Troubleshooting and Tuning your VoIP services in OpenSIPS 2.1

Răzvan Crainea

OpenSIPS Project

razvan@opensips.org

Nov 8 - 9

Overview

Overview

FAQ

- How many CPS does my platform support?
- How much memory/CPU should I use to support X CPS?
- Will my hardware support Y CPS?
- How do I increase the performance of my platform?

Dependencies

- Hardware (CPU, memory)
- Traffic patterns
- Type of service
- Calls per second
- Database Queries
- DNS Performance
- Network Bandwidth
- Gateways quality
- Script logic
- Average call duration
- Database Performance
- CNAM/LRN/LNP dips

Answer

- Testing
  - sipp
  - real traffic
- Observing
  - logs
  - statistics
- Tuning

Overview

Allocators

- F_MALLOC (Fast Malloc)
- Q_MALLOC (Debugging info)
- VQ_MALLOC (Obsolete)

Allocators

- F_MALLOC (Fast Malloc)
- Q_MALLOC (Debugging info)
- VQ_MALLOC (Obsolete)

- Issues
  - Startup fragmenting contention
  - Use a single "big lock" mechanism

High Performance Malloc

- Efficient Startup
  - Memory is already fragmented based on the previous pattern
  - The allocator "learns" the memory footprint
  - Pre-fragments the memory
- Multiple locks
  - one or multiple locks per bucket
  - reduced contention
- **Really fast** memory allocator

Statistics

- Used memory (`used_size`)
  - the memory used by your service
  - it is the same for all allocators
- Real used memory (`real_used_size`)
  - the total memory used, including metadata
- Maximum used memory (`max_used_size`)
  - the maximum amount of memory used

Configuration Tips

- Set the maximum amount of memory available
- Put an amount of testing traffic
- Scale it according to your real traffic

Configuration Tips

- Set the maximum amount of memory available
- Put an amount of testing traffic
- Scale it according to your real traffic

Result

The number obtained represents about 30% of the memory you have to use to accomodate real-time traffic

Overview

Load Problem

- High resource consumption
- Poor performance
- Bad response time
- Bad customer experience

Load Troubleshooting

- top
  - CPU (idle, user, system)
- Statistics
  - opensipsctl fifo get_statistics load:
- NIC queues
  - netstat -lpn | grep opensips

```
# netstat -lpn | grep opens
tcp        0   6548 127.0.0.1:5060        0.0.0.0:*         LISTEN      30726/opensips
udp        0      0 127.0.0.1:5090        0.0.0.0:*                     30726/opensips
udp   131328      0 127.0.0.1:5060        0.0.0.0:*                     30726/opensips
```

Decreasing Machines Load

- Increasing the number of listeners
  - per interface
  - global
- TM timer partitions
  - `timer_partitions`

Overview

Thresholds

- Message processing time
  - `exec_msg_threshold`
- DNS response time
  - `exec_dns_threshold`
- TCP operations
  - `tcp_threshold`
- Database response time
  - `exec_query_threshold`

Thresholds

- Message processing time
  - `exec_msg_threshold`
- DNS response time
  - `exec_dns_threshold`
- TCP operations
  - `tcp_threshold`
- Database response time
  - `exec_query_threshold`

No magic value

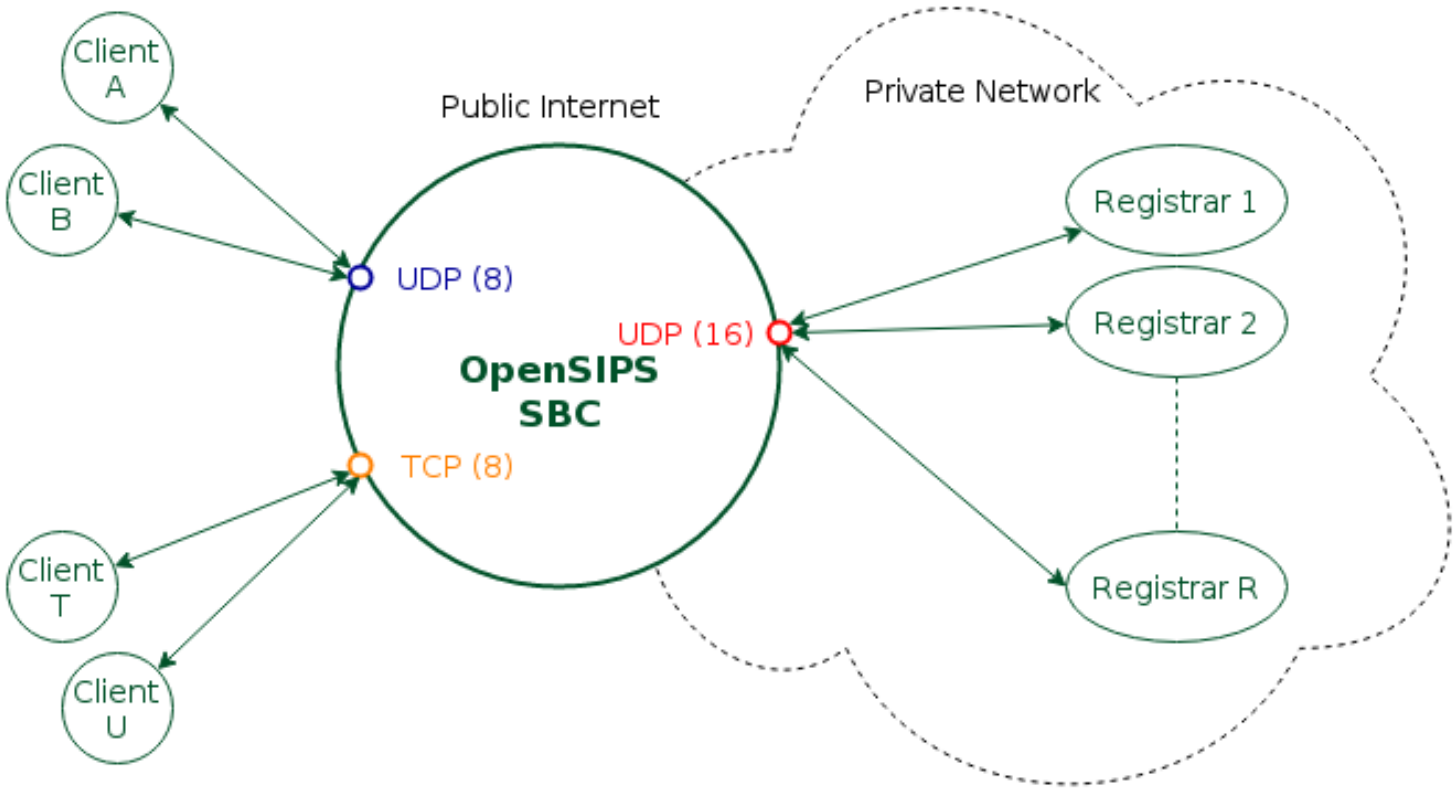
Tune it based on **your** platform!

Troubleshooting

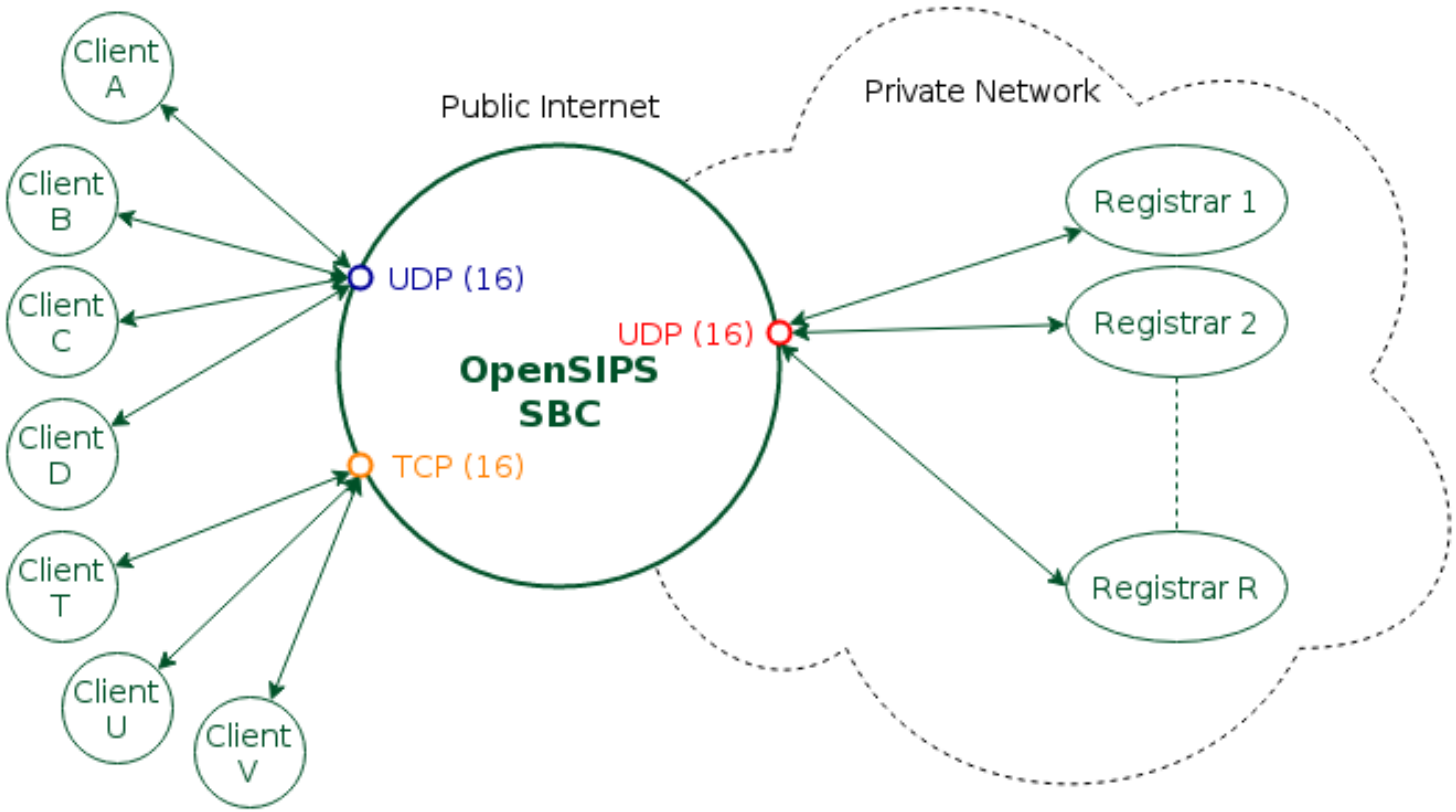
- Script Trace
  - `script_trace()` - logs how much time a function runs
- Benchmark Module
  - `bm_*` functions

Overview

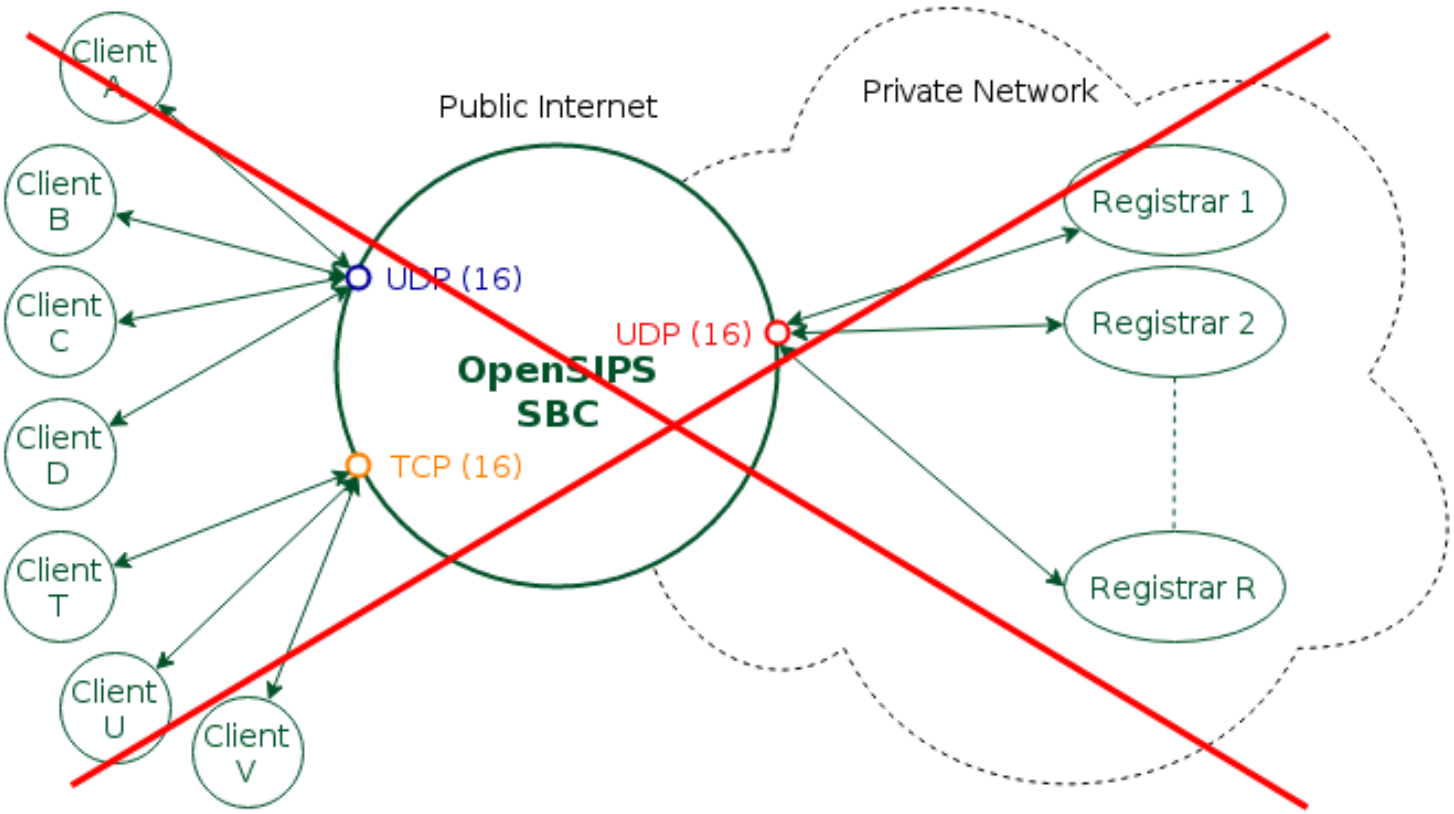
Topology


Client A
Client B
Client T
Client U
Public Internet
Private Network
UDP (8)
UDP (16)
OpenSIPS SBC
TCP (8)
Registrar 1
Registrar 2
Registrar R

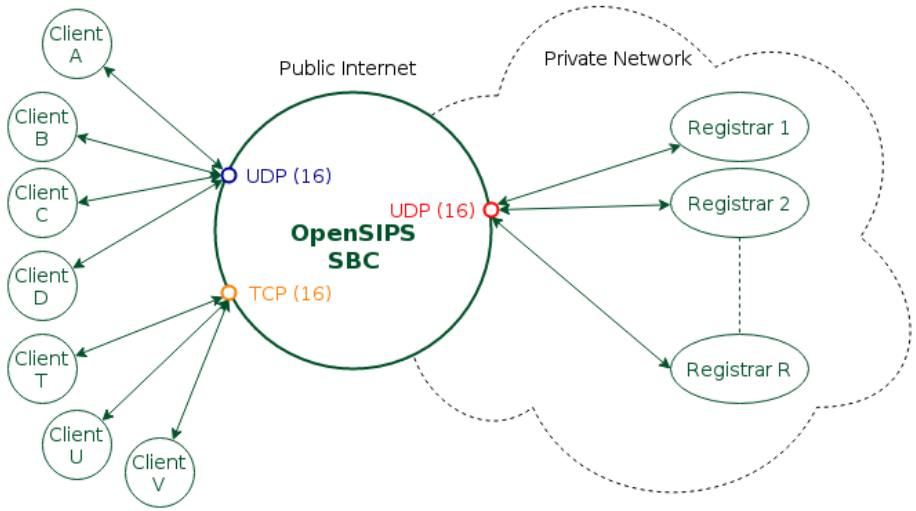
More customers



Service Down!

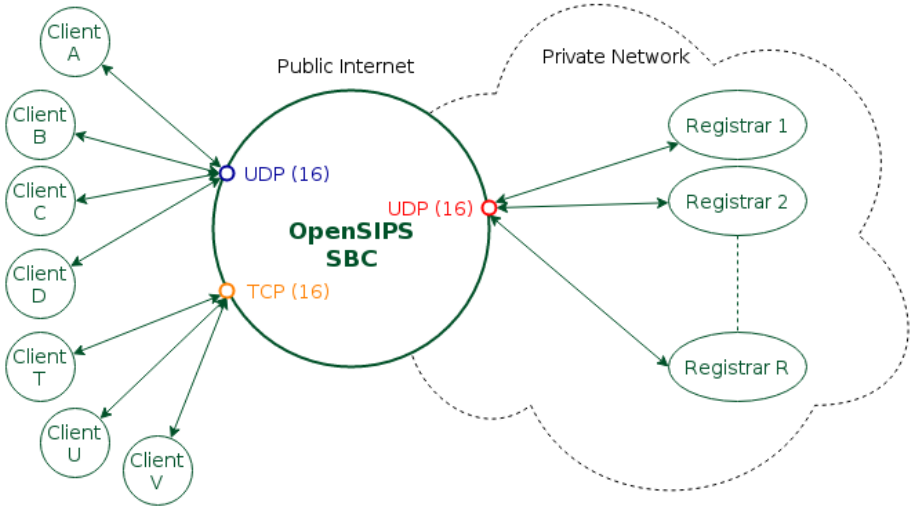


Troubleshoot



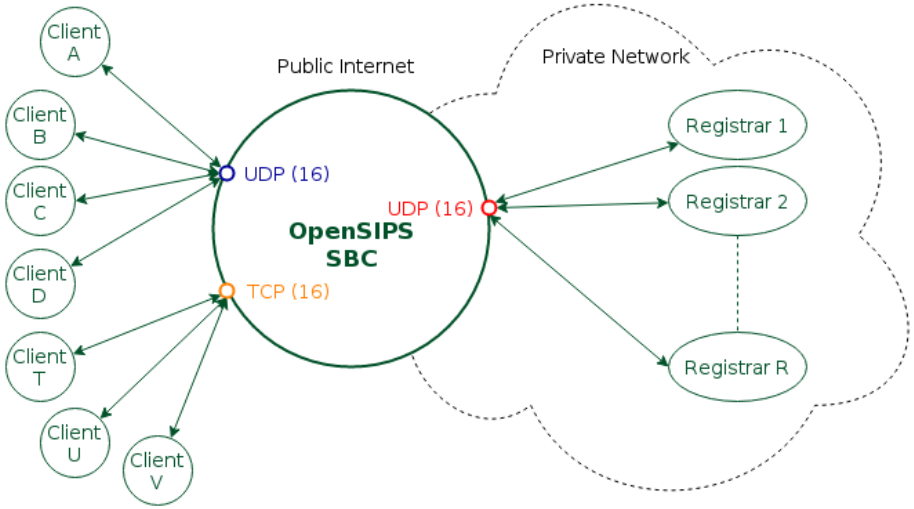
- `top`
  - CPU - 100%

Troubleshoot - continue



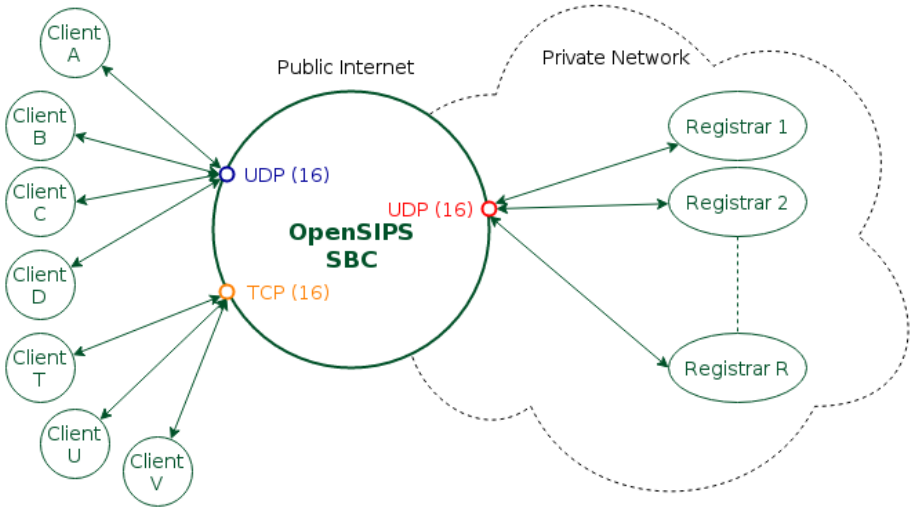
- Checked logs
  - Registrars timeout

Troubleshoot - continue



- `ngrep`
  - Registrars responding

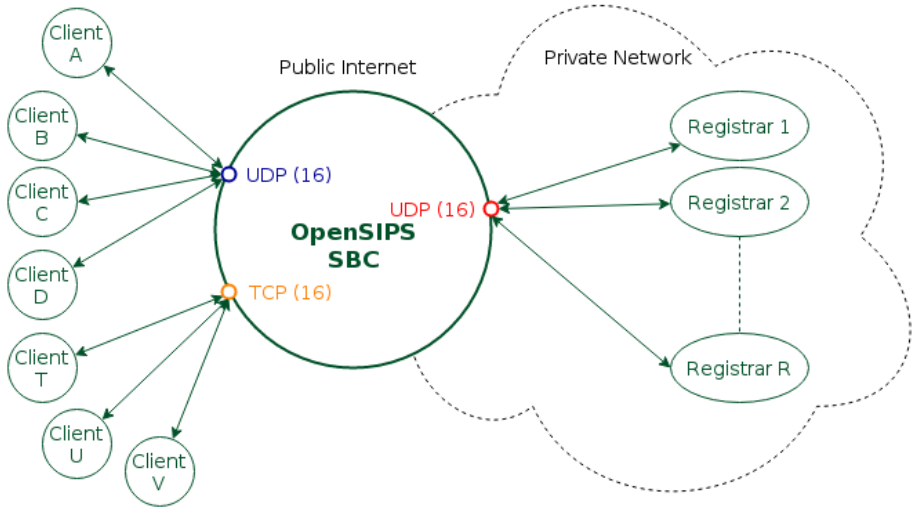
Troubleshoot - continue



- Statistics

```
# opensipsctl fifo get_statistics load:
load:udp:PUBLIC_IP:5060-load = 12
load:udp:PRIVATE_IP:5060-load = 100
load:tcp-load = 3
```

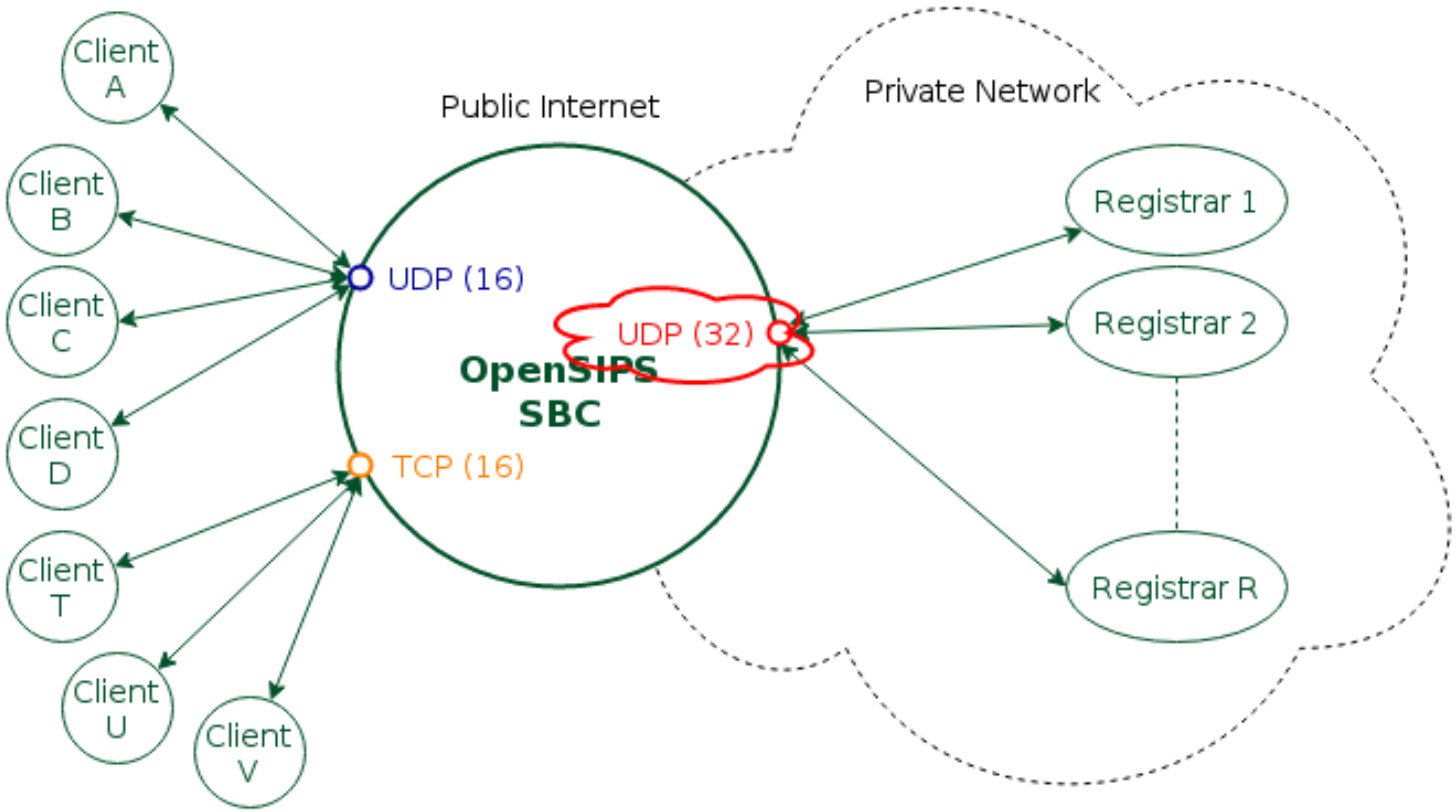
Troubleshoot - continue



- Network Queues

```
# netstat -lpn | grep opens
tcp        0      0 PUBLIC_IP:5060          0.0.0.0:*               LISTEN      30726/opensips
udp        0      0 PUBLIC_IP:5060          0.0.0.0:*                           30726/opensips
udp   131328      0 PRIVATE_IP:5060         0.0.0.0:*                           30726/opensips
```

Eureka!


Client A
Client B
Client C
Client D
Client T
Client U
Client V
Public Internet
UDP (16)
TCP (16)
UDP (32)
OpenSIPS SBC
Private Network
Registrar 1
Registrar 2
Registrar R

Overview

Conclusions

- There is no magic number you can generally use
- Test and stress your platform
- Understand your platform
- Tune your platform accordingly

Take-Away Message

Be more intimate to your platform!

- Răzvan Crainea
  - razvan@opensips.org

- Thank you!

- Răzvan Crainea
  - razvan@opensips.org