AstriCon
NOV 3-4 VIRTUAL CONFERENCE '21

Asterisk & LRKProxy

Mojtaba Esfandiari.S

dept. of R&D member of
NasimTelecom
Tehran, Iran

AstriCon
NOV 3-4 VIRTUAL CONFERENCE '21

Who am I…

. phd student in Azad university of Mashhad

. VoIP consultant and software engineer

. 15 years experience in developing voice over ip projects

. Wrote a book about Asterisk in Persian language.

. dCAP No. 2265 April 2014

. Member of Kamailio developer (lrkproxy, hiops)

AstriCon

NOV 3-4 VIRTUAL CONFERENCE '21

dCAP

Digium Certified Asterisk Professional

Mojtaba Esfandiari

has completed all requirements for
and is hereby recognized as

dCAP No. 2265

Date certified: April 18, 2014

Digium Training Manager

digium®

The Asterisk Company

Mark Spencer, CTO, Digium Inc.

AstriCon

NOV 3-4 VIRTUAL CONFERENCE '21

VOIP

Asterisk Softswitch

مرجع آموزش ویپ با سافت سوئیچ استریسک

مجتبی اسفندیاری
سید مجتبی نجفی مقدم

مرجع آموزش ویپ با
سافت سوئیچ استریسک

I started with Asterisk in 2002 while I was still in college (Asterisk was born in 1999), and was able to build a consulting business around it for over a decade. It's amazing how Asterisk has continued to be the definitive open source VoIP platform for building new applications and businesses. I've continued to follow Asterisk over the years because it stays close to my heart, and it's impressive to see how it has continued to grow and transform with the technological revolution of the internet.

Asterisk continues to be a strong platform for VoIP development, and I can't imagine a better base platform to learn if you're involved in modern VoIP applications.

I'm glad to see Asterisk is still a topic that people are interested in writing about. It's a pleasure to see that Mr. Esfandiari.S et al. have written a book about Asterisk with new topics and headlines in Persian. I hope that the growth of such books in different languages could help students and interested people to be more familiar with Asterisk and to be able to grow VoIP technology and adoption through deployments and whitepapers.

— Leif Madsen

9 789648 931952

انتشارات واژگان خرد

What is LRKProxy?

What is LREProxy

. The LREProxy is Kernel RTP engine for relaying RTP packets crossing NIC in your network.

The LREProxy architecture is composed of two different layers.

. LREP_Controlling Layer (LREP_CL):
The first layer is developed as User-Space application

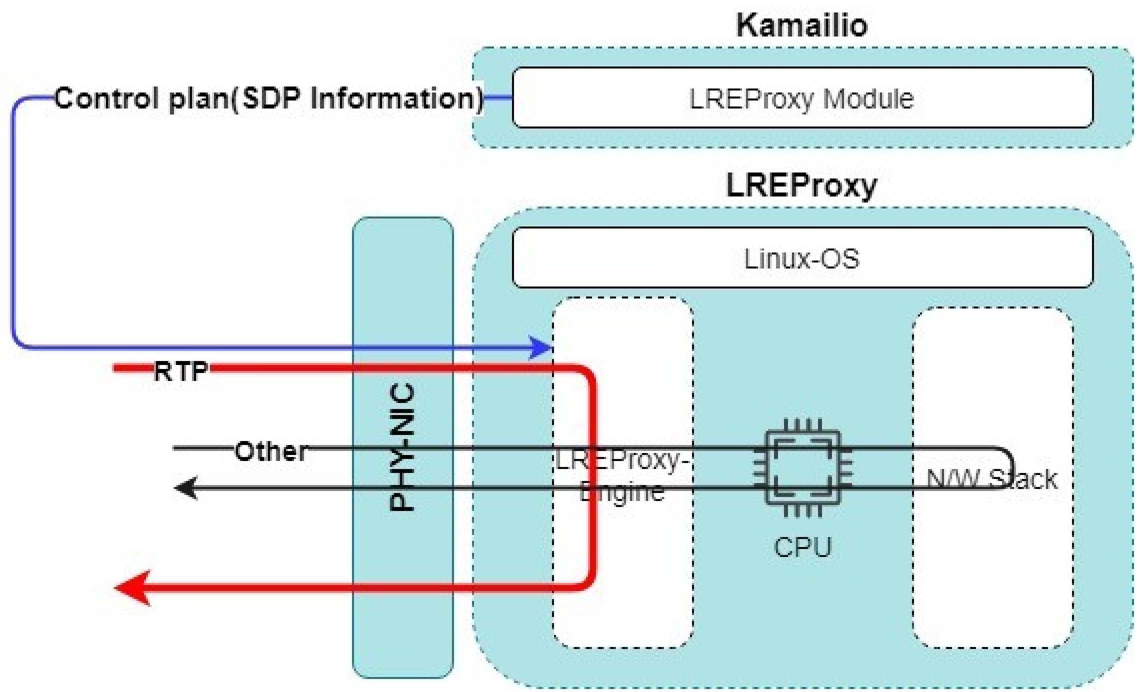
. LREP_Transport Stateful Layer (LREP_TSL)
The second layer is developed in Kernel-Space as a main decision point for RTP admission controller and Quickpath selector.

LREProxy Architechture

The LREProxy architecture is composed of two different layers.

LRKP_Controlling Layer (LRKP_CL)

LRKP_Transport Stateful Layer (LRKP_TSL)



Expanding Asterisk with Kamailio

. Deploying large networks have specific challenges.
. Most of those challenges are related to network design and topology

. Redundancy (Signaling layer – Transport Layer)
 - Active – Active
 - Active – Passive
. Scaling
 - Number of calls
 - Number of users
. High- Availability
 - Reliable service
 - Failover service

. Resources Management

. Security

Fred Posner
@fredposner

Two Important Factors

. Two important factors that increase the consumption of resources in a network.

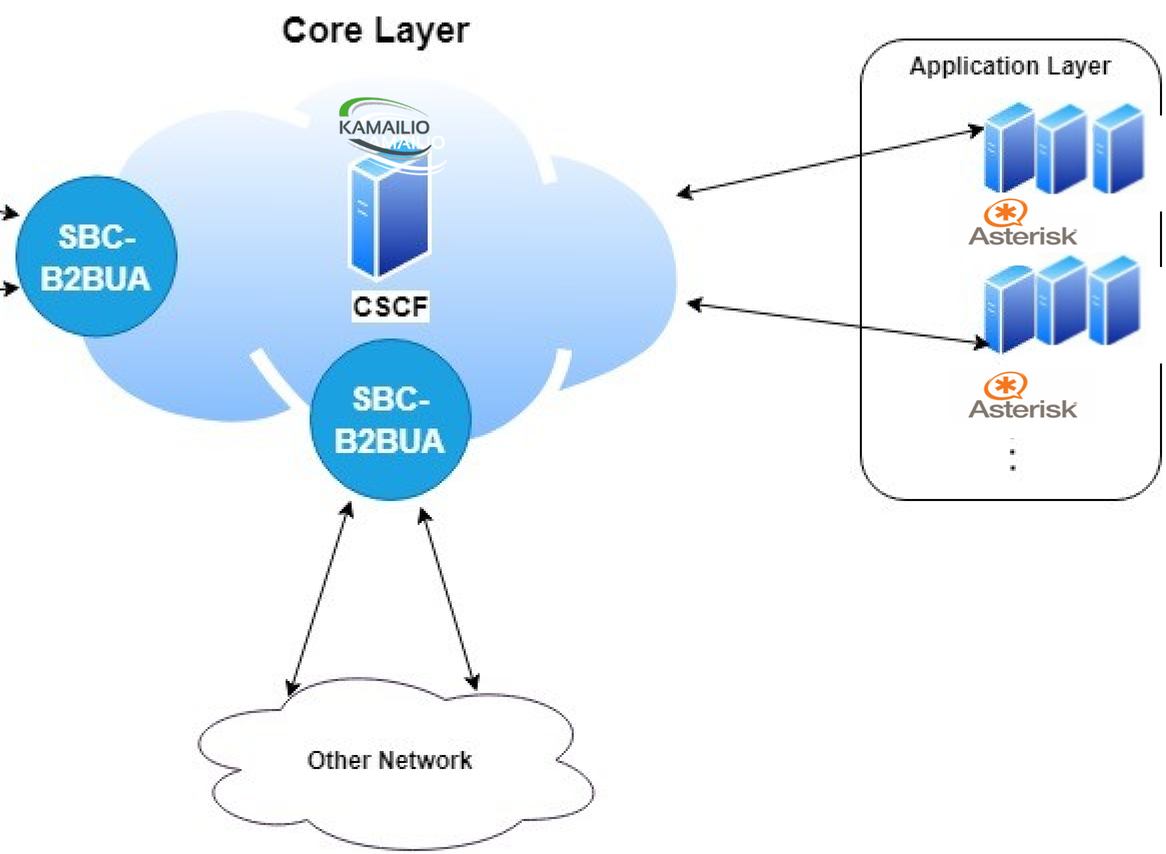
. Call per second (CPS) SIP signaling

. RTP relaying – RTP Flow

System Tester

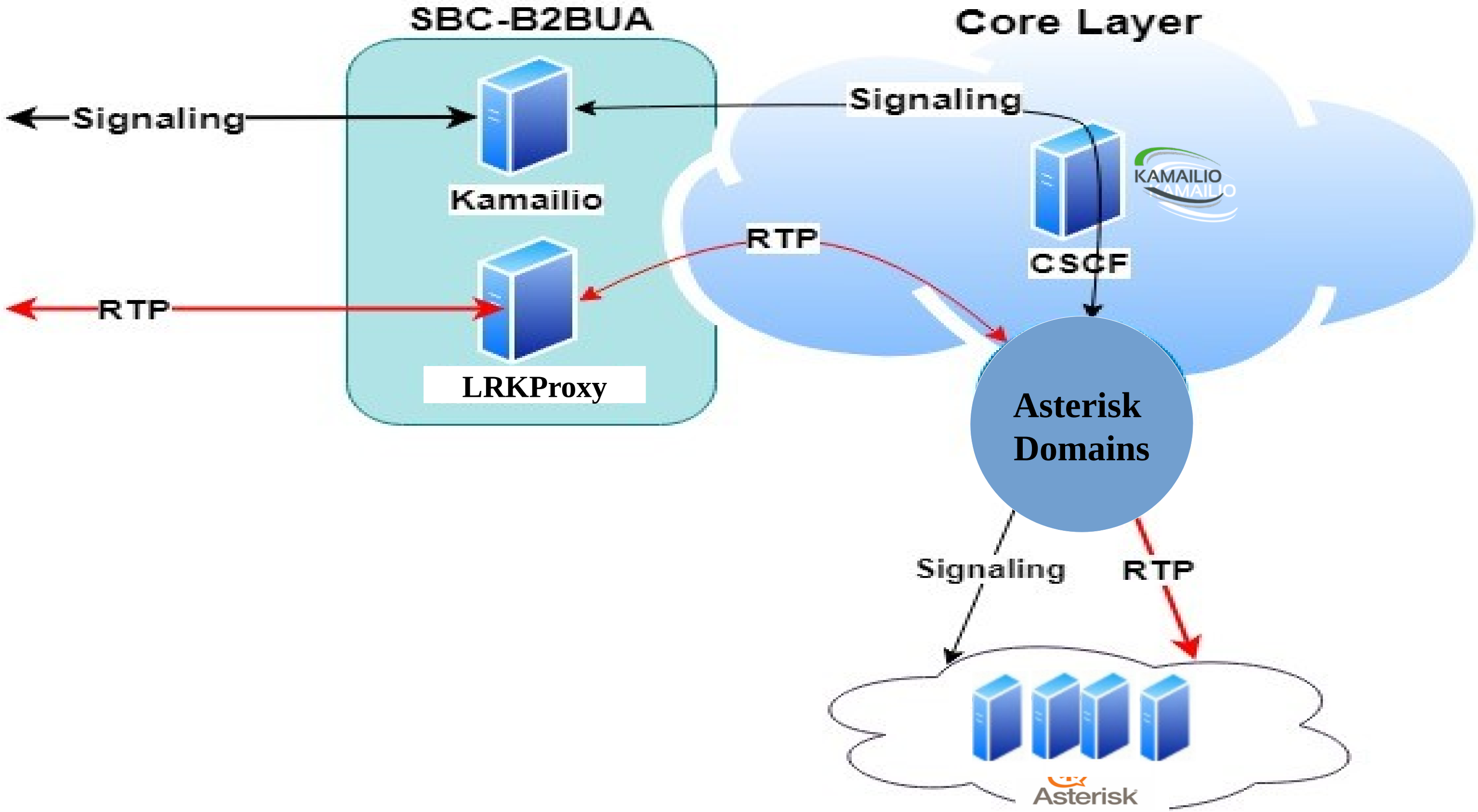
SPIRENT ABACUS 5000

SIPp



AstriCon NOV 3-4 VIRTUAL CONFERENCE '21

System of Model – Asterisk & LRKProxy

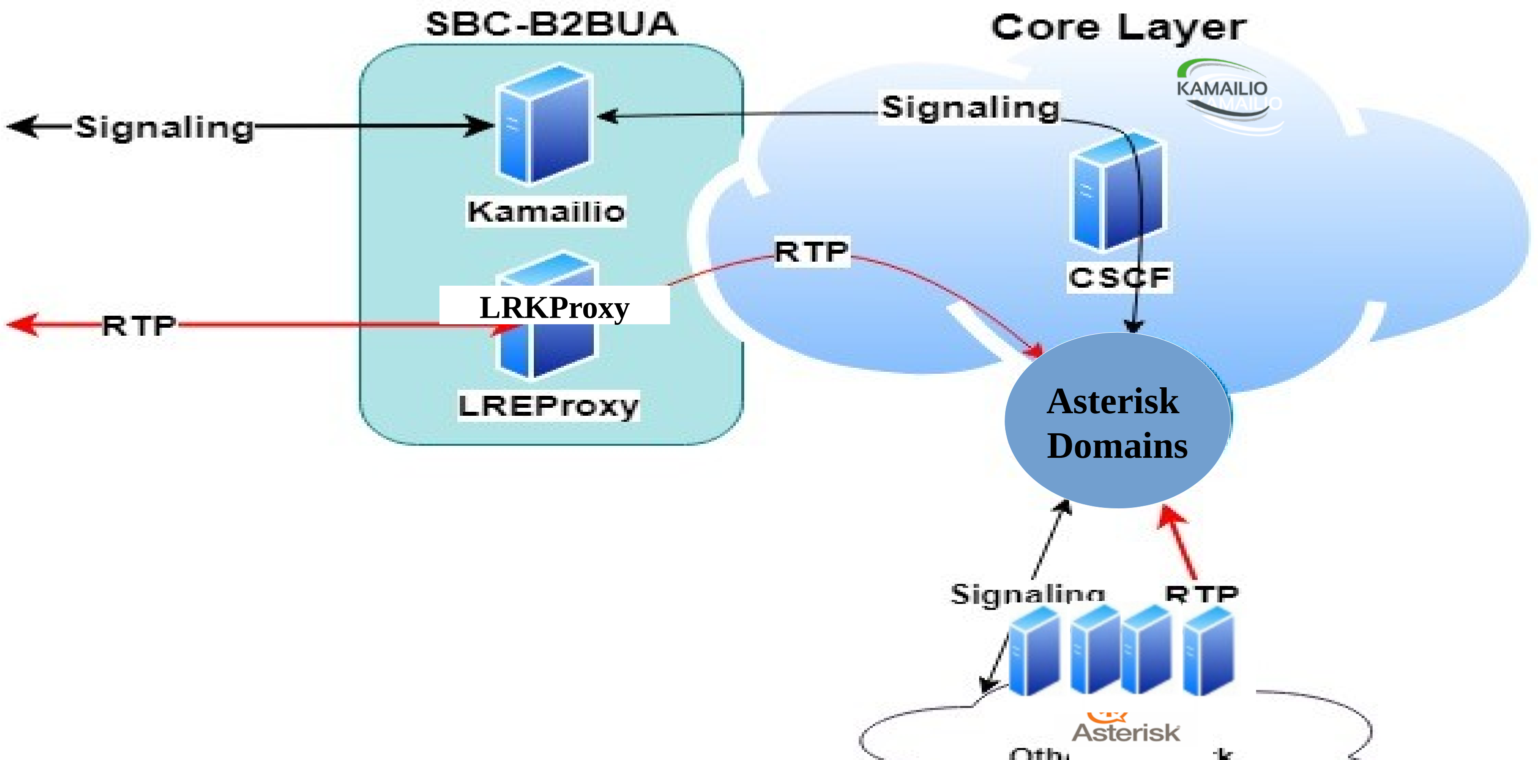


System of Model

Q2?

. How many calls does the SoM handle?

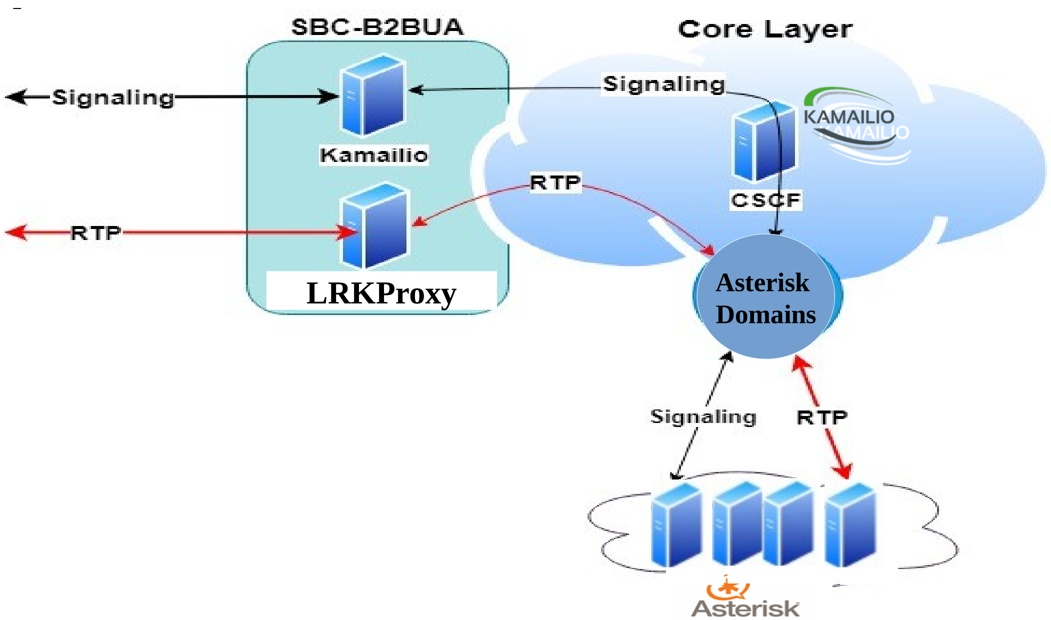
. Which Parts of the SoM is responsible for it?



Bottleneck Points

. Signaling Flow
- SBC(Kamailio) in Edge
- Kamailio in CSCF
- Asterisk Servers

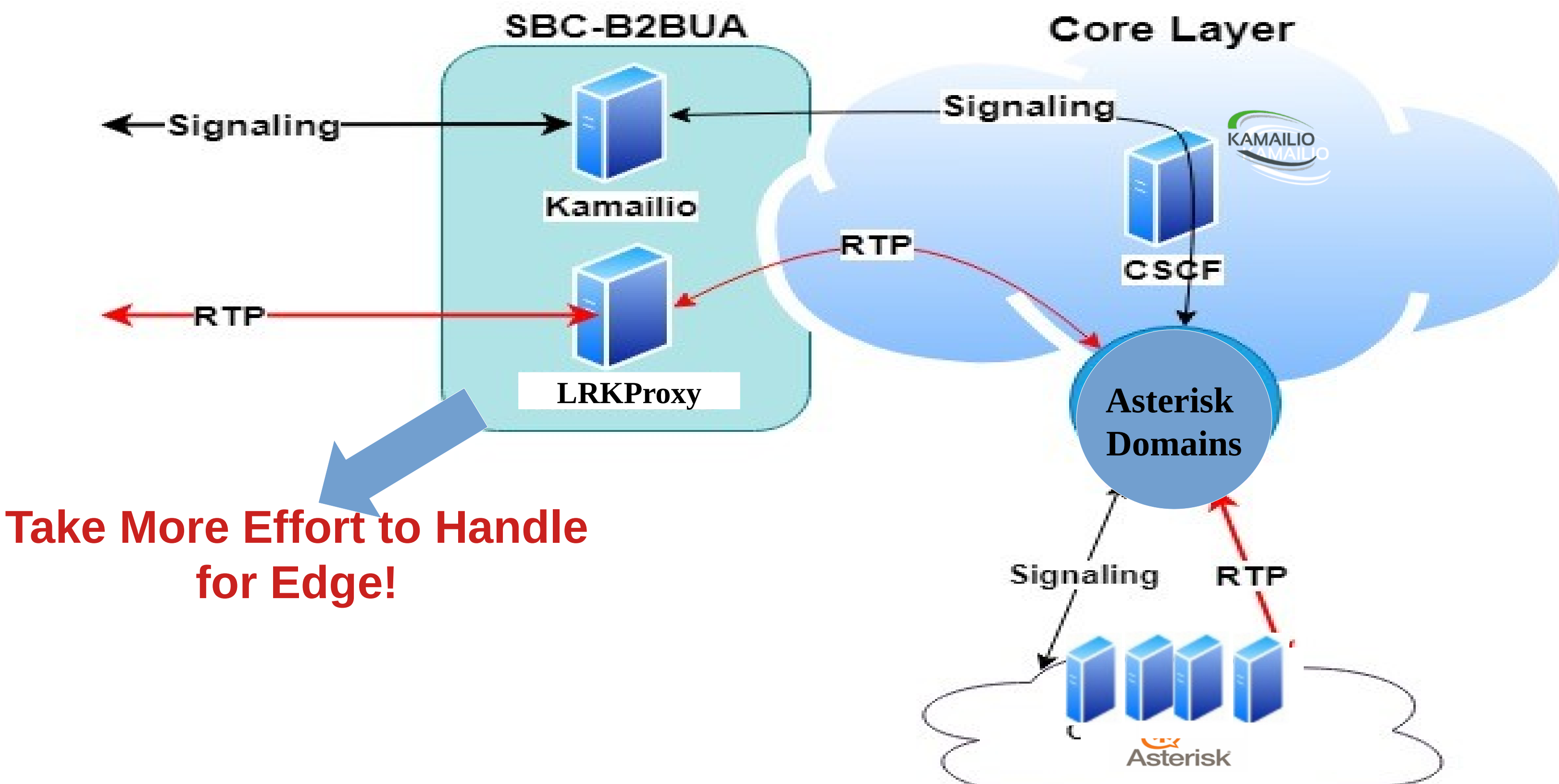
. RTP Flow
- LRKProxy in Edge
- Media servers in CSCF (rtpproxy, rtpengine, lrkproxy, …)
- Asteri



Resource Management

Q3?

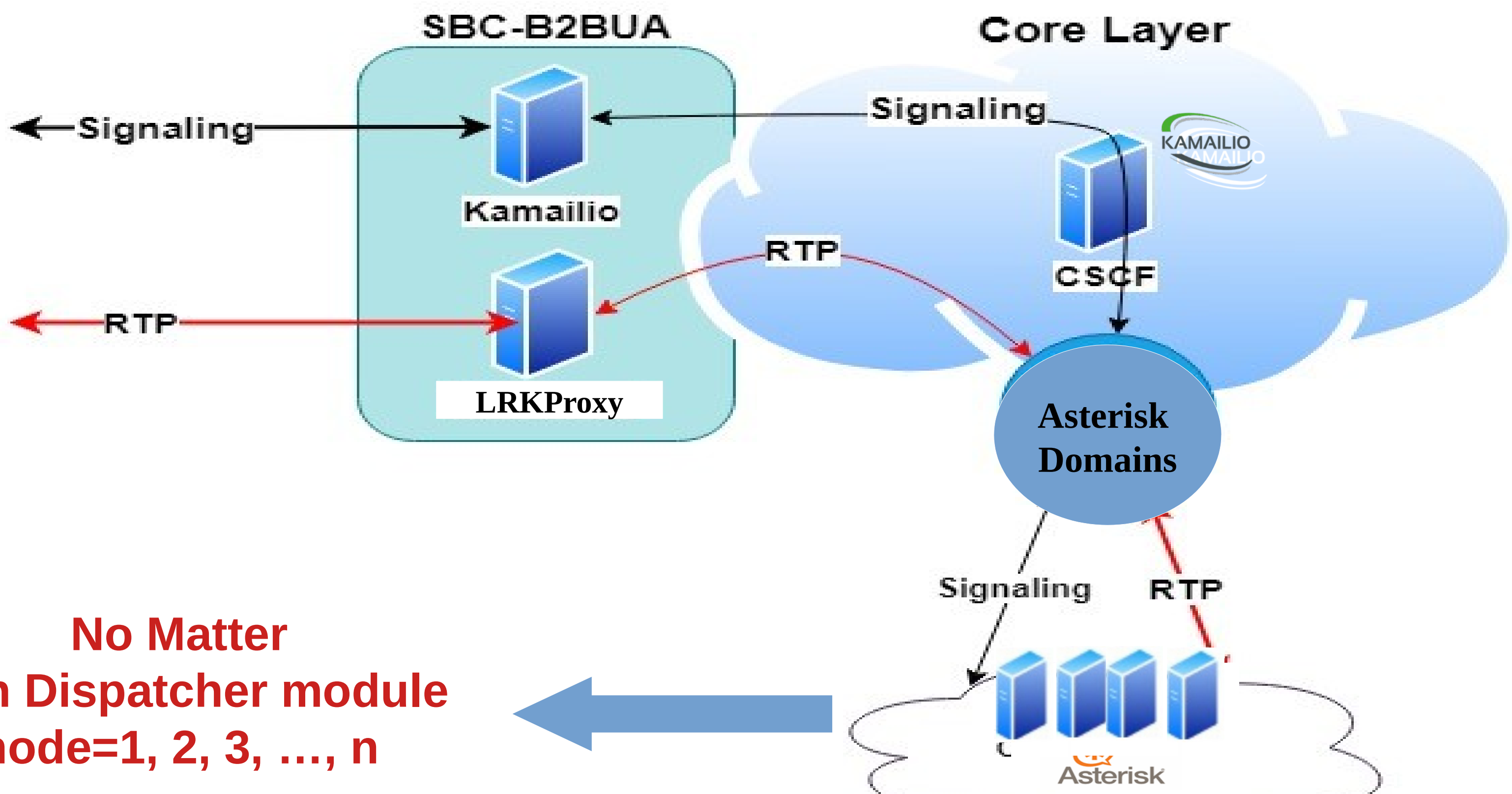
. How many recources (LoM) does the SoM need for handling 20000 calls or more?



Resource Management

Q4?

. How many Asterisk does the SoM need for handling 20000 calls or more?

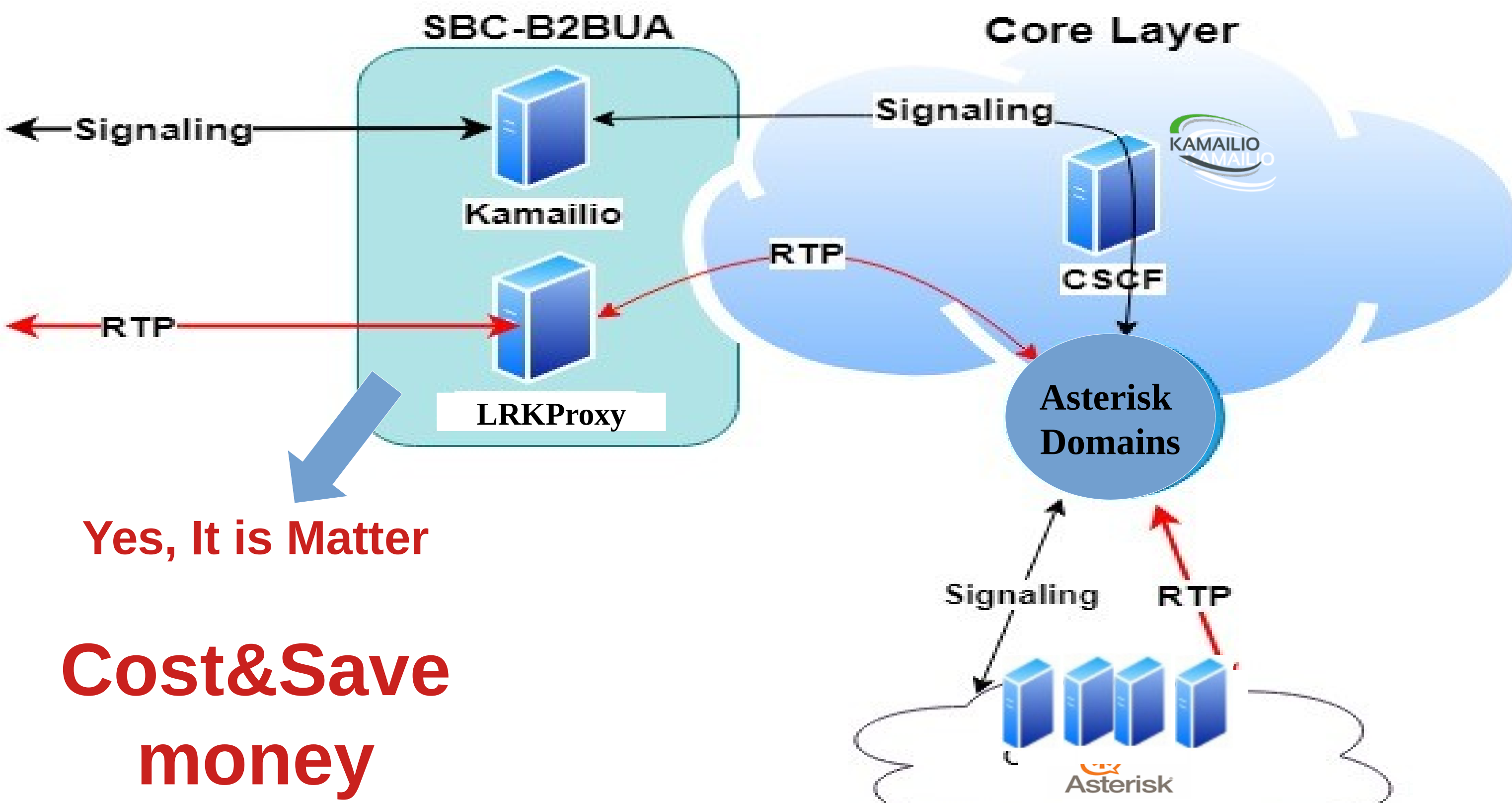


No Matter
With Dispatcher module
node=1, 2, 3, …, n

Resource Management

Q5?

. How many LRKProxy does the SoM need for handling 20000 calls or more?



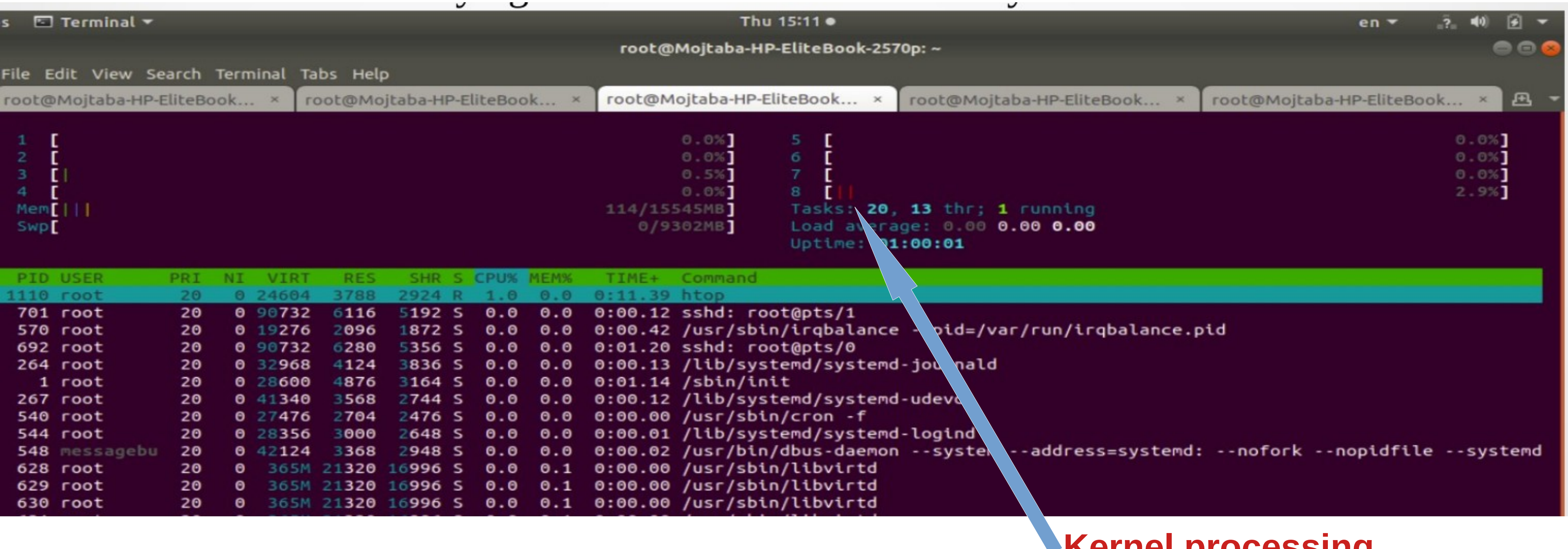
Yes, It is Matter

Cost&Save money

LRKProxy Resource Management

. In practical experiment, we used Sipp tools and Abacus 5000 for making calls and generating RTP media.

. The 20000 concurrent relaying RTP sessions with LRKProxy in one server.



Kernel processing

AstriCon NOV 3-4 VIRTUAL CONFERENCE '21

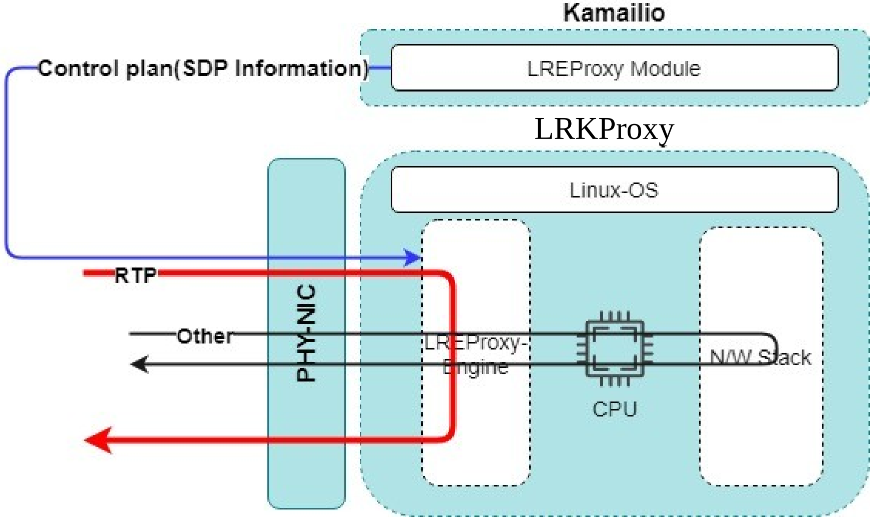
RTP Flow in LRKProxy

. LRKP_Controlling Layer (LRKP_CL)

The first layer gets all information from SDP body during signaling and relay them to the LRKP-Transport Stateful Layer (LRKP-TSL) for hashing.

```
2021-09-20 15:20:48,431 [81408] DEBUG  udp_socket_service line:53 Received Command S: 716_1 => b'716_1 S 192.168.43.205 194.62.43.227 192.168.43.227 93.115.147.156 16592 20012 20014 8000 60 361EE027-61486786
27D2-204F2700 '
2021-09-20 15:20:48,437 [81408] DEBUG  udp_socket_service line:87 Sent response to client: b'716_1 S 192.168.43.205 194.62.43.227 192.168.43.227 93.115.147.156 16592 20012 20014 8000 60 361EE027-614867860003
-204F2700 OK'
2021-09-20 15:20:48,450 [81408] DEBUG  udp_socket_service line:53 Received Command S: 699_1 => b'699_1 S 93.115.147.156 192.168.43.227 194.62.43.227 192.168.43.205 4000 20008 20010 15658 60 7ABC3232-61486785
B404-206F4700 '
2021-09-20 15:20:48,451 [81408] DEBUG  udp_socket_service line:87 Sent response to client: b'699_1 S 93.115.147.156 192.168.43.227 194.62.43.227 192.168.43.205 4000 20008 20010 15658 60 7ABC3232-61486785000D
-206F4700 OK'
2021-09-20 15:20:48,736 [1952] DEBUG  unix_socket_client_service line:172 Successfully sent data to kernel_space: b'716_1 S 192.168.43.205 194.62.43.227 192.168.43.227 93.115.147.156 16592 20012 20014 8000 6
1EE027-61486786000327D2-204F2700 '
2021-09-20 15:20:48,737 [1952] DEBUG  unix_socket_client_service line:172 Successfully sent data to kernel_space: b'699_1 S 93.115.147.156 192.168.43.227 194.62.43.227 192.168.43.205 4000 20008 20010 15658 6
BC3232-61486785000DB404-206F4700 '
```

Received Command S: 716_1 => b'716_1 S
src_ip: 192.168.43.205
dst_ip: 194.62.43.227
snat_ip: 192.168.43.227
dnat_ip: 93.115.147.156
sport: 16592
dport: 20012
snat_port: 20014
dnat_port: 8000
Timeout: 60
Call-ID:361EE027-61486786000327D2-204F2700 '



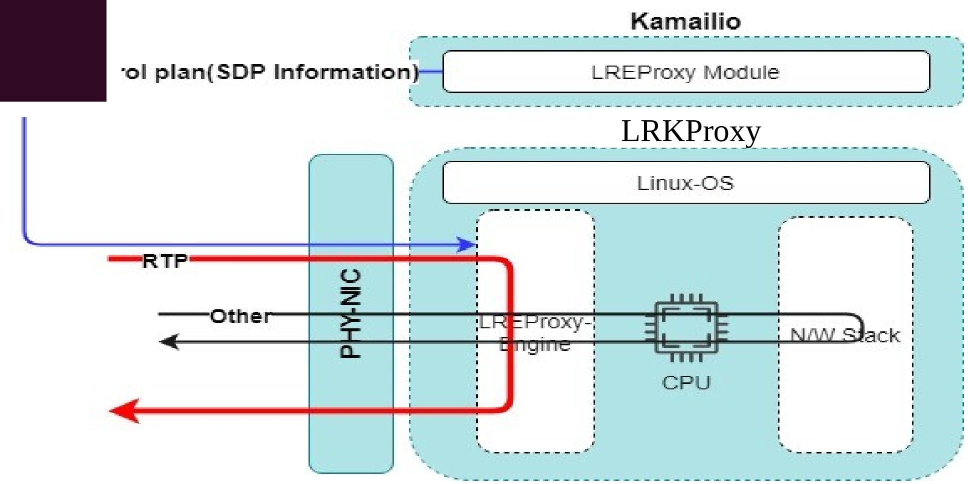
AstriCon NOV 3-4 VIRTUAL CONFERENCE '21

RTP Flow in LRKProxy

. LRKP_Transport Stateful Layer (LRKP_TSL)

The second layer is a main decision point for RTP admission controller and received packets should be forwarded with power of **NF_INET_PREROUTING** in Netfilter hook.

```
lrkproxy_force(): selected node: udp:127.0.0.1:22333
lrkproxy_force(): call_is: 361EE027-614867860003270D2-204F2700
lrkproxy_force(): viabranch: z9hG4bKcmtNpaKe
lrkproxy_force(): src_ipv4: 192.168.43.205
lrkproxy_force(): src_port: 16592
lrkproxy_force(): dst_ipv4: 194.62.43.227
lrkproxy_force(): dst_port: 20012
lrkproxy_force(): dnat_ipv4: 93.115.147.156
lrkproxy_force(): dnat_port: 8000
lrkproxy_force(): snat_ipv4: 192.168.43.227
lrkproxy_force(): snat_port: 20014
```



AstriCon
NOV 3-4 VIRTUAL CONFERENCE '21

LRKProxy Config

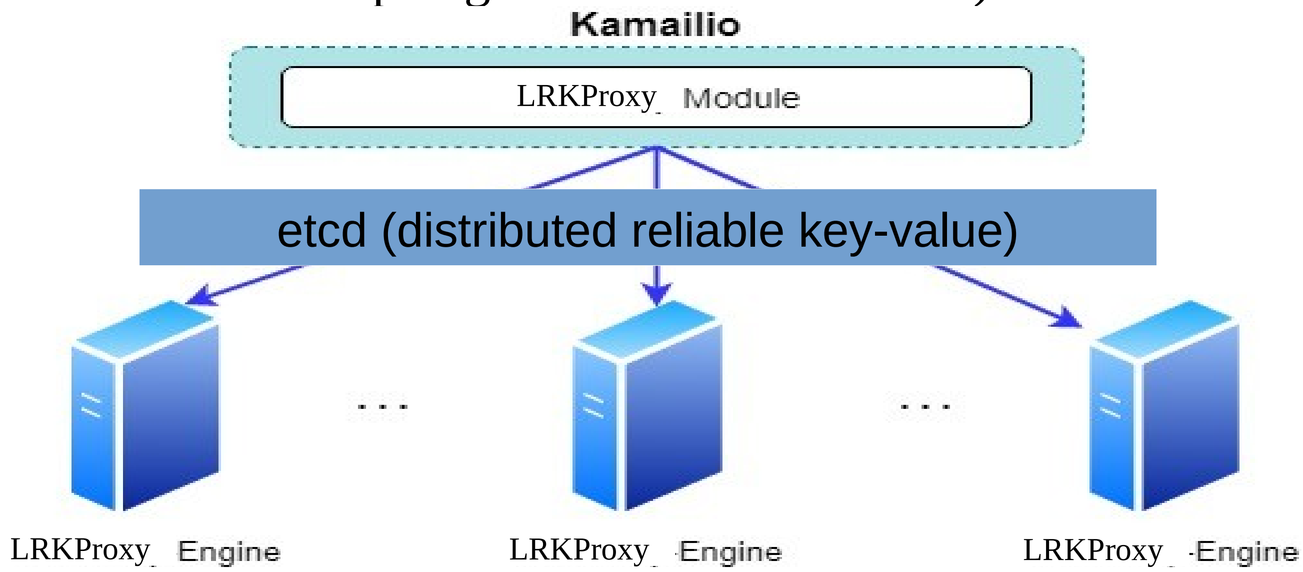
```
# ----------------- setting module-specific parameters ---------------
#!ifdef WITH_LRKPROXY
loadmodule "lrkproxy.so"
modparam("lrkproxy", "lrkproxy_sock", "udp:127.0.0.1:22333")
modparam("lrkproxy", "custom_sdp_ip_avp", "$avp(RR_CUSTOM_SDP_IP_AVP)")
#!endif

# ----------------- request_route ---------------
#!ifdef WITH_LRKPROXY
        if (is_present_hf("PRE_SOURCE"))
            $avp(RR_CUSTOM_SDP_IP_AVP) = $(hdr(PRE_SOURCE));
        set_lrkproxy_set("0");
        lrkproxy_manage("ei");          // or lrkproxy_manage("ie");
#!endif
```

/etc/pylrkproxy/pylrkproxy.ini

```
[kernel]
start_port : 20000
end_port : 40000
current_port : 20000
internal_ip : 192.168.43.227

;It is under development
external_ip : 194.62.43.227

[UDP socket]
socket_udp_host = 0.0.0.0
socket_udp_port = 22333

[Cache]
save_call_cache = False

[UNIX socket]
forward_to = /root/sock

[logger]
log_to_file = True
log_to_console = False
```

Multiple LRKProxy Config

. The LRKP_CL and LRKP-TSL could be run as independence functions on different machines.
. We could have one LRKP_CL with multiple LRKP-TSL on different machines.
. It possibles to do not lose sessions when a LRKProxy engine crashes. (under development with etcd - https://github.com/etcd-io/etcd)



LRKProxy: Pros & Cons

. Changing SDP parameter will be done by Kamailio, not by LRKProxy engine.
. Forwarding packet in NF_INET_PREROUTING hook.
. NAT Travelsal for endpoint.
. Set Custom SDP information by AVP
. Suppport RTCP.
. Resource port allocation with the consept of Game-Theory(under development) for reusing port in RTP sessions.
. Enable forwarding in sysctl’s conf file:

    net.ipv4.ip_forward=1

. Does not support transcoding
. Could make anchor for target RTP for Lawful interception (under development)

LRKProxy & IEEEExplore

. https://ieeexplore.ieee.org/document/9303608

10th International Conference on Computer and Knowledge Engineering (ICCKE2020) October 29-30, 2020 – Ferdowsi University of Mashhad - Iran

Improve Performance of RTP Relaying Sessions in IMS Transport Layer With LREProxy

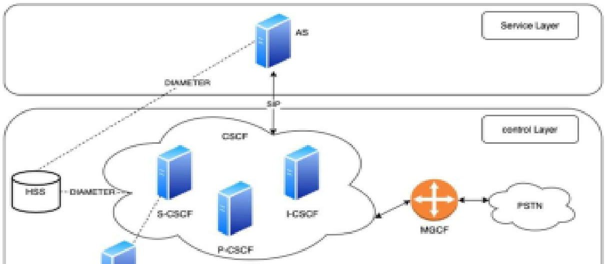
1st Mojtaba Esfandiari.S
dept. of Computer Engineering, Ferdowsi University of Mashhad, Iran
dept. of R&D member of NasimTelecom knowledge-base co, Tehran, Iran
Email: esfandiari@nasimtelecom.com

2nd Seyed Mojtaba Najafi
dept. of Telecommunication Engineering, Islamic Azad University, Mashhad, Iran
dept. of R&D member of NasimTelecom knowledge-base co, Tehran, Iran
Email: najafi@nasimtelecom.com

3rd Morteza Iravani
dept. of Computer Engineering, Ferdowsi University of Mashhad, Iran
dept. of R&D member of NasimTelecom knowledge-base co, Tehran, Iran
Email: iravani@nasimtelecom.com

4th Sajad sabri
dept. of Computer engineering, Shafagh Institute, Tonekabon, Iran
dept. of R&D member of NasimTelecom knowledge-base co, Tehran, Iran
Email: sabri@nasimtelecom.com

***Abstract*—The IP Multimedia Subsystem is an architectural network for delivering IP multimedia services and data. The IMS network has built on three layers which allows for the convergence of different access networks. Each layer in IMS is consisting of various elements and protocols that consequently process signaling or media flow to specific application service. While the Call Session Control Function (CSCF) is the main route decision of the IMS network, the most significant of resources are used by routing and delivering media flow in Transport data layer in IMS network. With raising of request for service in IMS, the usage of resource have been increased. The ingress and egress nodes in IMS network are critical points and could potentially being bottleneck because they have to transmit huge signaling and media packets from and to IMS network. In this paper, we focused on Transport data layer**



Thank you

Mojtaba Esfandiari.S

Email:
esfandiari@nasimtelecom.com
esfandiari.m84@gmail.com

Twitter:
@MespioS

dept. of R&D member of
NasimTelecom
https://nasimtelecom.com/en/