies in these days since it allows a user to access a potentially unlimited set of virtualized resources but, at the same time, raises new security issues that are not present in the case of an ad-hoc infrastructure. Indeed, Virtual Machines (VMs) of different tenants are often executed on the same hardware and this increases the possibilities that a virtual resource is accessed by unauthorized entities, for example due to a wrong configuration.

One solution to address these issues is to consider a group of VMs as an unique entity on which a security policy must be coherently enforced. A model that has been developed for this purpose is the Trusted Virtual Domain (TVD).

Trusted Virtual Domains (TVDs)

A TVD consists of a set of Execution Environments or EEs (e.g. Virtual Machines) and an abstract communication channel which allows EEs to securely communicate over the physical network. Through the TVD concept it is possible to enforce the following security properties:

- **Isolation**: TVD members can communicate only among themselves;
- **Confidentiality/Integrity**: communications among TVD members cannot be intercepted or modified by unauthorized entities;
- **Trust**: an EE can join a TVD only if the host which is running on satisfies the integrity properties specified in the TVD security policy.

At the first stage, the *Ontology-based Reasoner* does the enforcement of the first three security properties in the following three deployment scenarios.

Single Host. Isolation is guaranteed by connecting the VMs network interfaces to a virtual switch with VLAN support and by assigning to each TVD an unique VLAN ID.

Hosts connected through a Layer-2 network. In this case, packets are sent to the remote host by attaching the interface connected to the physical network to the virtual switch. The security properties are guaranteed by assuming the physical network that connects the hosts as trusted.

Hosts connected through a Layer-3 network. In order to allow a VM to communicate with other TVD members on a remote host, the infrastructure creates a GRE routing tunnel between the two involved hosts, whose endpoints are attached to the respective virtual switch, and data exchanged are protected through IPSec.

Modeling TVDs with Extended Libvirt

The core component of *Ontology-based Reasoner* is represented by *Extended Libvirt*, a version of Libvirt modified by TClouds to support TVDs.

With this component, it is possible to model the network resources and the desired behavior described in the previous section. In particular, as it can be seen in

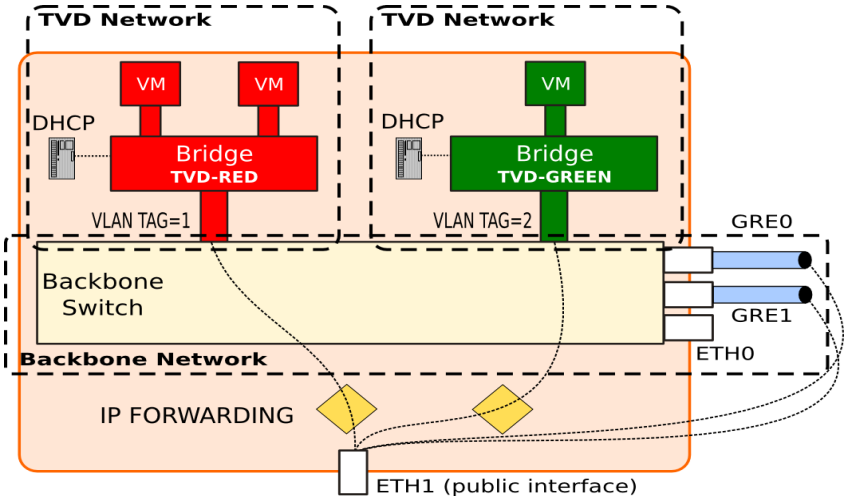

Figure 1: virtual networks modeled with Extended Libvirt

ONTOLOGY-BASED REASONER

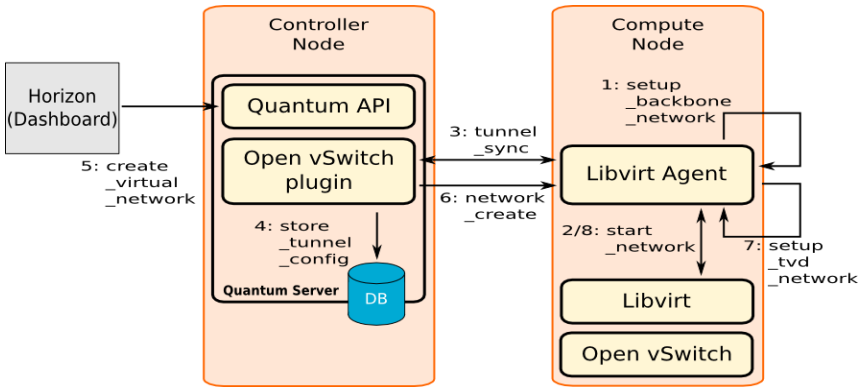

Figure 2: Ontology-based Reasoner integration into OpenStack Quantum

Figure 1, two types of virtual networks have been defined:

- **Backbone Network**: this network is represented by a virtual switch (Backbone Switch) that grants connectivity to VMs running on remote hosts through GRE tunnels or physical network interfaces;
- **TVD network**: this network configures a bridge that connects together VMs network interfaces and is connected to the backbone switch through an access port with the VLAN tag associated to the TVD. Optionally, it offers to VMs a connection to the outside.

Extended Libvirt translates XML configuration files of virtual networks and VMs into commands to the hypervisor, to Open vSwitch and to iptables.

Integration with Trustworthy OpenStack

Ontology-based Reasoner has been integrated into *Trustworthy OpenStack* as a modification of the Open vSwitch Quantum plugin. Quantum is an OpenStack component that provides networking services to VMs and is responsible to ensure isolation among virtual networks created by different tenants.

Figure 2 shows how *Ontology-based Reasoner* has been integrated into Quantum: the agent part of the Open vSwitch plugin has been replaced with *Libvirt Agent*. At startup, the latter generates the XML of a *Backbone Network* on each host with the setting of GRE tunnels to connect the local virtual switch with that of all other hosts and builds the XML of a *TVD network* each time a tenant creates a new virtual network. Then, *Libvirt Agent* sends the XML to Libvirt so that the latter configures the network. Main benefits of this solution are:

- **Easy network discovery**: network state from XML;
- **Interoperability with other cloud software stacks**: e.g Open Nebula.

Further Information

Further information about *Ontology-based Reasoner* can be found under Deliverable „D2.3.2—Components ad Architecture of Security Configuration and Privacy Management“ and „D2.4.2—Initial Component Integration, Final API Specification, and First Reference Platform“.

Disclaimer

The TClouds project has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement number ICT-257243.

TClouds at a glance

Project number:
257243

TClouds mission:

- Develop an advanced cloud infrastructure that delivers computing and storage with a new level of security, privacy, and resilience.
- Change the perceptions of cloud computing by demonstrating the prototype infrastructure in socially significant application areas.

Project start:
01.10.2010

Project duration:
3 years

Total costs:
EUR 10.536.129

EC contribution:
EUR 7.500.000

Consortium:
14 partners from 7 different countries.

Project Coordinator:
Dr. Klaus-Michael Koch
coordination@tclouds-project.eu

Technical Leader:
Dr. Christian Cachin
cca@zurich.ibm.com

Project website:
www.tclouds-project.eu