TEAS   Y. Lee
Internet-Draft   Samsung Electronics
Intended status: Informational   X. Liu
Expires: 11 January 2024   Alef Edge
                                              LM. Contreras, Ed.
                                                      Telefonica
                                                       July 2023

                        DC aware TE topology model
                    draft-llc-teas-dc-aware-topo-model-03

Abstract

   This document proposes the extension of the TE topology model for
   including information related to data center resource capabilities.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

   More and more service providers are deploying cloud computing
   facilities in order to host different kinds of services and
   applications. Such facilities can be generally referred as
   Datacenter Points of Presence (DC-PoPs). Those DCs will consist of a
   number of servers and networking elements for connecting all of them
   with the transport network. Depending on the number of servers in
   the data center, there will be distinct capabilities in terms of
   CPUs, memory and storage available for deploying and running the
   aforementioned services.

   In such distributed and interconnected DC-PoPs, both computing and
   topological information are of interest for determining the optimal
   DC where to deploy a given service or application.

   This document proposes a DC-aware extension for the topology model.

2. Datacenter information

   The relevant information for datacenter capabilities can be described
   in different ways. One potential manner is to describe resource
   capabilities such as CPU, memory, storage, etc. This can be done in
   terms of total, used and free capacity for each of the parameters of
   interest. Cloud management systems allow to obtain such kind of
   information. For instance, in the case of Kubernetes it is possible
   to retrieve information about the total and allocatable resources in
   a compute node. Alternatively, in the case of OpenStack it is
   possible to collect information about the total resources and those
   currently in use from such total.

Another form of populating the information is by describing those
resource capabilities as a bundled, usually referred as quota or
flavor. Well known cloud computing providers like Amazon Web
Services, Microsoft Azure or Google Cloud Platform follow such
schema, bundling CPU, RAM and storage units as flavors. In the case
of Amazon Web Services the bundle is known as "instance type", while
in Microsoft Azure is termed "virtual machine size" and in Google
Cloud Platform it is called "machine type".

Table 1 provides a potential example of generic predefined compute
flavors.

```
+==========+======+========+=========+
|  Flavor  | vCPU |  RAM   | Storage |
+==========+======+========+=========+
|  .tiny   |  1   | 512 MB |   1 GB  |
+----------+------+--------+---------+
|  .small  |  1   |  2 GB  |  20 GB  |
+----------+------+--------+---------+
| .medium  |  2   |  4 GB  |  40 GB  |
+----------+------+--------+---------+
|  .large  |  4   |  8 GB  |  80 GB  |
+----------+------+--------+---------+
| .2xlarge |  8   | 16 GB  |  160 GB |
+----------+------+--------+---------+
| .4xlarge |  16  | 32 GB  |  320 GB |
+----------+------+--------+---------+
| .8xlarge |  32  | 64 GB  |  640 GB |
+----------+------+--------+---------+
```

                  Table 1: Predefined Compute Flavors

Additional information to consider in both cases could refer to the
management capabilities of the compute infrastructure, such as
hypervisor details or virtualization technologies available.

Finally, all can be complemented with information related to the
networking details for reaching the aforementioned compute
capabilities (IP addressed, bandwidth, etc).

3. Model structure

```
module: ietf-dcpop-dc
   +--rw dcpop
      +--rw dc* [id]
      |  +-- comp_node* [id]
      |  |  +--rw hypervisor* [id]
      |  |  |  +--rw ram
```

```
          | | | | +--rw total?    uint32
          | | | | +--rw used?     uint32
          | | | | +--rw free?     uint32
          | | | +--rw disk
          | | | | +--rw total?    uint32
          | | | | +--rw used?     uint32
          | | | | +--rw free?     uint32
          | | | +--rw vcpu
          | | | | +--rw total?    uint16
          | | | | +--rw used?     uint16
          | | | | +--rw free?     uint16
          | | | +--rw instance*    -> /dcpop/dc/comp_node/instance/id
          | | | +--rw id           string
          | | | +--rw name?        string
          | | +--rw instance* [id]
          | | | +--rw flavor
          | | | | +--rw disk?    uint32
          | | | | +--rw ram?     uint32
          | | | | +--rw vcpus?   uint16
          | | | | +--rw bandwidth?    string
          | | | | +--rw id?       string
          | | | | +--rw name?     string
          | | | +--rw image
          | | | | +--rw checksum    string
          | | | | +--rw size        uint32
          | | | | +--rw format
          | | | | | +--rw container?   enumeration
          | | | | | +--rw disk?        enumeration
          | | | | +--rw id?         string
          | | | | +--rw name?       string
          | | | +--rw hypervisor?   -> /dcpop/dc/comp_node/hypervisor/id
          | | | +--rw port*         -> /dcpop/dc/comp_node/network/
subnetwork/port/id
          | | | +--rw project?      string
          | | | +--rw status?       enumeration
          | | | +--rw id            string
          | | | +--rw name?         string
          | | +--rw image* [id]
          | | | +--rw checksum    string
          | | | +--rw size        uint32
          | | | +--rw format
          | | | | +--rw container?   enumeration
          | | | | +--rw disk?        enumeration
          | | | +--rw id            string
          | | | +--rw name?         string
          | | +--rw flavor* [id]
          | | | +--rw disk?    uint32
          | | | +--rw ram?     uint32
```

```
      |  |  | +--rw vcpus?   uint16
```

```
         |  |  |  +--rw id        string
         |  |  |  +--rw name?     string
         |  |  +--rw dc-monitoring-param* [name]
         |  |  |  +--rw name             string
         |  |  |  +--rw value-string?    string
         |  |  +--rw network* [id]
         |  |  |  +--rw subnetwork* [id]
         |  |  |  |  +--rw port* [id]
         |  |  |  |  |  +--rw ip-address?   inet:ip-address
         |  |  |  |  |  +--rw instance?     -> /dcpop/dc/comp_node/instance/id
         |  |  |  |  |  +--rw project?      string
         |  |  |  |  |  +--rw status?       enumeration
         |  |  |  |  |  +--rw id            string
         |  |  |  |  |  +--rw name?         string
         |  |  |  |  +--rw project?   string
         |  |  |  |  +--rw status?    enumeration
         |  |  |  |  +--rw id         string
         |  |  |  |  +--rw name?      string
         |  |  |  +--rw dhcp-agent* [id]
         |  |  |  |  +--rw enabled?   boolean
         |  |  |  |  +--rw pools* [ip-address]
         |  |  |  |  |  +--rw ip-address    inet:ip-address
         |  |  |  |  +--rw project?   string
         |  |  |  |  +--rw status?    enumeration
         |  |  |  |  +--rw id         string
         |  |  |  |  +--rw name?      string
         |  |  |  +--rw project?      string
         |  |  |  +--rw status?       enumeration
         |  |  |  +--rw id            string
         |  |  |  +--rw name?         string
         |  |  |  +--rw dcpop-ref?       -> /dcpop/dcpop-id
         |  |  +--rw ap*                     -> /actn-vn:actn/ap
/access-point-list/access-point-id
         |  |  +--rw dcpop-ref?               -> /dcpop/dcpop-id
         |  |  +--rw id                  string
         |  |  +--rw name?               string
         |  +--rw name?              string
         |  +--rw id                 string
         |  +--rw ip-address?        inet:ip-address
         +--rw dcpop-id?    string
```

4. Security Considerations

The data-model in this document does not have any security
implications. The model is designed to be accessed via NETCONF
[RFC6241], thus the security considerations for the NETCONF protocol
are applicable here.

5. IANA Considerations

   This draft does not include any IANA considerations

6. References

   [RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed.,
              and A. Bierman, Ed., "Network Configuration Protocol
              (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011,
              <https://www.rfc-editor.org/info/rfc6241>.

Authors' Addresses

   Young Lee
   Samsung Electronics
   Seoul
   South Korea
   Email: younglee.tx@gmail.com

   Xufeng Liu
   Alef Edge
   Email: xufeng.liu.ietf@gmail.com

   Luis M. Contreras
   Telefonica
   Ronda de la Comunicacion, s/n
   Sur-3 building, 3rd floor
   28050 Madrid
   Spain
   Email: luismiguel.contrerasmurillo@telefonica.com
   URI: http://lmcontreras.com/