

Generic software switch with LiSA

Mihaela Mărtinaş, Andreea Hodea, Victor Duţă and Claudiu Ghioc

Faculty of Computers and Automatic Control

“Politehnica” University of Bucharest

Bucharest, Romania

martinas.mihaela21@gmail.com,

andreea.cristina.hodea@gmail.com,

duta.victor.marin@gmail.com,

claudiu.ghioc@gmail.com

Abstract — LiSA stands for Linux Switching Appliance. A few words about what it used to be, the initial authors, what it wants to do(very few words)

Keywords — *Generic Switch, Operating Systems, Linux Kernel, VLAN.*

I. INTRODUCTION

Describe the paper content in a paragraph.

II. LiSA CURRENT FEATURES

(WHY)

- Here will talk about what it currently does: switching on Layer 2 and 3, Cisco-like CLI, 802.1q support, CDP, RSTP and IGMP snooping
- Why is the Lisa kernel module the faster solution to switch packets (we have the reasons in Radu's diploma paper);
- Why isn't LiSA .ko acceptable
- Very short description of the solution(s)

III. BRIDGE + 8021Q COMPARED TO LiSA

Here I was thinking to mention the drawbacks of the Bridge + 8021q implementation, the fact that you need one virtual interface for each VLAN and again some other differences mentioned by Radu.

IV. GENERIC SWITCH ARCHITECTURE

(WHAT)

- Present the main three option of switch back-ends: Lisa, Bridge + 8021q and hardware switch
- The generic switch API structure (and a picture)
- A few words about how the CLI and the daemons would adapt to the new user-space architecture.

V. LiSA BACK-END IMPLEMENTATION

(HOW)

- Describe how exactly we implement the switch API using the LiSA kernel modules.
- We could mention how we have to overwrite all the functions in the API using an “operations” structure, the type of sockets used, the kernel style lists
- Specify the data held in the shared memory zone

VI. BRIDGE AND 8021Q BACK-END

- Describe the tools used: brctl, vconfig and the way they communicate with their kernel modules (sockets, ioctl)
- Implementation details about the list of subinterfaces, vlan specific data, allowed vlan bitmaps etc.
- Again, mention the switch operations structure and the fact the the CLI level doesn't have to change; The back-end can be chosen at compile-time.

VII. CDP AND RSTP

- Describe how these daemons work, how they communicate with the CLI;
- Specify what they use to communicate with the kernel (raw sockets, libpcap, libnet etc)
- More implementation details + picture

VIII. FURTHER WORK

- The third back-end implementation and its benefits: LiSA for hardware switch;
- Integrate with Openwrt, the multi-engine switch
- Include in a popular Linux distribution

REFERENCES

Radu's and Ionut's diploma thesis etc