PPPoE Server impl. using DPDK

KOM lab – SoSe - 2016

TECHNISCHE UNIVERSITÄT DARMSTADT

Govind Singh (govind.singh@stud.tu-darmstadt.de)
Puneet Arora (puneet.arora@stud.tu-darmstadt.de)
Sooraj Mandotti (sooraj.mandotti@stud.tu-darmstadt.de)

MSc. (Informatik) 3rd Semester

Index

TECHNISCHE UNIVERSITÄT DARMSTADT

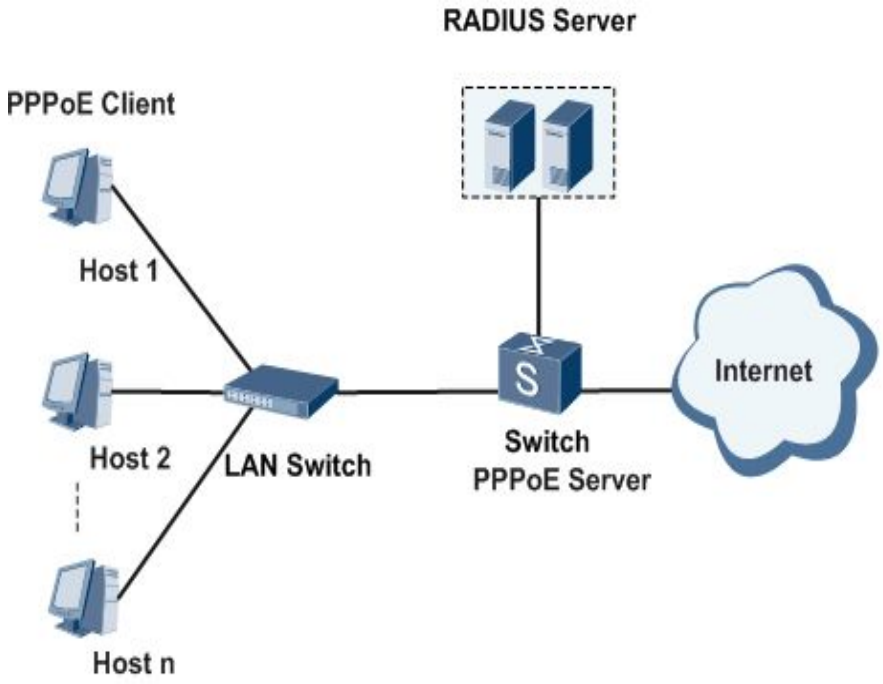
Existing PPPoE Design

TECHNISCHE UNIVERSITÄT DARMSTADT

What: network protocol for encapsulating PPP frames inside Ethernet frames (basically, a tunneling protocol).

Why: most DSL providers use PPPoE which provides authentication via the PAP protocol.

How:



Img Src: http://support.huawei.com/enterprise/docinforeader.action?contentId=DOC1000069540&partNo=10082

Motivation

TECHNISCHE UNIVERSITÄT DARMSTADT

- PPPoE deployment common flavors:-
  - PPPoEoA (PPPoE over ATM) i.e modem-router connected to a DSL service
  - PPPoE i.e DSL modem connected to an Ethernet router

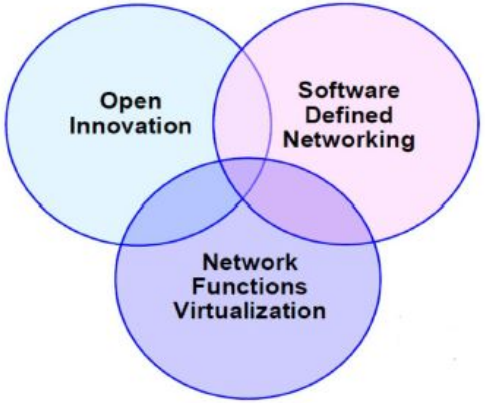
Using **dpdk**, it is possible to virtualize PPPoE server functions in a commodity hardware - low cost solution, easy maintenance, quick bug fixes and faster **Time To Market**.

- **Our Approach:** PPPoE server as an NFV implemented using DPDK libraries. Performance currently is upto 1 Mpps with 64 Bytes packet in Linux kernel.

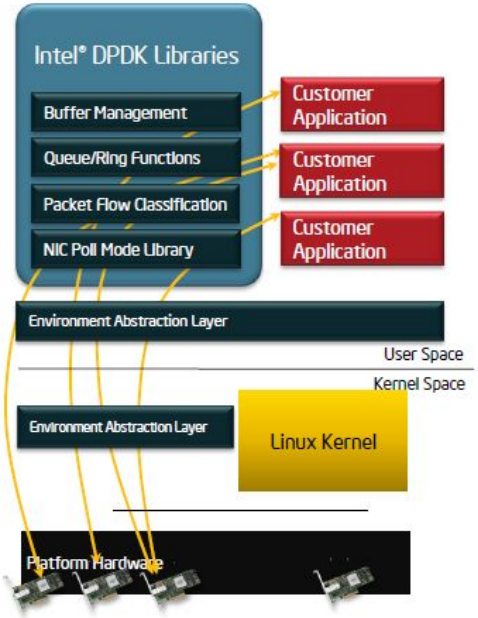
NFV and DPDK

TECHNISCHE UNIVERSITÄT DARMSTADT

NFV: means to implement network functions in software



NFV vs. SDN:
lifting off functions/
services vs. control plane



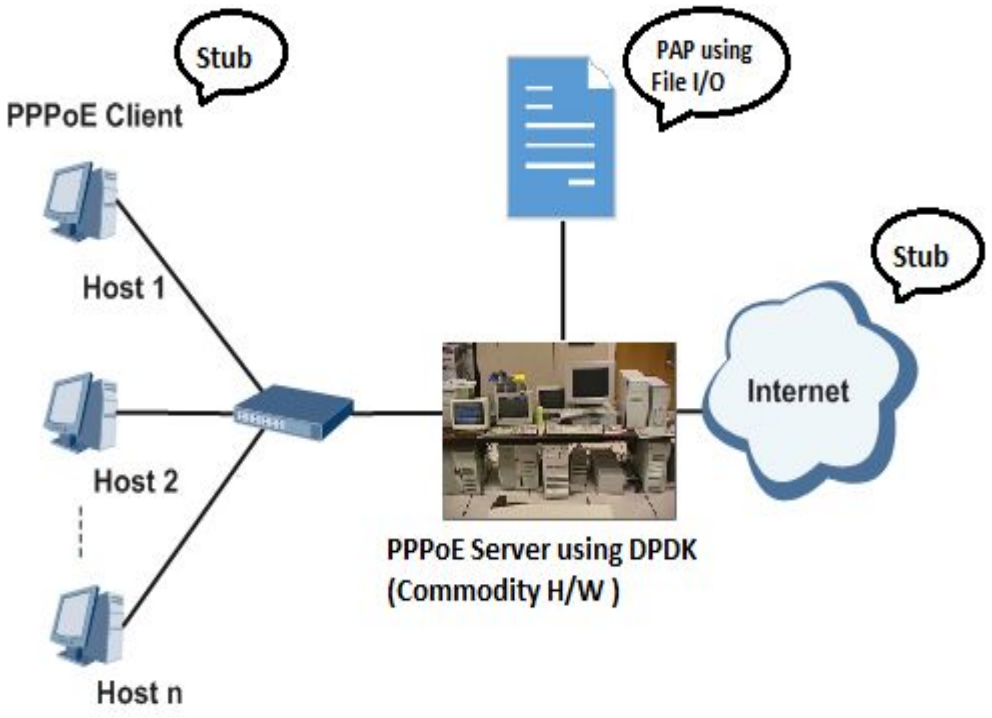
DPDK architecture:-
avoids Kernel bottleneck, DMA to Network Functions at user-space

[img src: http://www.intel.de/content/dam/www/public/us/en/documents/presenta overview-presentation.pdf]

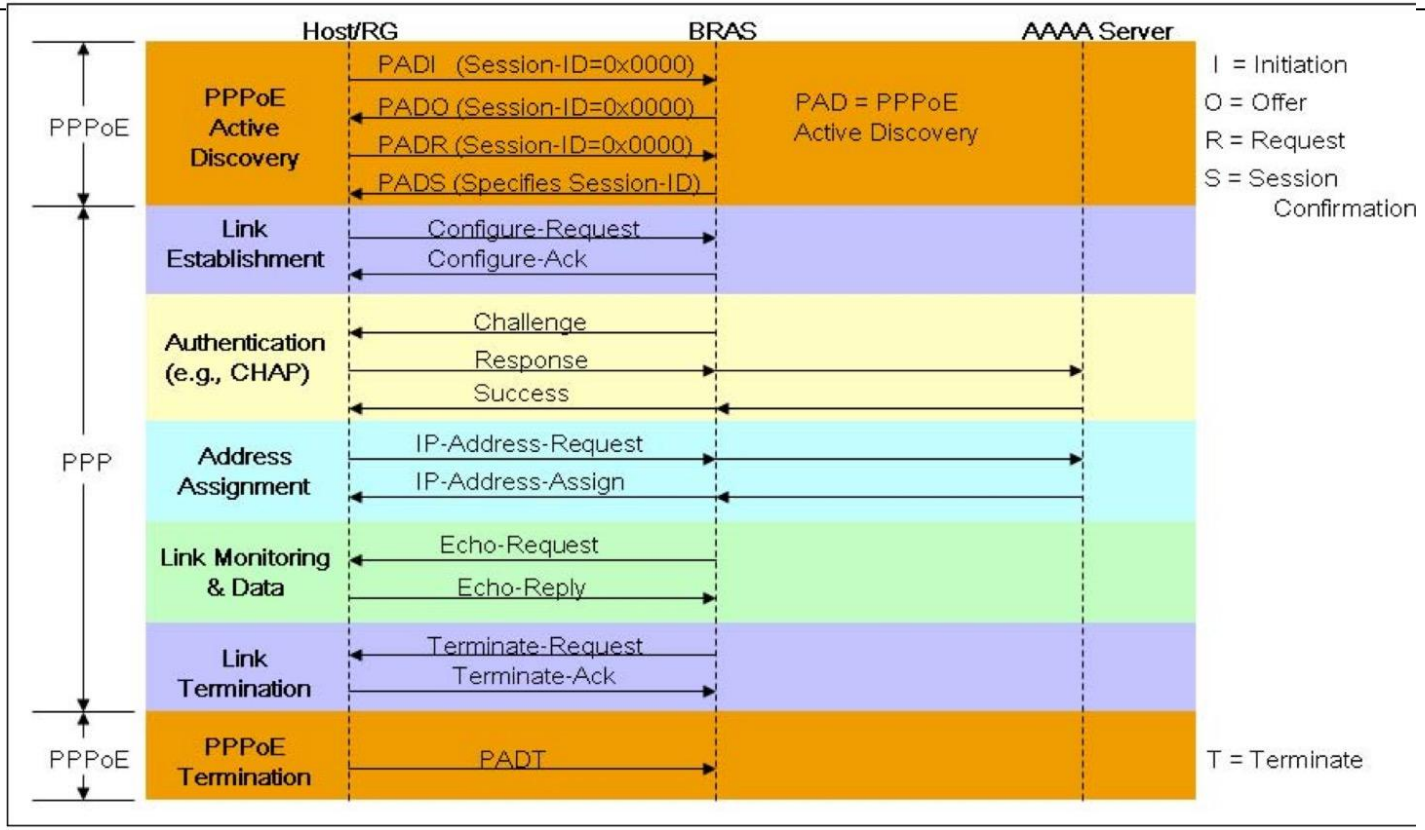
Proposed Design

TECHNISCHE UNIVERSITÄT DARMSTADT

- PPPoE Server (AC) on a commodity h/w using DPDK
- PPPoE client stub using traffic generators
- Password Auth. Protocol (PAP) through passwords in a txt file
- ISP interface stub using DPDK and custom programs



PPPoE in detail - Design



Different phases involved:

- Discovery stage - Allow host to discover all Access Concentrator and then select one.
- PPP session–Once PPP session is established, authentication & resource allocation happens.

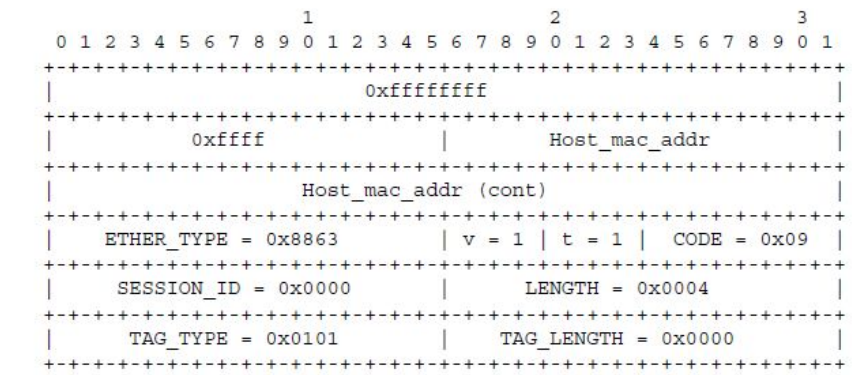
PPPoE in detail - Discovery

TECHNISCHE UNIVERSITÄT DARMSTADT

There are four steps to the Discovery stage:

- The PPPoE Active Discovery Initiation (PADI) packet
- The PPPoE Active Discovery offer (PADO) packet
- The PPPoE Active Discovery Request (PADR) packet
- The PPPoE Active Discovery Session-confirmation (PADS) packet

A PADI packet:



- Our Approach:
  - Identify PADI packet (using CODE field in above figure) from host using DPDK packet modification functions.
  - Modify packet to send to host a PADO packet and wait for PADR from host.
  - Reply with PPP session cconfirmation packet.

PPPoE in detail - Authentication

TECHNISCHE UNIVERSITÄT DARMSTADT

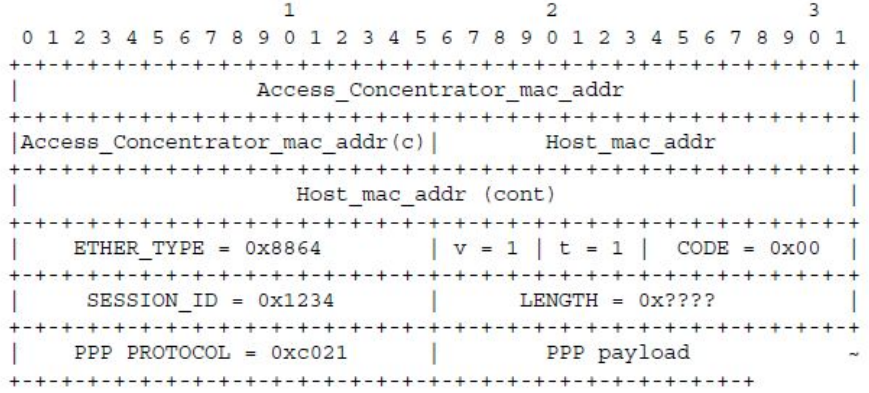
- Peer needs to authenticate itself before allowing network-layer protocol packets to be exchanged.

- Our Approach:
  - Use Password Authentication protocol (PAP)
  - Server reads from a file containing username and password.
  - When Authenticated, server proceeds with further steps in PPP
  - When failed server proceeds to Link termination phase.

PPPoE in detail – Session Maintenance

TECHNISCHE UNIVERSITÄT DARMSTADT

- Once the PPPoE session begins, PPP data is sent as PPP encapsulation.
- All Ethernet packets are unicast.
- ETHER_TYPE =0x8864
  PPPoE CODE = 0x00
  SESSION_ID = value assigned in Discovery stage and must not change for the entire PPPoE session



- Our Approach:
  - Generate a session ID (Peer ethernet address + session ID define PPPoE session uniquely)
  - Maintain a table that stores following tuple:
    <SESSION_ID, Host ETHERNET_ADDR, Host IP_ADDR>

Session Traffic Optimization

TECHNISCHE UNIVERSITÄT DARMSTADT

- Hash mapped session to IP lookup with O(1) lookup time.
- Automatic session termination using timer.
- Packet processing in parallel using multi-cores.
- Looking for more optimization approaches.

Milestone

TECHNISCHE UNIVERSITÄT DARMSTADT

- Topic research - 5th May
- Design finalization - 11th May
- Basic server implementation - 5th June
- Stubs implementation and end-to-end test - 18th June.
- Optimization - 30th June

TECHNISCHE UNIVERSITÄT DARMSTADT

Thank you for your attention!

Questions?