SDN Layers and Architecture Terminology

draft-haleplidis-sdnrg-layer-terminology

IETF – 88 Vancouver

Evangelos Haleplidis (ehalep@ece.upatras.gr)
Spyros Denazis (sdena@upatras.gr)
Kostas Pentikousis (k.pentikousis@eict.de)
Jamal Hadi Salim (hadi@mojatatu.com)
Odysseas Koufopavlou (odysseas@ece.upatras.gr)
Draft Motivation

• Several frameworks are self-defined as ‘SDN’, and
  ◦ most, if not all, have defined their own SDN layer model accompanied by distinct terminology
  ◦ earlier work at the IETF fits well into the SDN sphere but uses different terminology

• What does “SDN” encompass exactly?
  ◦ Which “layers” are key?
  ◦ What are the interactions between the layers?
Draft Goals

- Create a reference document for SDNRG discussions
  - address “Survey of SDN approaches and Taxonomies” in the RG Charter for Potential Work Items
  - in contrast with an academic survey which expresses one’s pov, this is a document based on RG review and consensus

- Agreement on common terms as we move forward in SDN research
  - Create a reference layered model for SDN
  - Map current frameworks on SDN model
Draft Non-goals

- No new specification
  - Instead we focus on documenting what has already been specified at IETF and other relevant bodies

- No new standard
  - This is an informational draft
SDN

- Functionality Separation:
  - Model
  - Separate via interface
  - Service APIs northbound
- Applied to networking
- Why stop at the forwarding plane?
- What about the management plane?
- Which protocols fit in this model and how?
Reference Layer Model

Application Plane

Service Abstraction Layer (SAL)

Control Plane

Control Abstraction Layer (CAL)

Service Interface

Management Plane

Management Abstraction Layer (MAL)

Device Abstraction Layer (DAL)

Forwarding Plane

Network Device

Operational Plane

CP Southbound Interface

MP Southbound Interface
Current Draft

- Maps few frameworks to reference layer model as proof-of-concept
  - ForCES
  - NETCONF
  - I2RS
  - OpenFlow
Reference Layer Model Mapping

- **Service**
- **Application Plane**
- **App**

**Service Abstraction Layer (SAL)**

**Control Abstraction Layer (CAL)**
- **I2RS Agent**

**Control Plane**

**Forwarding Plane**
- **ForCES model**

**Network Device**

**Operational Plane**
- **YANG model**

- **RestAPIs, CORBA, NETCONF/RESTCONF, I2RS**
- **ForCES protocol**
- **OpenFlow**
- **NETCONF**

**Management Plane**
- **Management Abstraction Layer (MAL)**

- **ForCES model**
- **ForCES protocol**
Moving Forward

• Comments / Feedback
  ◦ Thanks to David Meyer, Salvatore Loreto and Sudhir Modali
  ◦ Looking forward to your comments!

• Care to suggest text and join the effort?
  ◦ map your framework and provide us with details/comments

• Request this document to be adopted as an RG document