EVE-OS Overview

How to Modernize the Edge and Stay Secure
Learning Content

› Edge Computing Challenges
› How EVE Modernizes the Industrial Edge
› Commercial Ecosystem Opportunities
› EVE Technology and Security Overview
› Embracing LF Edge Open Source Community Collaboration
Challenges at the Edge

- **Security**
  - No guarantee of network security
  - No guarantee of physical security
  - Onerous security overlays at the edge

- **Diversity of deployed infrastructure**
  - Mixture of remote devices
  - Plethora of apps to orchestrate
  - App integration with several Clouds

- **Scale and automation**
  - Huge # of edge devices, geographically disperse
  - Long maintenance lifecycle (7+ years)

- **Unreliable connectivity**
  - Network outages, latency, expensive bandwidth
  - Might not even control edge network
How EVE Modernizes the Industrial Edge

EVE addresses the unique properties of distributed edge computing nodes deployed outside of the traditional datacenter

**Diversity**
Inherent diversity of technology and domain expertise required

**Scale**
Unprecedented scale and geographic distribution of deployed nodes

**No Perimeter**
No physical or network perimeter dictates a zero trust security model

The distributed edge needs a standard foundation for orchestration and virtualization that is flexible, open and agnostic
Challenges Solved with Edge Virtualization

**Edge Virtualization Engine (EVE)**
- Abstraction layer designed for the edge
- Created and donated by ZEDEDA to LF
- Open sourced under Apache License v2
- Part of Linux Foundation LF Edge Project

**Any Cloud**
- Historian, SCADA or On-Premises System

**Any Application**

**VM or Container**

**Any Gateway at IoT Scale**

**No Compromise to Security**
- (TPM and vTPM)

**Deploy, Secure and Manage**
- Gateway and Apps at Scale

**Any Gateway at IoT Scale**

**Open EVE Controller**

**Open Source**

**CLI interface tool**

**Hardware**

**SaaS**
- Web Console
- App Marketplace

**Commercial**

**Open EVE Controller**

**Open Source**

**ZEDEDA**

**LF Edge Project**
Example ZEDEDA Enterprise Integration
EVE Technology and Security Overview
EVE Architecture

EVE-OS
- RAM overhead: 500M
- CPU overhead: 1 core
- Disk overhead: 500M

EVE managed, workload-centric storage

User Edge Compute Hardware

Open API

EVE Controller

Hypervisor (KVM [default] or Xen, ACRN)

Partition A

Partition B

Disk overhead: 500M

RAM overhead: 500M

CPU overhead: 1 core
EVE-OS to EVE Controller “Onboarding”

- Cryptographic device identity created when EVE-OS installed (factory)
  - Key pair generated in TPM; private key never leaves TPM
  - Device is imprinted with the controller to trust (a root CA certificate)
- Device can be pre-onboarded in factory, optionally with applications too
- User registers their hardware using device certificate or serial number
- See [https://github.com/lf-edge/eve/blob/master/docs/REGISTRATION.md](https://github.com/lf-edge/eve/blob/master/docs/REGISTRATION.md)
Remote Manage Any Edge Node

- Any type of silicon and device
- Automated on-boarding
- Autonomous operations

No field expertise required
Publicly Documented APIs

**EVE-OS**
Secure API over TLS

**EVE Controller (ZEDCloud or Open EVC)**
Secure API over HTTPS

---

**EVE-OS API**
[https://github.com/lf-edge/eve/tree/master/api](https://github.com/lf-edge/eve/tree/master/api)

**ZEDCloud API**
[https://zedcontrol.zededa.net/api/v1/docs/](https://zedcontrol.zededa.net/api/v1/docs/)

**Open EVC Interface (API)**
EVE API Security Works Through Firewalls, Proxies

1. TLS to trusted parties (direct to controller and/or via proxy)
2. End-to-end signature over payload (proxy can not view nor modify)
3. Sensitive data encrypted end-to-end (also at rest)
Zero Trust
People, Process, and Technology

- **People**
  - Remove need for device usernames/passwords
  - Role-based access control (RBAC) and multi-tenancy in controller

- **Process**
  - “Zero Touch” hardware deployment to field
  - Design for 7+ year lifetime at the edge
  - Secure, scalable distribution of updates
  - API reports (resource usage, firewall violations) enable analytics in controller

- **Standard security technologies for the user edge**
  - Hardware root of trust (e.g., TPM)
  - Crypto-based identification
  - Measured boot and remote attestation
  - Encryption at rest and in-flight (TLS); keys sealed by TPM
  - Signed images for EVE-OS and applications
  - Use hypervisors for strong isolation and defense in depth
  - Distributed firewall for every app
  - Physical security – port isolation
  - Support deployment of virtual security appliances
Embracing LF Edge Open Source Collaboration
Community Collaboration Resources

Project page https://www.lfedge.org/projects/eve/
Wiki https://wiki.lfedge.org/display/EVE/EVE
  › Mailing list https://lists.lfedge.org/g/eve
  › Zoom calls (calendar subscription on wiki)
GitHub https://github.com/lf-edge/eve
Slack https://lfedge.slack.com

Roadmap
https://wiki.lfedge.org/display/EVE/Feature+Roadmap
Ready to Transform Your Edge?