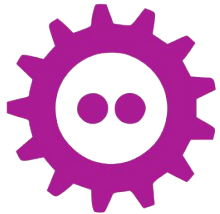


Real-Time Clustering with OpenSIPS



FOSDEM

3-4 Feb 2018

Răzvan Crainea
and
Liviu Chircu

- 4th Feb 2018 -



-
- Introduction
 - OpenSIPS cluster features
 - Distributed SIP user location
 - Conclusions

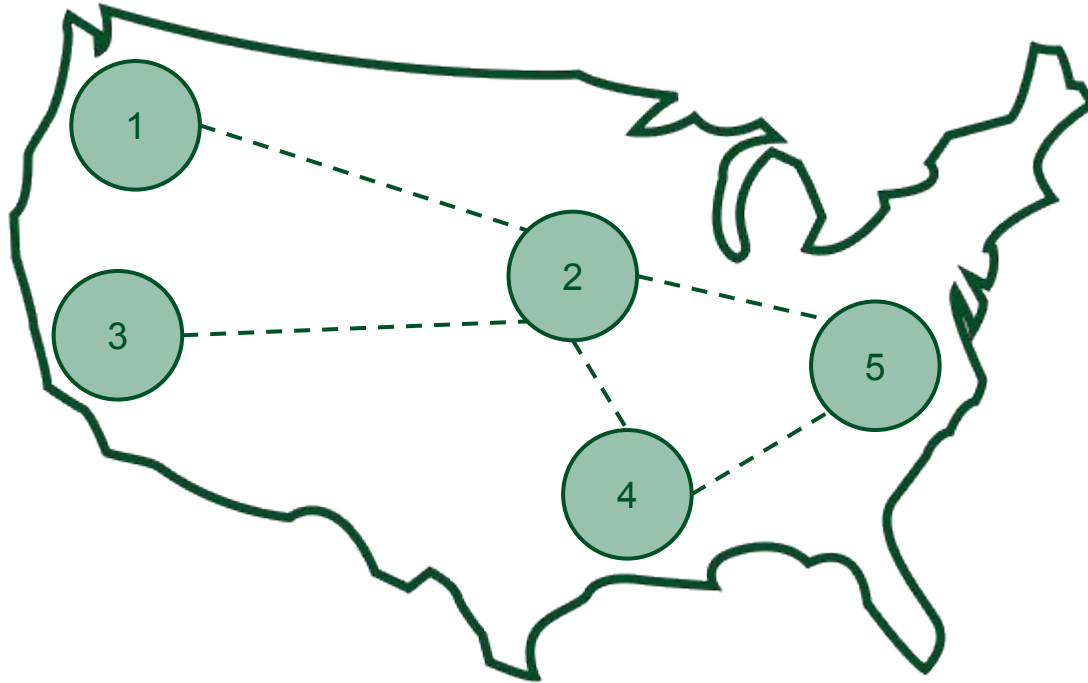
Introduction

- What is a cluster?
 - Multiple nodes with the same config that behave as a whole
- Why cluster OpenSIPS instances?
 - High availability
 - Scalability
 - Geographic distribution
- All nodes need to share data between them
- How to share data in a cluster?
 - Using a shared database
 - Directly between nodes using a custom communication channel

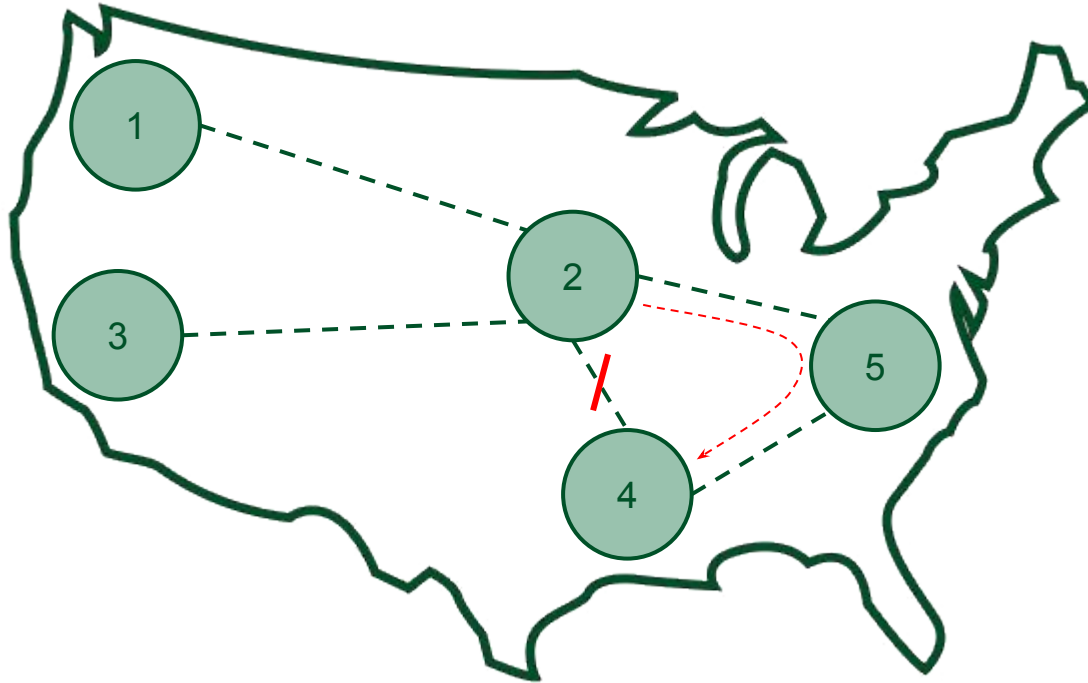
- Built on top of **proto_bin**
 - Efficient binary protocol
 - Built over TCP
 - Asynchronous
- Uses heartbeat packages
 - Maintain a valid view of the cluster
 - Packets sent by the application layer
 - Uses a “link-state” algorithm to ensure the global consistency of the entire cluster
- Simple
 - A node needs only one link to any of the nodes within the cluster

OpenSIPS Cluster Features

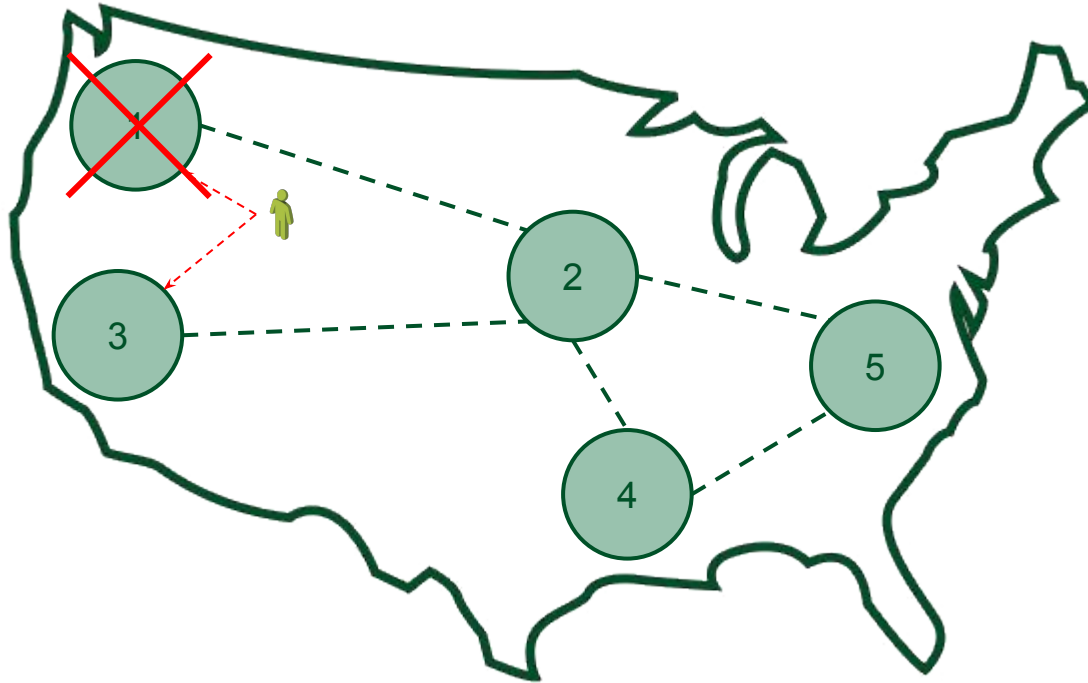
Diverse Topologies



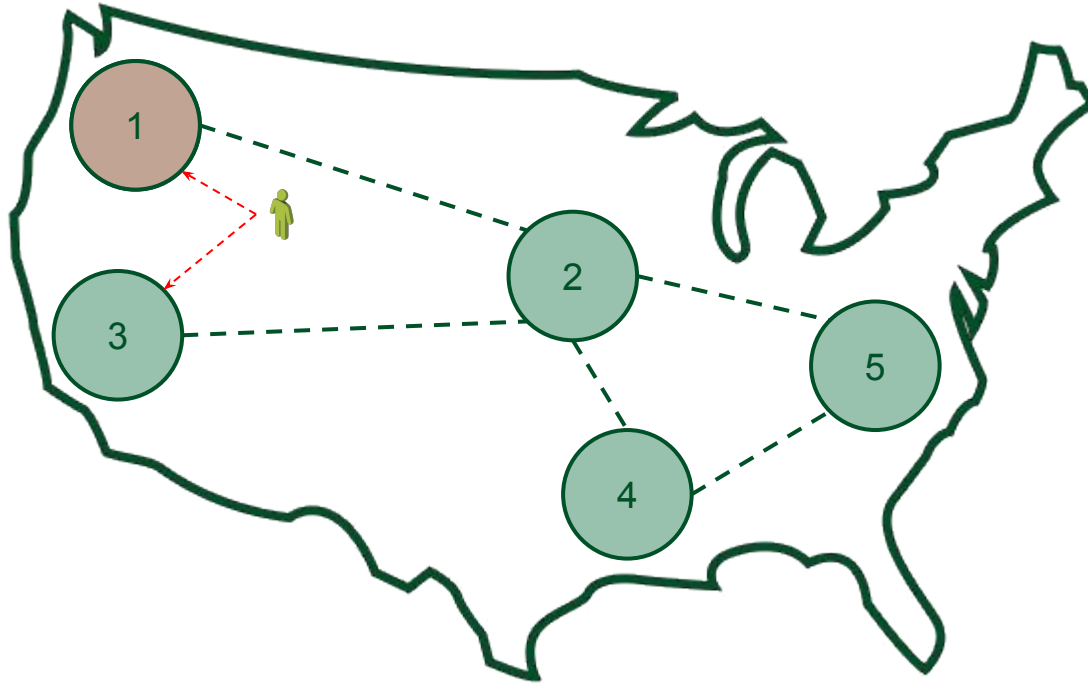
Link Redundancy



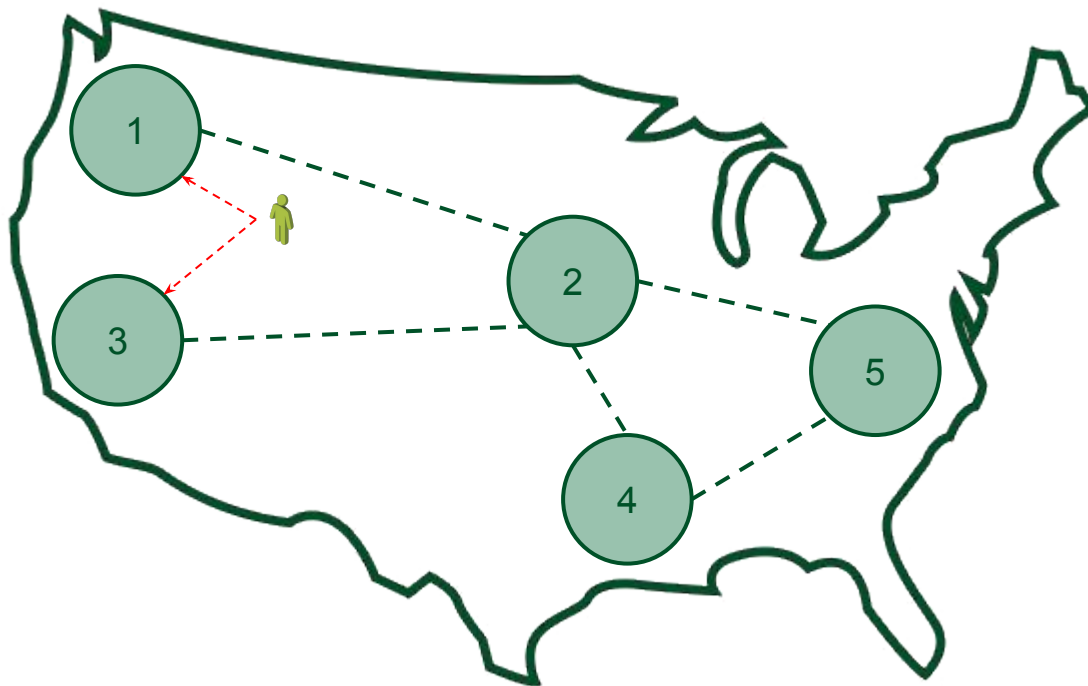
High Availability Failover



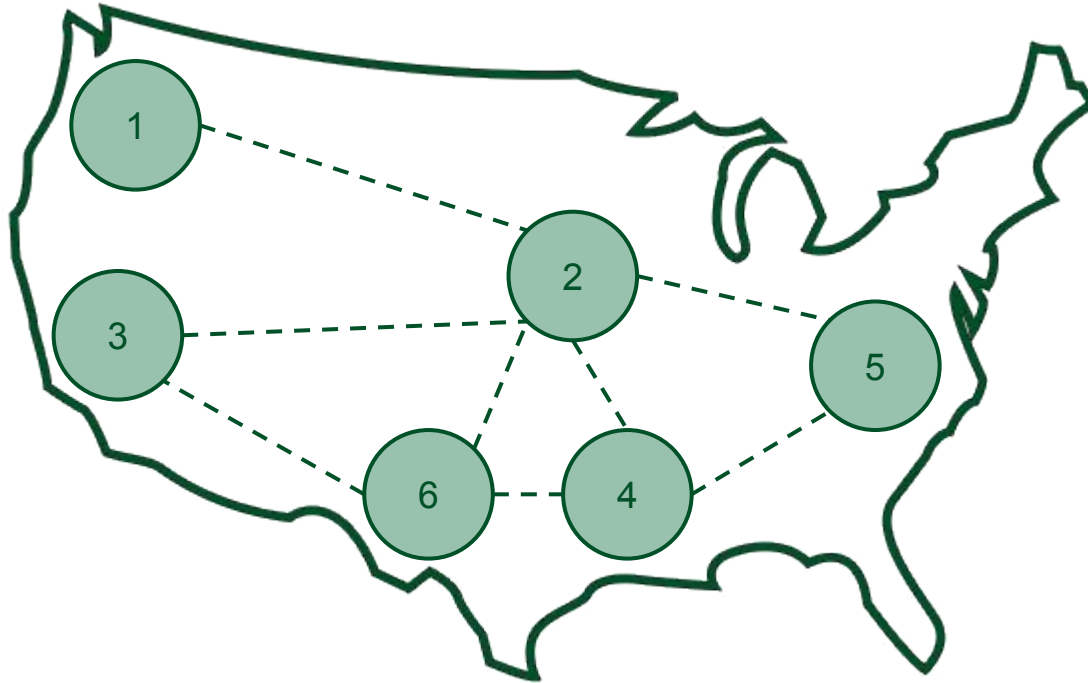
Anycast



Platform-wide CPS limiting



Self-Discovering Cluster



Inserting a new node in the cluster



- Find out network topology
 - fetch the topology from DB
 - connect to any node and exchange topology information
- The new node connects to all known nodes
 - fetched from DB or learned from the joining node
- The new node queries the cluster for a fully sync'ed node
- One of the fully sync'ed nodes start sending the entire data snapshot to the new node

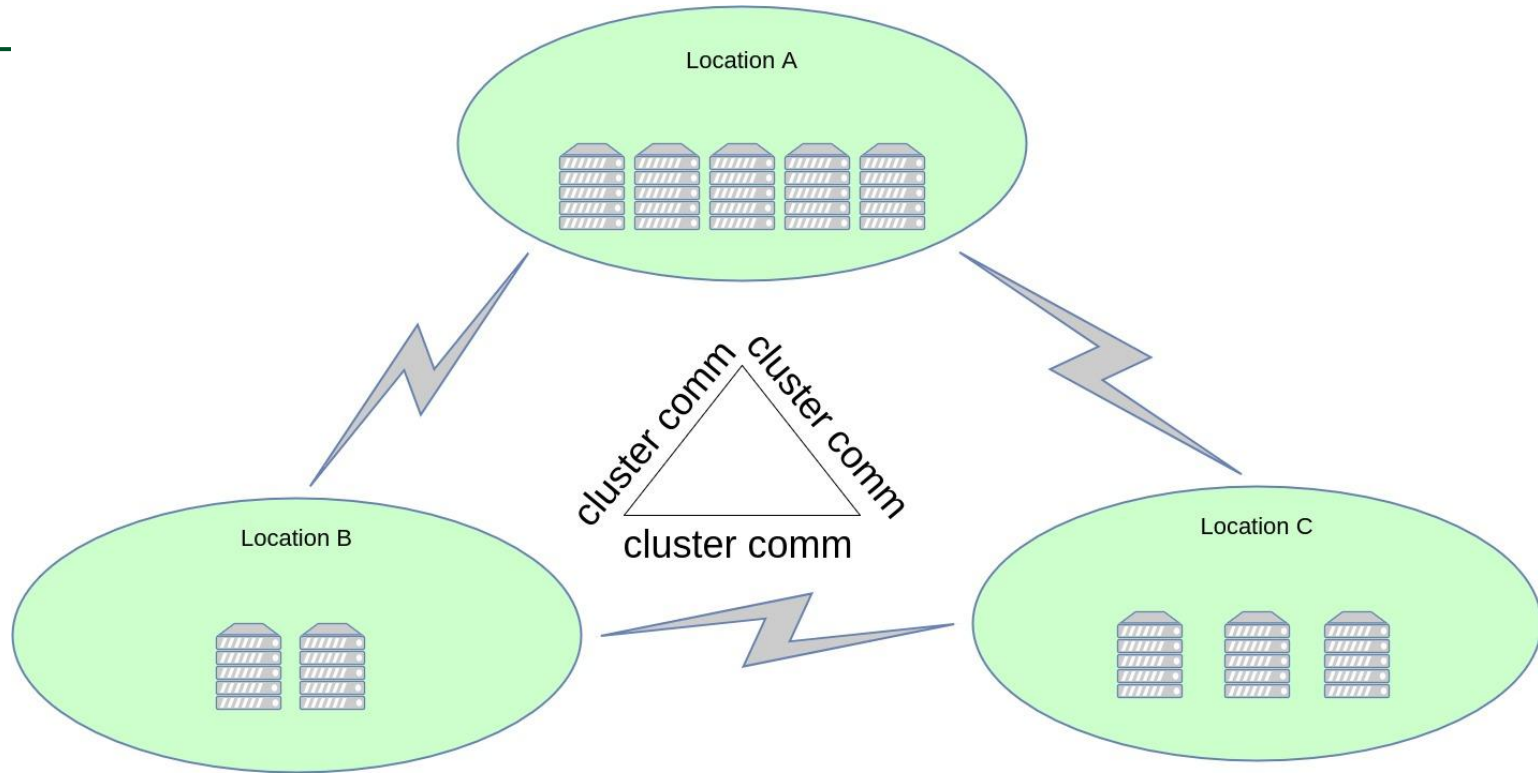
Distributed SIP User Location

Requirements



- Geo distribution
- Horizontally scalable
- Highly available
- NAT traversal
- NAT keepalives

“User facing” model



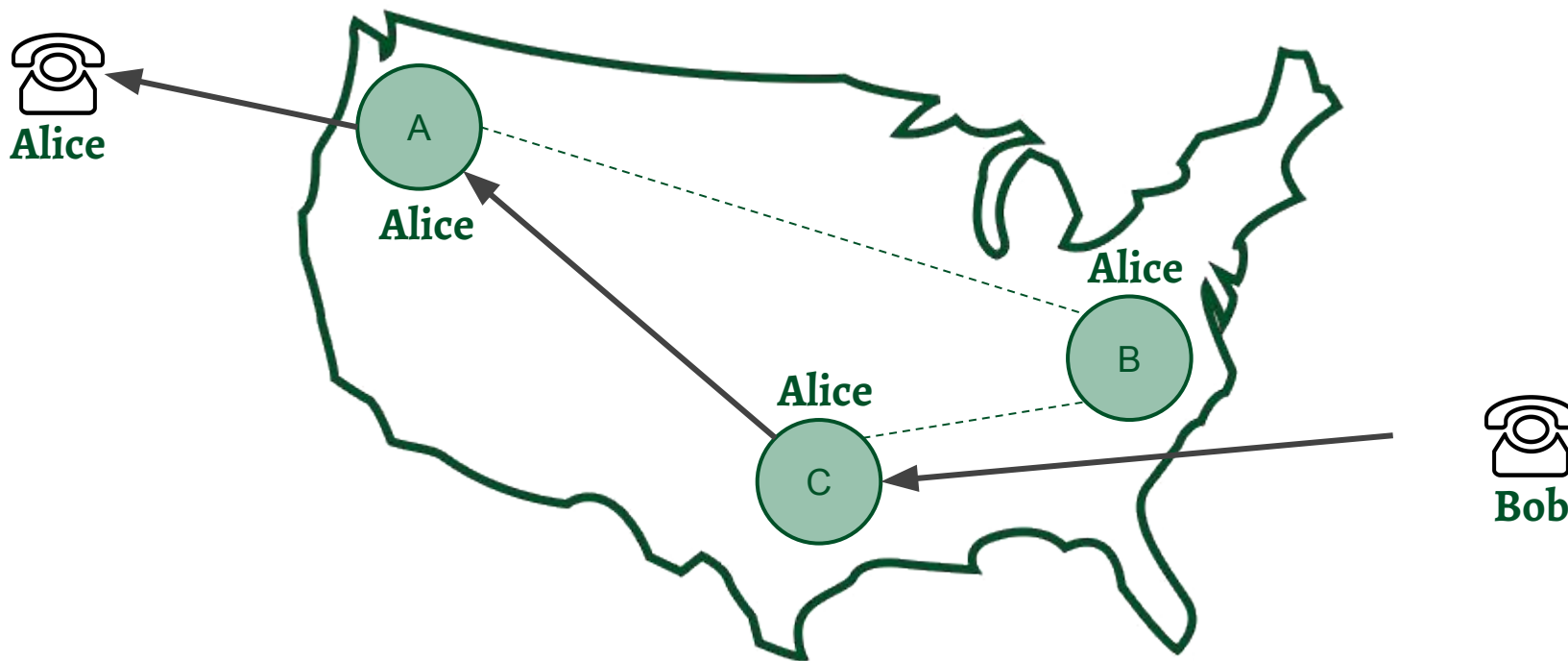
“User facing” model



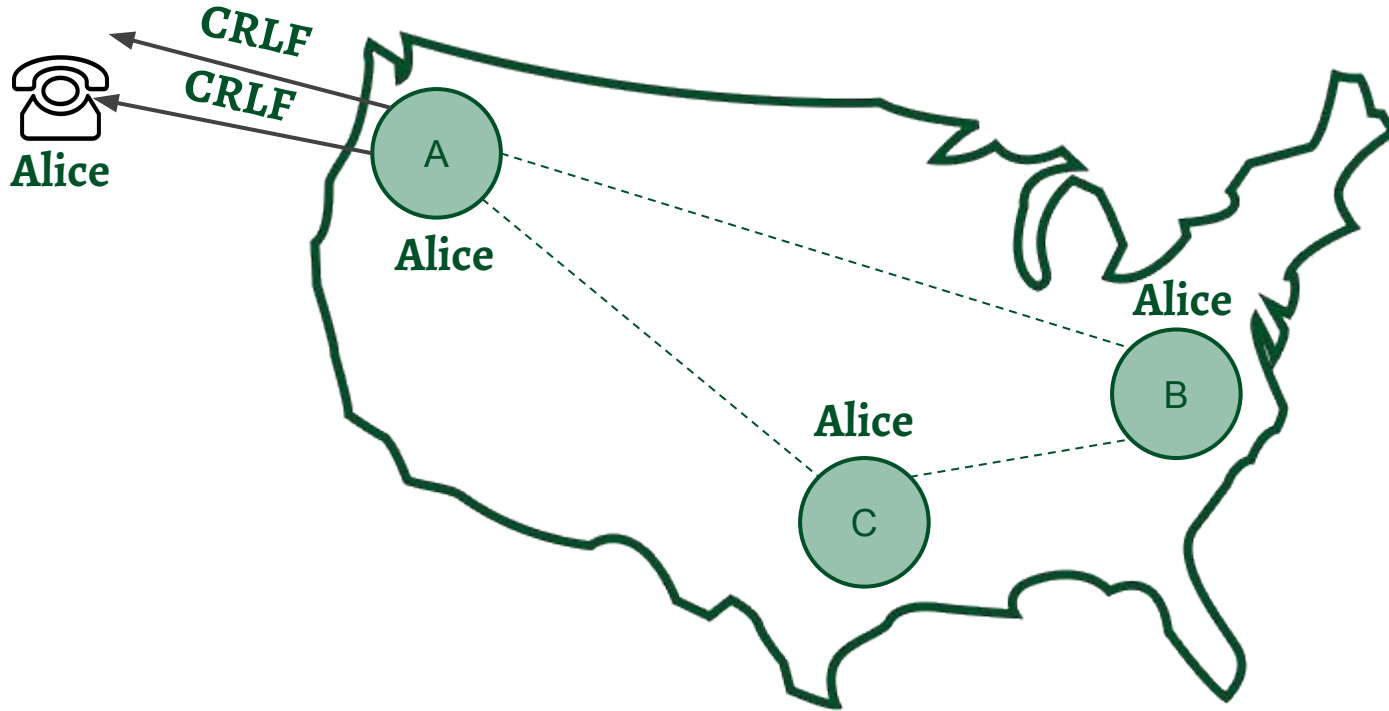
- metadata-driven
- on-the-fly platform resizing
- optimal for multiple locations
- **trivial** OpenSIPS scripting (`save()` / `lookup()`)

opensips.org/Development/Design-Distributed-User-Location

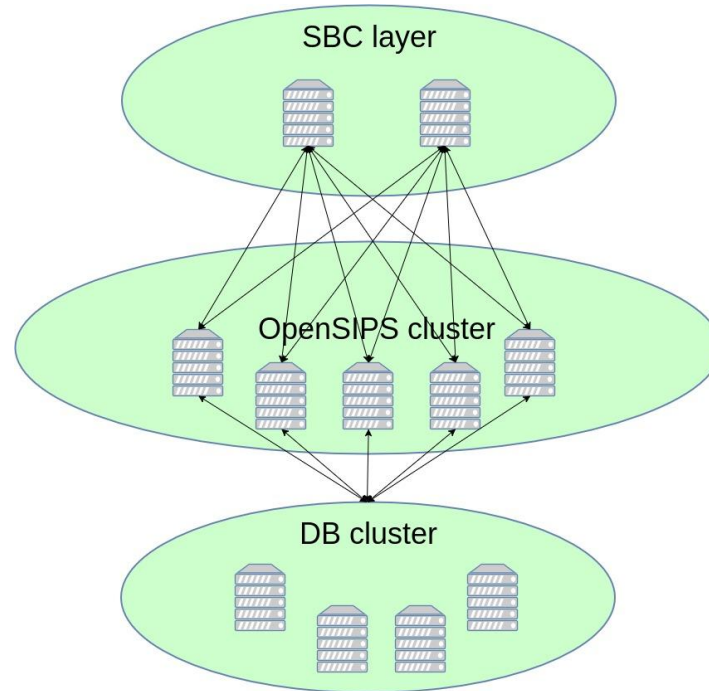
NAT traversal



NAT keepalives



“DB-driven” model



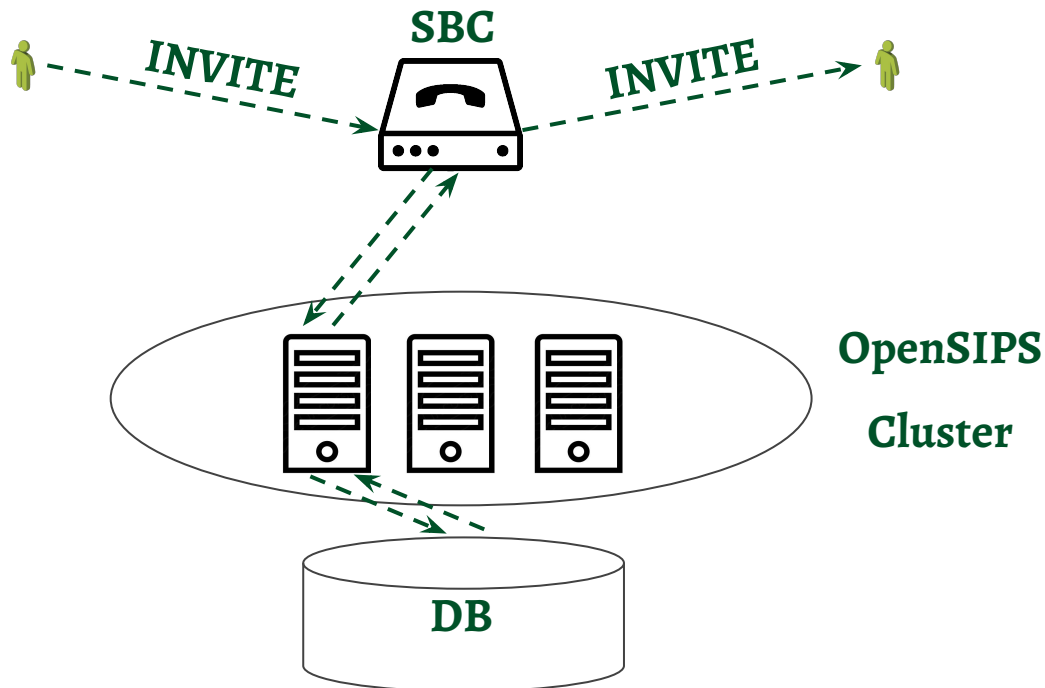
“DB-driven” model



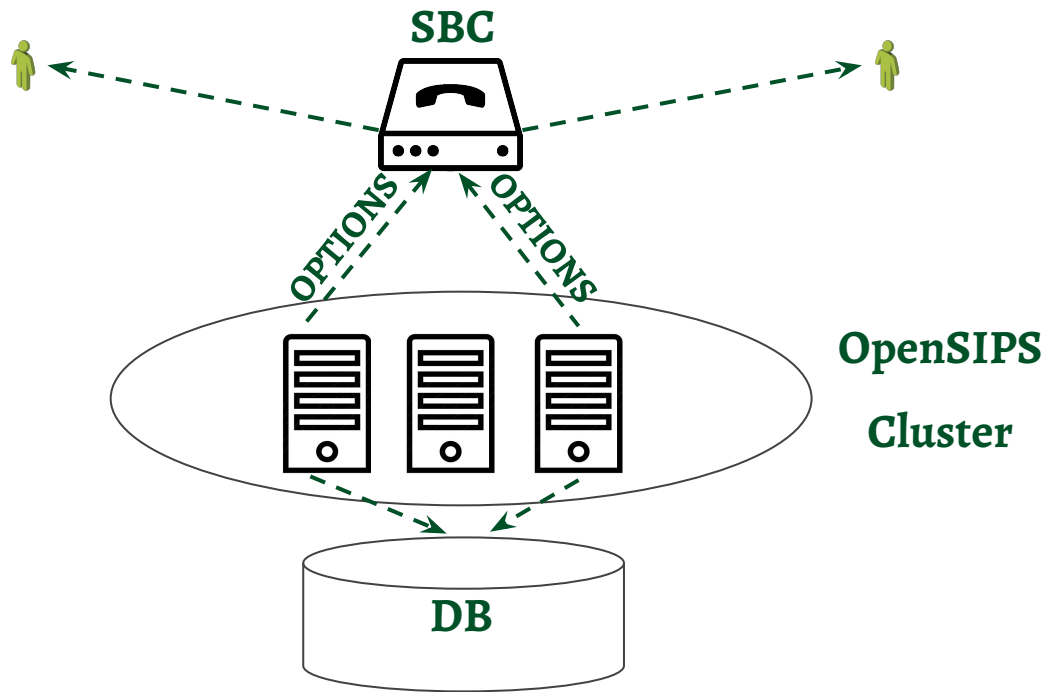
- DB cluster holds **all** user location data
- on-the-fly platform resizing (OpenSIPS / DB)
- optimal for single sites / high subscriber populations
- **trivial** OpenSIPS scripting (save() / lookup())

opensips.org/Development/Design-Distributed-User-Location

NAT traversal



NAT keepalives



Conclusions

OpenSIPS 2.4 - “clustering”



- distributed user location
- usrloc node sync on startup
- self-discovering cluster
- dialog sync on startup

Influence Development!



- IRC: #opensips on FreeNode
- Slack: #opensips
- Mailing list: users@opensips.org

opensips.org/Development/Design-Distributed-User-Location

Take-Away Message

OpenSIPS clustering - a community-driven design -

- Răzvan Crainea and Liviu Chircu
 - OpenSIPS Project: www.opensips.org
 - razvan@opensips.org, liviu@opensips.org



AMSTERDAM
OPENSIPS
SUMMIT 2018



**Open Source Weekend
Discount**

b3e737ac-9019-493f-8888-9f6c16af3bc9

**Save 50% for
Summit and Training**



- It's not an entry point in the cluster
 - Any node can be an entry point
- Acts as an arbiter when multiple clusters merge
 - Sorts the split-brain problem