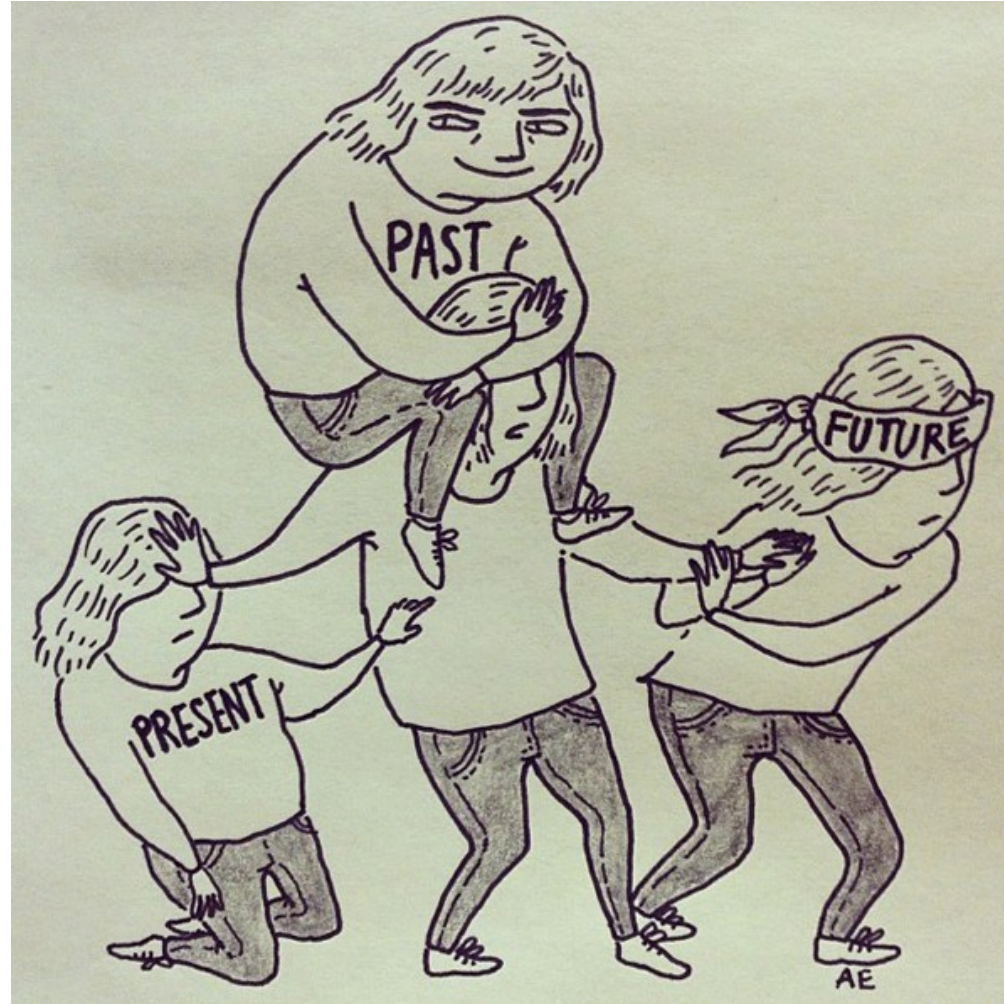


OpenSIPS Summit - Keynotes

Bogdan-Andrei Iancu
Founder OpenSIPS Project
OpenSIPS Solutions

Past, Present, Future



Present - OpenSIPS 2.2

Philosophy

- Consolidate existing features / functionalities
- First milestone for built-in clustering support

Extended Async

- RADIUS support
- LDAP support
- HEP / BIN support
- aysnc sleep()
- Consistent async support (variables, forking, context)

**"LACK OF
CONSISTENCY
CAN BRING ON A
LACK OF
INTEREST"**

Consolidate by consistency - protocols

- WebSocket Secure (WSS) support added
- HEP protocol as module
- BIN protocol as module

Consolidate by consistency - events

- Balancing and Failover support (event_virtual)
- Reliable events (event_flatstore)

Consolidate by consistency - memory

- Easier/uniform way of getting memory statistics
- Monitor memory usage per module
- Easier way to debug memory issues

Consolidate by consistency – SQL caching

- Built-in support for auto data caching
- Easy way of using it for any SQL table
- “ready to be used” concept
- sql_cacher module

Consolidate by consistency – accounting

- Bye, bye flags, welcome functions
- Easier to use and control
- No hidden behavior

Consolidate by consistency – registration state

- Keep track of the registration reachability
- Stateful probing to check
- Tighter integration between the usrloc and nathelper modules

Consolidate by consistency – SSL certificates

- New `tls_mgm` module to handle SSL certificates
- Used by TLS and WSS protocols
- DB driven provisioning

Consolidate by consistency – SIP capturing

- Improve flexibility and performance
- All chains addressed (siptrace, sipcapture, HEP)
- Opens new possibilities for HEP proxying

Perfect integration with



Clustering

The concept of putting together multiple OpenSIPS instances to act a single service.



Clustering

Building a cluster around a DB (in terms of sharing)
results into a bottleneck



Clustering

- Built-in support
- Generally available for all modules requiring clustering
- Based on BIN protocol (OpenSIPS-2-OpenSIPS)
- Used for sharing or replicating data

Future - OpenSIPS 2.3

More clustering

- Enhance the clustering support to be fault-tolerant (for nodes) and re-route traffic inside the cluster
- Hot plug-in for new nodes in the cluster
- Distributed User Location based on clustering; support also partitioning scenario.
- MI clustering support (propagate reload command in the entire cluster)

More async

- Add async support for more I/O operations, like DNS, noSQL databases
- Increase consistency at scripting level when comes to aysnc operations (variables lifetime, transactions state)
- “under the hood” enhancements of the async reactor for inter-process communication (ability to pipe a job/task to a certain process)

IMS support

- DIAMTER driver for the AAA interface
- Add support for I-CSCF, P-CSCF, S-CSCF nodes

Project EcoSystem

OpenSIPS, as SIP server, is a part of a large ecosystem of services and applications



Future - OpenSIPS 3.x

Config File re-work

- Re-structure the provisioning across multiple sections (routing logic to be separated from modules and global parameters)
- Easier module provisioning
- Script variable re-naming to help in understanding the scope of the variables

Config File re-work

```
loadmodule [uri] {  
    use_uri_table = 0  
}
```

```
loadmodule [tm] {  
    fr_timeout = 5  
    fr_inv_timeout = 30  
    restart_fr_on_each_reply = 0  
    onreply_avp_mode = 1  
}
```

Config File re-work

\$msg.from.uri.username instead of \$fU

\$dlg.val(my_var)

\$dlg.lifetime

\$trans.fr_timeout

Reload of routing logic

- Reload the OpenSIPS routing logic at runtime
- No package loss on network level
- No call / registration / transaction lost
- No downtime !

SIP message modification

- Drop the lumps support
- Apply changes over SIP message in realtime
- All changes must be at SIP-header level

External routing logics

- Be able to use an external application as routing script
- To interact with OpenSIPS SIP stack via a routing API
- Libraries to be provided in various languages (Python, Java, C++)

- The end result ?



Thank you for your attention
You can find out more at www.opensips.org
bogdan@opensips.org
www.opensips-solutions.com

Questions are welcome