OMEC

Project Overview

April 2019
Open Mobile Evolved Core (OMEC)

A project is intended to become an open source production grade Evolved Packet Core (EPC).

OMEC is built using an NFV (Network Function Virtualization) architecture

OMEC includes:

- Complete connectivity, billing and charging capabilities
- 3GPP Release 13 compatibility (more on this in a bit)
- Support for large numbers of subscribers with a high performance DPDK based data plane
- Optimization for lightweight cost effective deployments and IoT applications
- Integrated CI/CD test and verification capabilities

https://github.com/omec-project
“Visible” Projects

Single Frame (1 instance of each component)
40K Users
1K Control Plane TPS
42-80 CPU Cores

Yes, there is no connection from the PCRF to anything yet
Infrastructure / Support Projects

Infrastructure projects

- oss-util – CLI support
  - Uses RESTful interface
  - Provides local CLI callback to RESTful API
  - One source for configuration / command changes to watch

- Freediameter – fork of freediameter
  - Fixes
  - Performance improvements
  - **Release 14** Diameter interfaces

Test, CI/CD and deployment resources

- omec-project-ci
- deployment (terraform based tools)
- ci-test
- il_trafficgen – DPDK based S1U traffic simulator
Basic Functions

Default Bearers
Offline Billing
Child Protections (gating by domain or 5 tuple)
Basic MME support (initial attach, detach, etc.)
UPF DNS lookup per TS 29.244

Planned
- User Level Packet Copying (based on open Lawful Intercept for CUPS)
- SAEGW modes for roaming
- OAM (alarms, measurements, etc)
- Common CLI
- Dedicated Bearers
- Handover Scenarios
- Restoration
ngic-rtc Communication Modes

**Current**
- FPC Agent (ODL – not OMEC)
- Control Plane (CP)
- Data Plane (DP)

*Agent based or direct*

**Future**
- Control Plane (CP)
- Data Plane (DP)
- Sx

*Direct mode via Sx (aka CUPS or N4)*

FPC Agent is still an option

Direct mode uses a protocol over ZMQ
ngic-rtc deployment Roles

**Today**

- MME
- eNB
- SGW-C
- S11
- S5/S8-C
- PGW-C
- SGW-U
- S1-U
- PGW-U
- PDN(s)

**Under Development**

- Inbound Roamer (acts as SGW)
- Home Network
- Outbound Roamer (acts as PGW)
Continuous Integration / Continuous Delivery (CI/CD) process

Github (changes ready) → Pull request → Github CI/CD checks (e.g. license) → multiple CI/CD checks Including performance

Performance is a part of our CI/CD process

- il-trafficgen
- Other tools, e.g. ng4t

If you are concerned about user plane performance you can download and run il-trafficgen prior to submitting!
What Version are you at?

Current
Diameter – R14
GTP – R12
S1AP – tested against production eNodeBs – based on R10 with some R11/R12 features
S1-U – R12

Future
Sx - R15
GTP – R15
Gx – R14 with necessary AVPs for R15 (as required)
Deployment?

VM based
Container options as well

Terraform scripts are out there!
Where can you contribute?

Documentation
Bugs submissions
Test, test, test!

Functions
MME will always need work
PCRF – current one is basic