Goals

- Understand basic system components
  - hardware: GPE, NPE, CP, Line Card
  - control software: SRM, RMP, SCD
- Understand configurable resources
  - slow path endpoints
  - fastpath logical interfaces, queues and filters
- Various utilities and daemons
  - scfg, ip_fpd, ip_fpc
  - what they are and what they’re good for
  - some basic uses
Agenda

- Overview – Jon Turner
  » system components and their roles
- Using the SPPs – Ken Wong
  » reserving and configuring SPP resources
  » monitoring traffic through an SPP app
- Forest Demonstration – Jon Turner
  » example of a GPE-only application
- IPv4 Fastpath Demonstration – John DeHart
  » example of a fastpath application
- Hands-on Session – all
  » using the IPv4 fastpath

SPP Nodes

- SPP is a high performance overlay hosting platform
- Designed to be largely compatible with PlanetLab
- How it’s different from PlanetLab
  » multiple processors, including an NP-blade for slice fastpaths
  » multiple 1 GbE interfaces
  » support for advance reservation of interface bw, NP resources
SPP Deployment in Internet 2

Washington DC Installation

two more nodes Houston and Atlanta later this year
Details

I2 Internet Service

12 Router

SALT

SPP

ProtoGENI

KANS

SPP

ProtoGENI

WASH

ProtoGENI

SPP PLC

7

I2 Optical Connections

Hosting Platform Details

External Switch

Line Card

CP

net FPGA

General Purpose Processing Engine

GPE

NPE

Chassis Switch

Line Card

Network Processing Engine

parse

lookup

header format

queues
Key Control Software Components

- **System Resource Manager (SRM)**
  - runs on Control Processor
  - retrieves slice definitions from SPP-PLC
  - manages all system-level resources and reservations

- **Resource Management Proxy RMP**
  - runs on GPEs (in root vServer)
  - provides API for user slices to access resources

- **Substrate Control Daemon (SCD)**
  - runs on NPE management processor
  - supports NPE resource configuration, statistics reporting

- **Slice Configuration tool (scfg)**
  - command-line interface to RMP

Application Framework

- **Fastpath/slowpath**
  - fastpath mapped onto NPE
  - control daemon in vServer on GPE

- **Configurable elements**
  - code option – determines how packets processed by parse, header format
  - fastpath interfaces
    - map to physical interface
    - provisioned bandwidth
  - TCAM filters
  - Queues
    - length, bandwidth

- **Control daemon can configure fastpath through RMP**
  - or users can configure manually with scfg
Logical Topology and Addresses

sppsalt1.arl.wustl.edu
   .210
   .214
   .218
64.57.23.* .2
           10.1.7.*
           10.1.8.*
           .2

sppwash1.arl.wustl.edu
   .194
   .198
   .204
64.57.23.* .2
           10.1.3.*
           10.1.4.*
           .2

sppkans1.arl.wustl.edu
   .178 .182 .186
64.57.23.* 10.1.1.*

Working with SPPs

- Define new slice using SPP-PLC
  » just like PlanetLab
- Login to slice (on GPE) to install application code
- Reserve resources needed for your experiment
  » includes interface bandwidth on external ports and NPE fastpath resources
- To run experiment during reserved period
  » “claim” reserved resources
  » setup slowpath endpoints
  » configure fastpath (if applicable)
  » setup real-time monitoring
  » run application and start traffic generators
Steps in Running Experiment

- Reserve resources in advance
- At start of session
  - claim resources
  - setup slow path endpoints for use by GPE code
  - configure fastpath (if applicable)
    - specify physical interface, port number, bandwidth
    - bind queues to logical interfaces, set reserved bandwidth
    - install fastpath filters
      - result includes next-hop (IP, port), bandwidth
  - setup traffic monitoring using sliced and sppMon
- At end of session
  - kill sliced, release resources in use, cancel reservation
Creating a Slice

Preparing a Slice

SFTP connection for downloading code

Requires Network Address Translation
datapath detects new connection, LC control processor adds filters
**Configuring a Slowpath Endpoint**

- Request endpoint with requested port number on specific interface thru Resource Manager Proxy (RMP)
- RMP relays request to System Resource Manager (SRM)
- SRM configures LC filters for interface
- Arriving packets directed to slice, which is listening on socket

**Setting Up a Fast Path**

- Request fastpath through RMP
- SRM allocates fastpath
- Specify logical interfaces and interface bandwidths
- Specify #of filters, queues, binding of queues to interfaces, queue lengths and bandwidths
- Configure fastpath filters
Displaying Real-Time Data

- Fastpath maintains traffic counters and queue lengths
- To display traffic data
  - configure an external TCP port
  - run sliced within your vServer, using configured port
    - sliced --ip 64.57.23.194 --port 3552 &
  - on remote machine, run SPPmon.jar
    - use provided GUI to setup monitoring displays
    - SPPmon configures sliced, which polls the NPE running the fastpath
    - SCD-N reads and returns counter values to sliced
    - can also display data written to file within your vServer

Command Line Tools

- **scfg** – general slice configuration tool
  - scfg --cmd make_resrv (cancel_resrv, get_resrvs, ...)
  - scfg --cmd claim_resources
  - scfg --info get_ifaces (...)
  - scfg --cmd setup_sp_endpoint (setup_fp_tunnel, ...)
  - same features (and more) available thru C/C++ API
- **sliced** – monitor traffic and display remotely
- **ip_fpd** – slowpath to go with IPv4 fastpath
- **ip_fpc** – fastpath configuration tool for IPv4 fastpath