LF Edge Akraino Project Presentation

Ike Alisson
LF Edge Akraino TSC member &
LF Edge Akraino Documentation Sub-committee TSC Chair
November 24th, 2021
Table of Contents

1. LF Edge Akraino Project
   1.1 Akraino Overview
   1.1.1 LF Edge TAC Stage 3 Project
   1.1.2 Akraino Integration Projects (Blueprints-BPs) Lifecycle and Review Phases
   1.1.3 Akraino BPs Rel. 5 and Rel. 4 Overview
   1.1.4 Akraino Analytics and Performance Metrics - example
   1.1.5 Akraino TSC Sub-committees
      1.1.5.1 Akraino Security, SW Validation (BluVal - Blueprint Validation) & API Sub-committees
      1.1.5.2 Akraino IoT Area

2. LF Edge Akraino Contact for future inquiries and information
1.1. LF Edge Akaino Project Overview

- 20 < Blueprints (aka Integration Projects), BPs Proposals & Development Projects
  - set of Open Infrastructures & Application Blueprints (BPs)

- Coordination & Co-operation with Multiple Upstream Open Source Communities/SDOs as:
  - Airship,
  - LFN Anuket
  - OpenStack,
  - ONAP,
  - ETSI MEC,
  - GSMA,
  - TIP,
  - CNCF
  - O-RAN

Objective: To deliver a fully integrated stack
- LF Edge Stages - Definitions & Expectations
  Every Foundation Project has an associated Maturity Level, as voted on under the approved Project Lifecycle Document (PLD) Process.

- Projects of all maturities have access to Foundation Resources.

- Stage 3: Impact Stage ('Top-Level') Definition
  - Projects that have reached their Growth Goals and are now on a Self-sustaining Cycle of Development, Maintenance, and Long-term Support.

Impact Stage projects are widely used in Production Environments and have Large, Well-established Project Communities with a number of Contributors from at least two (2) Organizations.
1.1.2 Akraino Project (Blueprint) Lifecycle States and Reviews phases

- Five (5) states that Projects goes through.

- A Project Lifecycle may extend across Multiple Projects and Akraino Releases.

- The Procedure of moving from one(1) State to the next one is independent from the Akraino Release Lifecycle and the pace depends on each individual Project.

- In order to effectively review Project progress, four (4) Reviews are built-in to the Project Lifecycle, namely,

  1. Proposal,
  2. Incubation,
  3. Mature,
  4. Core
  5. Archived
1.1.3 LF Edge Akraino Project Overview - R5 & R4 BPs
1.1. 4 LF Edge Akraino Project Analytics - Commits by Contributors and Companies - 1
1.1.4 LF Edge Akraino Project Analytics - Commits by Contributors and Companies - 2
1.1.5 Akraino Project TSC Sub-committees

Akraino TSC Sub-Committees

Subcommittees

- API Sub-committee
- CI, Blueprint Validation Lab sub-committee
- Documentation Sub-committee
- Outreach Sub-committee
- Process, Project review and recommend, documentation sub-committee
- Security Sub-committee
- Technical Community Sub-committee
- Upstream & Downstream Sub-committee
- TAC
- TAC Level Activities
In a stand-out finding of interest to our Telco Customers, 95% of respondents from the Telecommunications Industry report using Open Source.

The high-level takeaway of the report is that: "Using Open Source SW across all Industries is no longer principally about making best use of IT Budgets.

*Lower Cost of Ownership has fallen off the top spot and now sits in sixth (6th) position.*

Today, the Strategic Benefits of using Open Source are valued more, including:

### Top benefits of using enterprise open source

1. Higher quality software **35%**
2. Access to latest innovations **33%**
3. Better security **30%**
4. Ability to safely leverage open source technologies **30%**

### Top benefits of using enterprise open source

<table>
<thead>
<tr>
<th>Region</th>
<th>Benefit</th>
<th>U.S.</th>
<th>EMEA</th>
<th>APAC</th>
<th>LATAM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>35%</td>
<td>35%</td>
<td>38%</td>
<td>35%</td>
</tr>
<tr>
<td></td>
<td>Higher quality software</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>33%</td>
<td>33%</td>
<td>33%</td>
<td>34%</td>
</tr>
<tr>
<td></td>
<td>Access to latest innovations</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>32%</td>
<td>31%</td>
<td>30%</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ability to safely leverage open source techs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Akraino Edge Stack
Security Sub-Committee

September 24, 2021

Daniil Egranov
Security Sub-Committee Co-Chair, Akraino

Randy Stricklin
Security Sub-Committee Chair, Akraino
Akraino Security Team 2021 Accomplishments

› Automated Lynis, Vuls and Kube-Hunter Log Output Pass/Fail Analysis

› Lynis – Reviewed Required Tests
  › Formalized and Documented Lynis Incubation vs Maturity Requirements

› Platform Security for Akraino Blueprints
  › Arm
  › x86

› Release 4 and 5 Blueprint Reviews
Akraino Security Team Future Plans

› Develop Minimum OS Version Support Document
  › Ubuntu, CentOS, RHEL CoreOS, Debian

› Develop Minimum Security Tool Version Support Document
  › Lynis, Vuls, Kube-Hunter, and OVAL (Vuls) database

› BluVal (Blueprint Validation):
  › Integrate Automated Lynis, Vuls and Kube-Hunter Pass/Fail
  › Enforce minimum versions of Vuls, Lynis and Kube-Hunter

› Version 1.0 Platform Security Whitepaper

› Investigate using LFX Security
ETSI MEC Update of the collaborations with OpenSource, with special focus on LF Edge Akraino

Presented by: Jane Shen, VP of Technology Strategy, Mavenir
Technical Expert, ETSI MEC ISG

© ETSI 2021

15.11.2021
An API Portal For Edge Developers

- An API info hub of Akraino projects
- Cross reference with other relevant API information sites, e.g. ETSI MEC wiki, forge.etsi.org etc.
- Highlight API offerings from Akraino projects

https://apiportal.Akrai.n.org

© ETSI 2021
An API Map For Akraino Project APIs

From LF Edge Akraino
Number of publicly known "IoT Platforms" (2015-2019)

Source(s): IoT Analytics Research
Number of Identified IoT Platforms – By industry (Dec 2019)

**Consumer IoT**

- Home: 18%
- Mobility: 10%
- Lifestyle: 8%
- Health: 8%
- Manufacture / Industrial: 34%
- Energy: 32%
- Mobility: 31%
- Smart Cities: 23%
- Health: 21%
- Supply Chain: 19%

**Business IoT**

- Retail: 50%
- Agriculture: 13%
- Public Sector & Services: 12%
- Smart Building: 9%
- Telecom: 7%
- Finance: 7%
- Enterprise: 2%
- Other Business: 36%

**Note:** Percentages do not add up to 100% as most companies focus on several segments

**Source:** IoT Analytics 2020 List of IoT Platform Companies, n=620

IoT Platforms Competitive Landscape & Database 2020

AKRAINIO

Copyright © 2019 by www.iot-analytics.com All rights reserved
1.1.5.1 Akraino IoT Area

https://wiki.akraino.org/display/AK/IoT+Area

IoT Area

Skapad av Tina Tsou, senast ändrad av Ike Alisson den 07, 2021

**Blueprint Families**

ELIOT: Edge Lightweight and IoT Blueprint Family

IIoT at the Smart Device Edge (family)

oneM2M IoT Service Layer (SL) Platform

Project Cassini - IoT and Infrastructure Edge Blueprint Family

**Security**

PARSEC, the opensource CNCF project has been adopted for edge deployments as it offers a common API that abstracts secure roots of trust which are required to protect devices outside of the datacenter. This enables the cloud native principle of being able to freely move your applications from one platform to another while maintaining level of security that was not possible in the past.

Building on this abstraction, PARSEC can mediate access to hardware security primitives and create isolated key stores for a multi-tenancy environment.

Starting your project with the right platform for security will accelerate your deployments and scale.

Come read about Parsec at: https://parallaxsecond.github.io/parsec-book

And talk to the experts during our weekly community calls (see github).

Or join us on the CNCF slack channel: https://cloud-native.slack.com
ELIOT – Overview

ELIOT is a project under the Akraino approved blueprint family. It intends to develop a fully integrated edge network infrastructure and running edge computing applications on lightweight Edge Nodes.

ELIOT targets on making the edge node a lightweight software stack which can be deployed on resource constraint edge devices like IoT-Gateway and uCPE, by leveraging lightweight OS, container running environment and container orchestration applications.

ELIOT BP family target 2 use case:
- IoT gateway
- SD-WAN, WAN edge, uCPE
Akraino Blueprint: Smart Cities

The purpose of Smart Cities blueprint is to provide edge computing platform base on Arm Soc, Improve deployment flexibility and security in the edge computing. The high-level relationship between the functional domains is shown in the figure below:

Figure 1. Smart Cities Functional Domains.
oneM2M IoT Service Layer (SL) Platform

oneM2M Overview

The oneM2M Global Organization creates Technical Specifications (TSs) to ensure that Machine-to-Machine (M2M) Communications can effectively operate on a Worldwide scale.

Seven (7) of the World's leading Information and Communications Technology (ICT) Standards Development Organizations (SDOs) launched in July 2012 a new Global Organization to ensure the most efficient Deployment of Machine-to-Machine (M2M) Communications Systems.

The new organization, called oneM2M, develops specifications to ensure the Global Functionality of M2M—allowing a range of Industries to effectively take advantage of the benefits of this emerging Technology.

The seven (7) majors ICT SDO founders of oneM2M are:

- The European Telecommunications Standards Institute (ETSI), Europe
- The Association of Radio Industries and Businesses (ARIB), Japan
- The Telecommunication Technology Committee (TTC), Japan
- The Alliance for Telecommunications Industry Solutions (ATIS), USA
- The Telecommunications Industry Association (TIA), USA
- The China Communications Standards Association (CCSA), China
- The Telecommunications Technology Association (TTA), Korea

The members of the organization are devoted to developing Technical Specifications and Reports to ensure M2M Devices can successfully communicate on a Global scale.

The oneM2M Standardization work is split in five (5) WG:
1.1.5.1 Akraino IoT Area

OPC UA Standard IEC 62 541 for Open Platform Communication Unified Architecture

OPC UA (Open Platform Communication Unified Architecture) provides the necessary infrastructure for interoperability across the Enterprise, from Machine-to-Machine (M2M), Machine-to-Enterprise (M2E) and everything in-between.

The OPC UA was initially released in 2006 - 2008 and has a very broad Market deployment footprint since then. OPC UA specifies a Platform independent Service-oriented Architecture, that integrates all the functionality of the individual OPC Classic Specifications into one (1) extensible Framework.


The current version of the OPC UA specification is on 1.04 (22 November 2017). The new version of OPC UA has added Publish/Subscribe in addition to the Client/Server communications infrastructure.

The OPC UA Information Model is a so-called Full Mesh Network based on nodes. The OPC UA Architecure supports two (2) Protocols. This is visible to Application programmers only via changes to the URL. The binary protocol is opc:tcp://Server and http://Server is for Web Service. Otherwise OPC UA works completely transparent to the API.

After the initial release in 1996, the OPC Foundation was created to maintain the Standard. As OPC has been adopted beyond the field of Process Control, the OPC Foundation changed the name to Open Platform Communications in 2011. The change in name reflects the Applications of OPC Technology for Applications in Building Automation, Discrete Manufacturing, Process Control and many others. OPC has also grown beyond its original OLE (Object Linking and Embedding) implementation to include other Data transportation Technologies including Microsoft's .NET Framework, XML, and even the OPC Foundation's binary-encoded TCP format.

The OPC UA Multi-Layered approach accomplishes the original design specification goals of:

- **functional equivalence**: all COM OPC Classic specifications are mapped to UA
- **Platform independence**: from an embedded micro-controller to cloud-based infrastructure
- **Secure**: encryption, authentication, and auditing
- **Extensible**: ability to add new features without affecting existing applications
- **Comprehensive information modeling**: for defining complex information

**Functional Equivalence**

Building on the success of OPC Classic, OPC UA was designed to enhance and surpass the capabilities of the OPC Classic specifications. OPC UA is functionally equivalent to OPC Classic, yet capable of much more.
For further information on the OPC UA, please see attached below the OPC UA Open IEC 62 541 (current) Documentation from Jan 2021.
1.1.5.1 Akraino IoT Area - 5 (OPC UA IEC 62 541)

2.1 Building the Library

2.1.1 Building with CMake on Ubuntu or Debian

sudo apt-get install gcc
build-essential
python2.7-config

# enable additional features
sudo apt-get install cmake-curses-gui
for the cmake graphical interface

# encryption support
sudo apt-get install libssl-dev

# unit tests
sudo apt-get install check

# documentation generation
sudo apt-get install python-sphinx

# documentation style

od open62541
mkdir build

cd build

cmake ..

# select additional features

cmake ..

# build documentation

make doc

# html documentation

make doc html

# pdf documentation (requires LaTeX)

make doc pdf

2.1.2 Building with CMake on Windows

Here we explain the build process for Visual Studio (2013 or newer). To build with MinGW, just replace the compiler section in the call to CMake.

- Python 2.7.x (Python 3.x works as well): https://python.org/downloads
- CMake: http://cmake.org/cmake/resources/software.html

2.3 Building the Examples

Make sure that you can build the shared library as explained in the previous steps. Even easier way to build the examples is to install open62541 in your operating system (see following open62541).

Then the compiler should automatically find the includes and the shared library.

cp /path/to/examples/tutorial_server_firststeps.c . # copy the example server
gcc -std=gnu99 -o server tutorial_server_firststeps.c -open62541

2.4 Building for specific architectures

The open62541 library can be built for many operating systems and embedded systems. This document shows a small excerpt of already tested architectures. Since the stack is only using the C99 standard, there are many more supported architectures.

A full list of implemented architecture support can be found in the arch folder.

2.4.1 Windows, Linux, MacOS

These architectures are supported by default and are automatically chosen by CMake.

Have a look into the previous sections on how to do that.

2.4.2 freeRTOS + LwIP

Cefitto @chablontinas

This documentation is based on the discussion of the PR: https://github.com/open62541/open62541/pull/2511. If you have any doubts, please first check the discussion there.

3.2 Prebuilt packages

3.2.1 Pack branches

GitHub allows you to download a specific branch as .zip package. Just using this .zip package for open62541 will likely fail:

- CMake uses git describe --tags to automatically detect the version string. The .zip package does not include any git information
- Specific options during the build stack require additional git submodules which are not included in the .zip

Therefore we provide prepackaging branches. They have the prefix pack and are automatically updated to match the referenced branch.

Here are some examples:

- pack/master.zip
- pack/1.0.zip

These pack branches have refined submodules and the version string is hardened. If you need to build from source, but do not want to use git, use these pack specific versions.

3.2.2 Prebuilt binaries

You can always find prebuilt binaries for every release on our Github Release Page.

Nightly single the releases for Linux and Windows of the last 50 commits can be found here:

https://open62541.org/releases

3.2.3 Debian

Debian packages can be found in our official PPA:

- Daily Builds (based on master branch): https://launchpad.net/~open62541-team/+archive/ubuntu/daily
- Release Builds (starting with Version 0.4): https://launchpad.net/~open62541-team/+archive/ubuntu/ppa

Install them with:

- sudo add-apt-repository ppa:open62541-team/ppa
- sudo apt-get update
- sudo apt-get install libopen62541-1-dev

3.2.4 Arch

Arch packages are available in the AUR:

- Stable Builds: https://aur.archlinux.org/packages/open62541/
- Unstable Builds (current master): https://aur.archlinux.org/packages/open62541-git/

In order to add custom build options (Build Options), you can set the environment variable

OPEN62541_CMAKE_FLAGS
1.1.5.1 Akraino IoT Area

oneM2M IoT Service Layer (SL) Platform

oneM2M Overview

The oneM2M Global Organization creates Technical Specifications (TSs) to ensure that Machine-to-Machine (M2M) Communications can effectively operate on a Worldwide scale.

Seven (7) of the World’s leading Information and Communications Technology (ICT) Standards Development Organizations (SDOs) launched in July 2012 a new Global Organization to ensure the most efficient Deployment of Machine-to-Machine (M2M) Communications Systems.

The new organization, called oneM2M, develops specifications to ensure the Global Functionality of M2M—allowing a range of Industries to effectively take advantage of the benefits of this emerging Technology.

The seven (7) majors ICT SDO founders of oneM2M are:

- The European Telecommunications Standards Institute (ETSI), Europe
- The Association of Radio Industries and Businesses (ARIB), Japan
- The Telecommunication Technology Committee (TTC), Japan
- The Alliance for Telecommunications Industry Solutions (ATIS), USA
- The Telecommunications Industry Association (TIA), USA
- The China Communications Standards Association (CCSA), China
- The Telecommunications Technology Association (TTA), Korea

The members of the organization are devoted to developing Technical Specifications and Reports to ensure M2M Devices can successfully communicate on a Global scale.

The oneM2M Standardization work is split in five (5) WG:
1.1.5.1 Akraino IoT Area

- ELIoT: Edge Lightweight and IoT Blueprint Fam
- IIoT at the Smart Device Edge (family)

**oneM2M IoT Service Layer (SL) Platform**

- 3GPP 5G HMTC (High Performance Machine)
- oneM2M Cloud Vendor Independent & ETSI
- oneM2M IoT SL and AI/ML use
- oneM2M IoT SL and CIM NGSI-LD (Context I
- oneM2M IoT SL Architecture
- oneM2M IoT SL Common Service Functions
  - oneM2M IoT SL pre-integrated with 5G (3GF
  - oneM2M IoT SL Release Roadmap
  - oneM2M Semantic enablement and ASD (A
  - oneM2M Test Suite Structure (TSS) and Test I
  - oneM2M Use Cases (UCs) and SAREF (Smart
  - OPC UA Standard IEC 62 541 for Open Platf
- Project Cassini - IoT and Infrastructure Edge Bl
- Tami COVID-19 Blueprint Family
- Automotive Area
- Blueprint Proposals
- Akraino Feature Projects (a.k.a Development Project)
1.1.5.1 Akraino IoT Area - 8

Personal IoT Networks (PINs)

Existing control
5G local Control
5G Data under local Control

Figure 4.2.10-3: Roaming with Home-routed architecture for ATSSS support (UE registered to different PLMNs)

Figure 6.3.1.2-1: The traffic splitting based on the QoS rule (with ATSSS rule)
3GPP RAN Rel-16 progress and Rel-17 potential work areas

July 18, 2019

https://www.3gpp.org/news-events/2058-ran-rel-16-progress-and-rel-17-potential-work-areas

Slide 7

**Release 16 progressing towards completion**

**5G V2X**
- Targeting advanced use cases beyond LTE V2X

**Industrial IoT and URLLC enhancements**
- Adding 5G NR capabilities for full wired Ethernet replacement in factories: Time Sensitive networking, etc... with high reliability

**5G NR operation in unlicensed bands**
- Includes both Licensed Assisted Access (LAA), as well as Standalone Unlicensed operation

**System improvements and enhancements**
- Positioning
- MIMO enhancements
- Power Consumption improvements
Figure 2-6: Release 16 5G features and enhancements supporting verticals
Redundant User Plane (UP) Paths based on Dual Connectivity

**Figure 6.1.1-2: Solution architecture**

Static approach:
This applies to both IP and Ethernet PDU sessions. The solution is illustrated in the Figure below:

**Figure 6.1.1-3: Static UPF selection**

Dynamic approach:
This applies to Ethernet PDU Sessions. The solution is illustrated in the Figure below:

**Figure 6.1.1-4: Dynamic UPF Selection: anchor change after DC setup for Ethernet PDU Sessions**
1.1.5.1 Akraino IoT Area

Figure 5.34.4-1: User plane Architecture for the Uplink Classifier controlled by I-SMF

Figure 5.6.4.3-1: Multi-homed PDU Session: service continuity case

Figure 5.6.4.3-2: Multi-homed PDU Session: local access to same DN
1.1.5.1 Akraino IoT Area - 12

Figure 4.2-4: 5GS providing access to EAS without UL CL/BP for LBO roaming scenario

Figure 4.3-1: 5GC Connectivity Models for Edge Computing
1.1.5.1 Akraino IoT Area - 13

3GPP 5G System Idle Inactive Connected UE states

Figure 4.2.1-1: UE state machine and state transitions in NR

Figure 4.2.1-2: UE state machine and state transitions between NR/5GC, E-UTRA/EPC and E-UTRA/5GC

Figure 2 Comparison of Signalling involved in Legacy Idle-to-Connected transition (Left) versus Inactive-to-Connected Transition (Right)
The OPG believes that, for Operators to develop a Federated Edge Computing Platform such as the OP, Requirements must be enforceable in Contracts by a Published Set of Standards.

To this end, the OPG proposes selecting ETSI ISG MEC and 3GPP to provide a Standard Reference for an Edge Service End to End (E2E) definition.

We note that 3GPP EDGEAPP Architecture and ETSI ISG MEC Architecture could complement each other in a way that is acceptable to OPG:

*Figure 14: Relationship between ETSI ISG-MEC and 3GPP EDGEAPP architectures*
General

- A first discussion in 3GPP RAN on 5G-Advanced
  - Starting from Rel-18. Official logo is shown on the right
  - 500+ submissions from ~80 different companies/organizations
  - 1200+ checked-in participants

- Careful organization and balanced discussion of topics in three agenda items, aiming for both immediate and longer-term commercial needs
  - Three agenda items:
    - eMBB (evolved Mobile BroadBand)-driven evolution;
    - Non-eMBB-driven evolution;
    - Cross-functionalties for both eMBB-driven and non-eMBB-driven evolution
  - Some stats (very coarse) as shown on the right → generally a balanced evolution in the three directions based on the submissions
2. LF Edge Akraiino Contact for future inquiries and information

Asif Mahmood Awan
Ph.D. Computer Engineering at Jeju National University, Republic of Korea,

Akraiino PTL for Akraiino Blueprint: MEC-based Stable Topology Prediction for Vehicular Networks

E-mail: malikasifmahmoodawan@gmail.com