RESEARCH REVIEW 2019

KalKi: High Assurance Software-Defined IoT Security

Sebastian Echeverria

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM19-1110

- DoD urgently needs to embrace **commodity IoT technologies** in its tactical systems.
- Security concerns over **untrusted supply chains** are an obstacle.
- We are developing a **solution that remains resilient and trustworthy**, even in the presence of a powerful attacker.

Attacks on IoT Devices

Microsoft catches Russian state hackers using IoT devices to breach networks
arstechnica

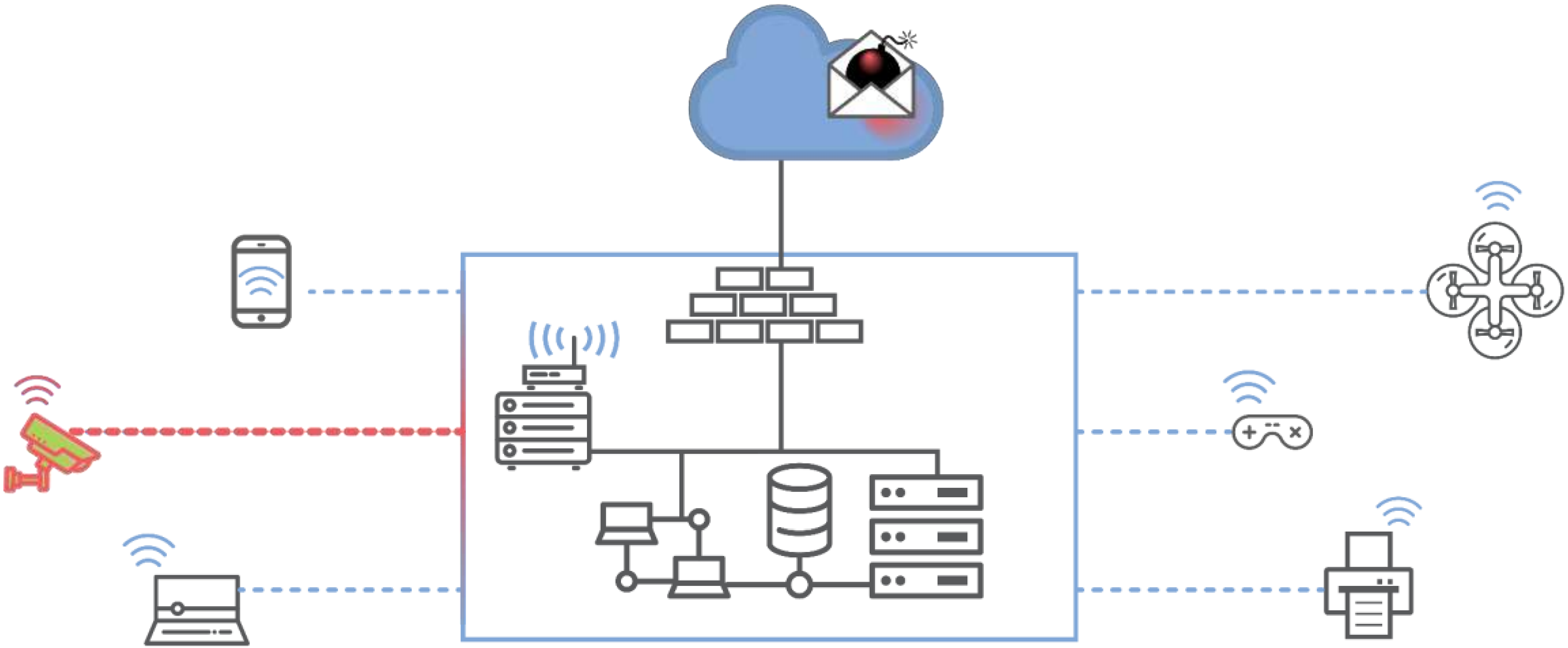
Unpatched Routers Being Used To Build Vast Proxy Army Spy On Networks
arstechnica

A 100,000-router botnet is feeding on a 5-year-old UPnP bug in Broadcom chips
arstechnica

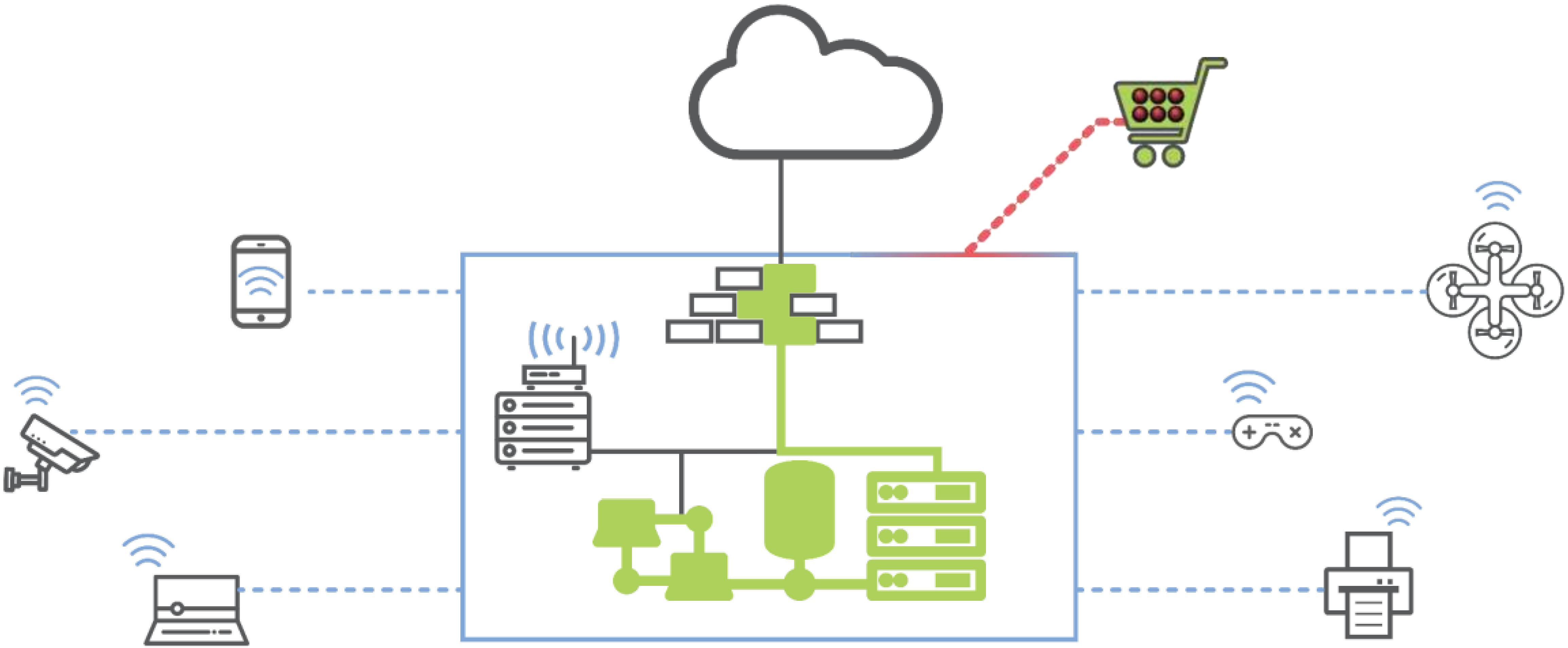
Latest Mirai variant targets routers and other IoT devices using 13 exploits
cyware.com

Your smart air conditioner could help bring down the power grid
Hacked appliances could overwhelm the grid, researchers say.
cnet.com

IoT Threats – Vulnerable Device



IoT Threats – Compromised Device



KalKi:
High Assurance Software-Defined IoT Security Platform

Solution: Move Security Enforcement to the Network

Create an IoT security platform highly resilient to a collection of prescribed threats

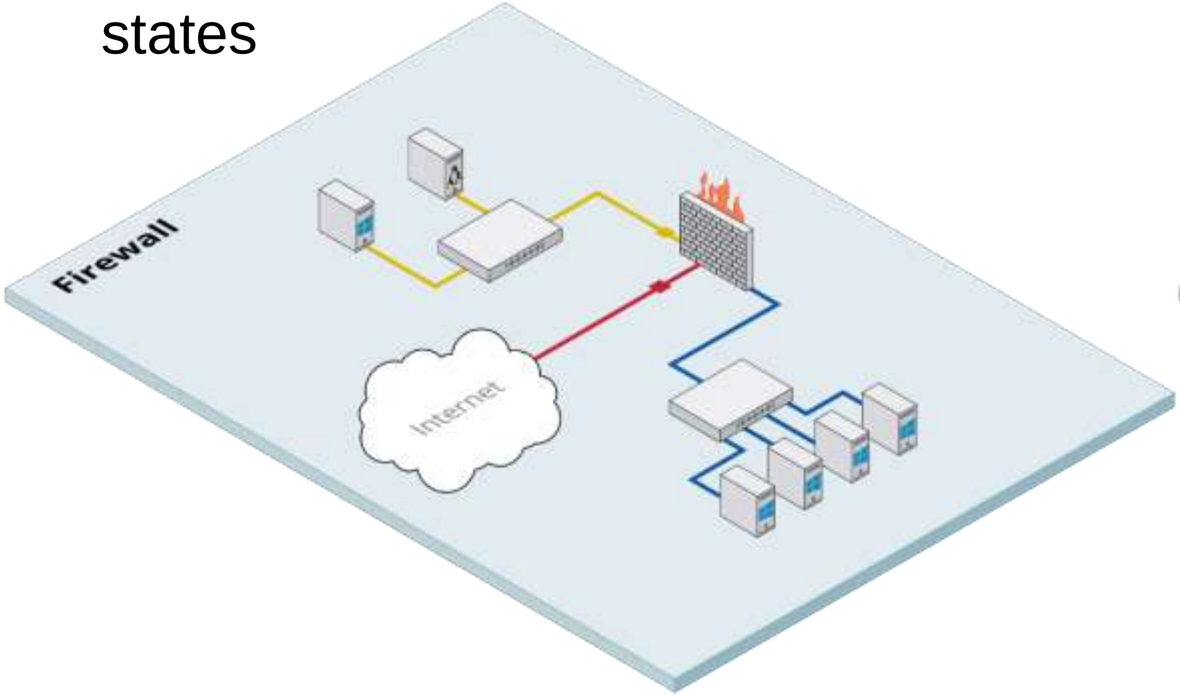
- Enables the integration of IoT devices into DoD networks
- Protects the networks even if the IoT devices are not fully trusted or configurable

The term "KalKi" is of Sanskrit origin, and it is the name of an avatar of the god Vishnu, the destroyer of filth and bringer of purity, truth and trust.

Limitations of Existing Systems

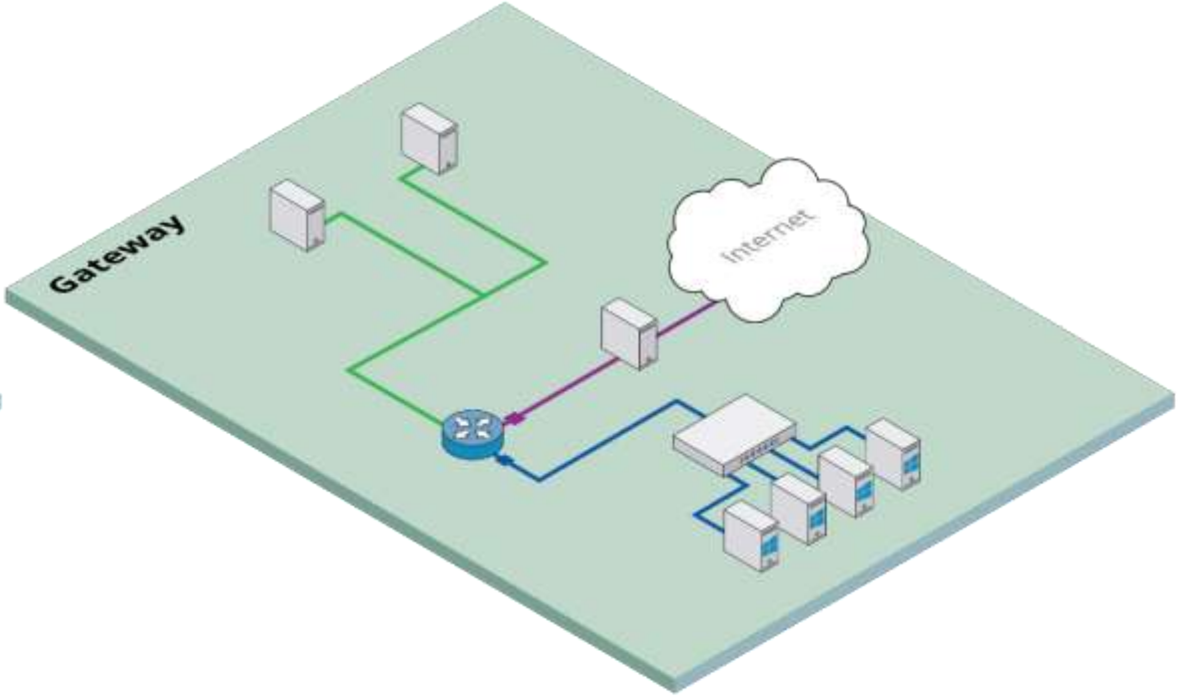
Static Firewalls

- Are not device-specific
- Cannot adapt to changing security states



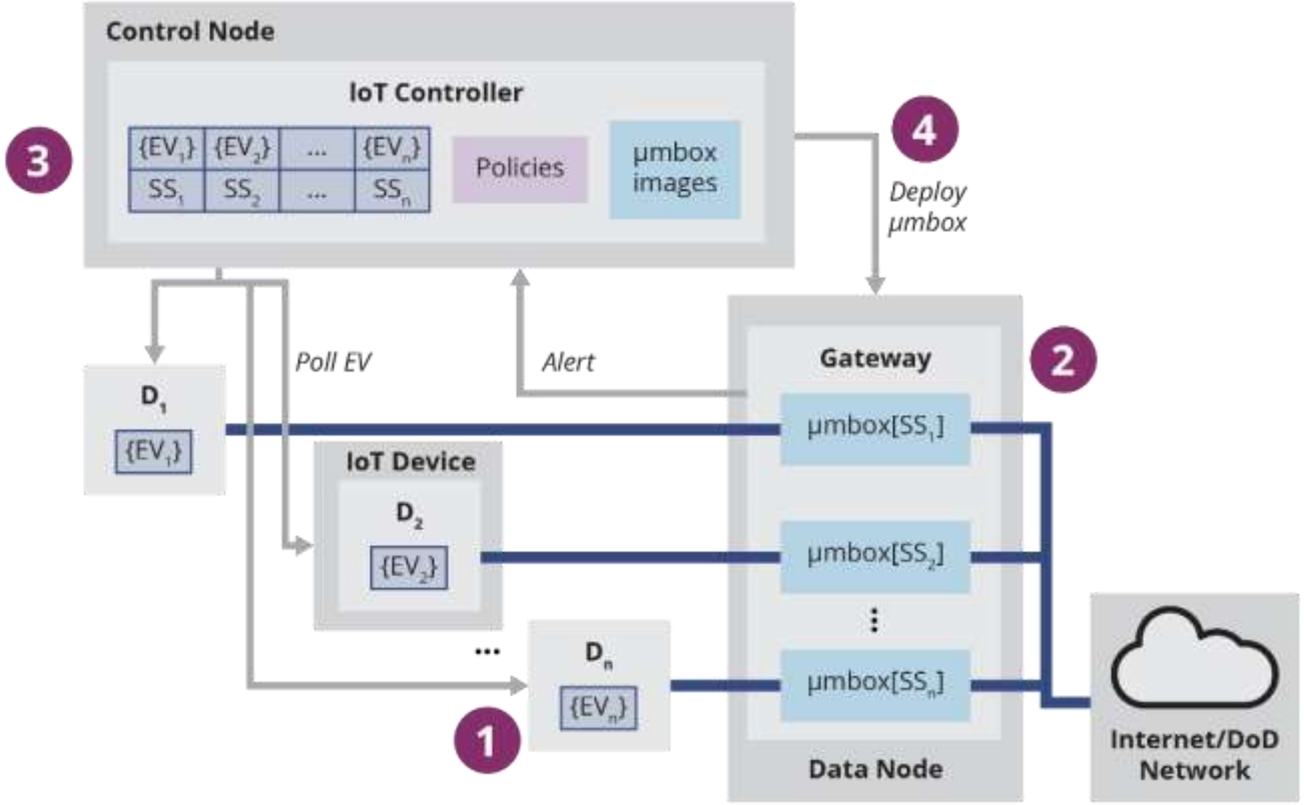
Gateways/Firewalls

- Can become compromised



Software-Defined Aspect

Use software-defined networking (SDN) and network function virtualization (NFV) to create a highly dynamic IoT security platform.



1. Each IoT device, D, senses/controls a set of environment variables, EV.

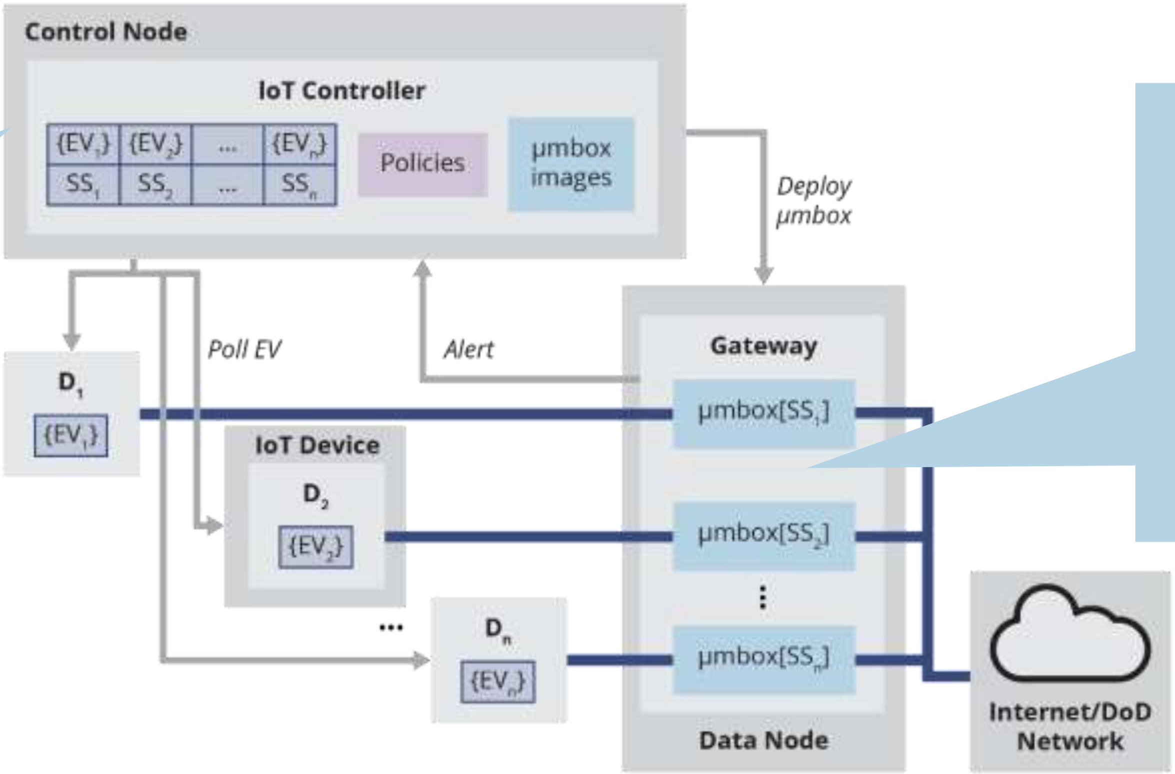
2. Network traffic to/from each device is tunneled through μmboxes that implement the desired network defense for the device's current security state.
   μmbox[SS_1] = Firewall
   μmbox[SS_2] = IPS, ...

3. IoT controller maintains a shared statespace composed of {EV} and security state (SS) for each device.
   SS = {Normal, Suspicious, Attack}

4. Changes in the shared statespace are evaluated by policies and may result in the deployment of new μmboxes.

High Assurance Aspect

Incrementally develop and verify security properties of elements of the software-defined IoT security platform using überSpark/überXMHF, a framework for building secure software stacks.



Control Node Properties

- Policy data integrity, including security state machine

Data Node Properties

- µmbox image storage integrity
- µmbox deploy-time integrity, including integrity of data flow definition

Year 1 Accomplishments

Initial Threat Model to guide development

Policy Model to set conditions to change security state, and actions to be taken

Initial Architecture and prototype of the IoT Security Platform

FUNCy Views (Secure) system architecture: hardware-assisted, low-latency, low-TCB, compartmentalization of legacy code on x86 platforms

Initial Dashboard to configure system

Year 2 Accomplishments

IoT Security Platform prototype full development

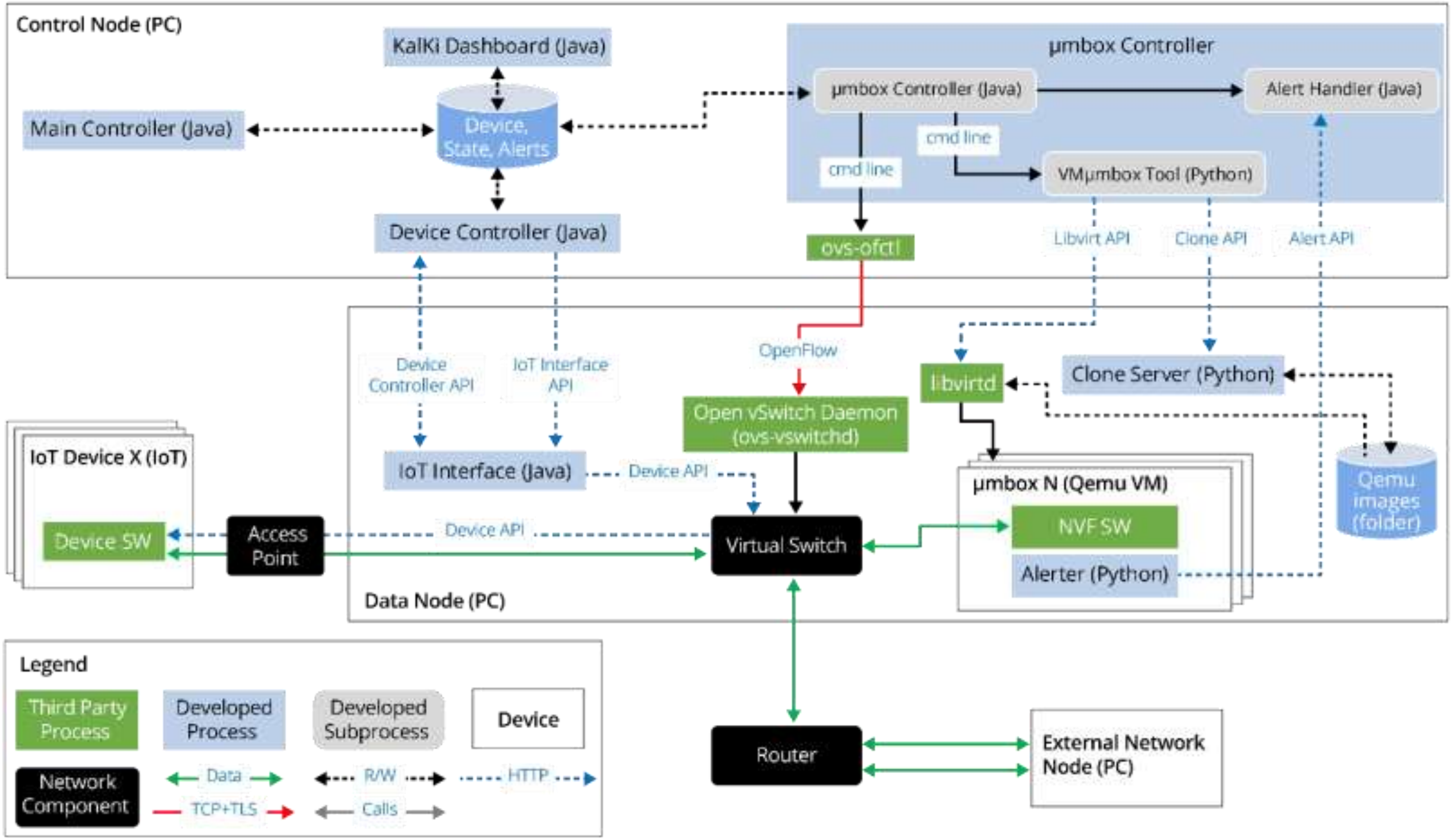
Dashboard Update

Creation of Policies and µmboxes for four representative IoT devices

Experiment to Test different scenarios and red team attacks

Extension of überXMHF and überSpark to include überObject protections for sensitive areas of the Control node and Data node

Year 2 Accomplishments – IoT Security Platform Prototype

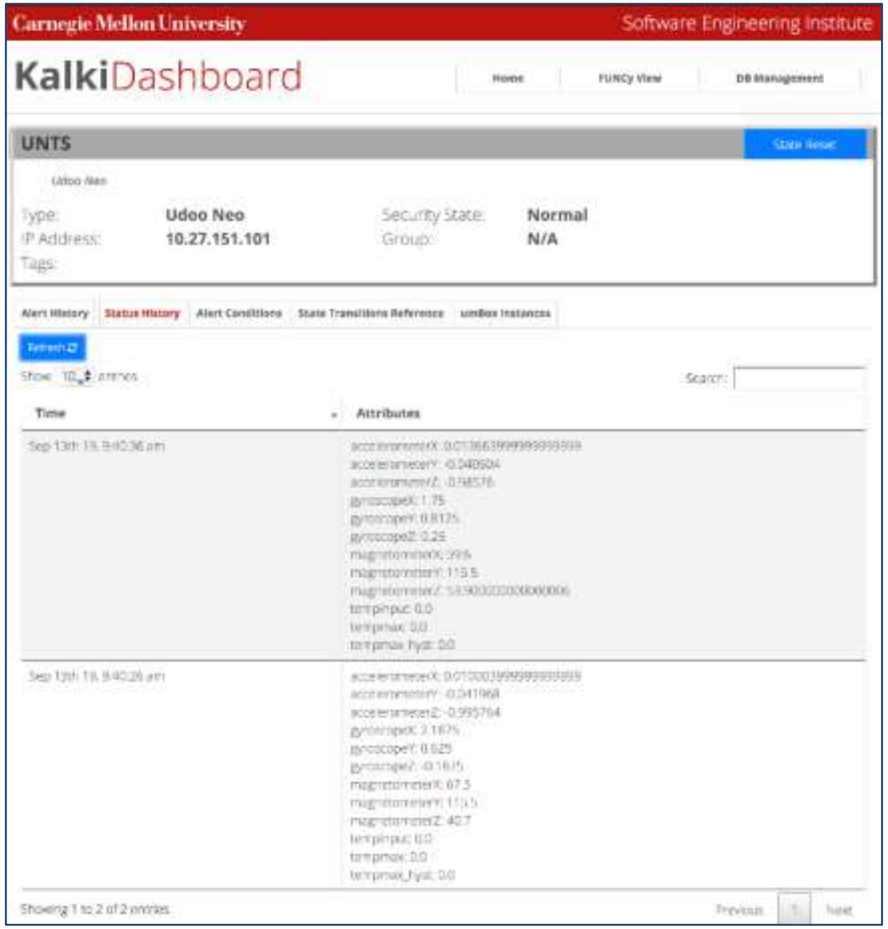


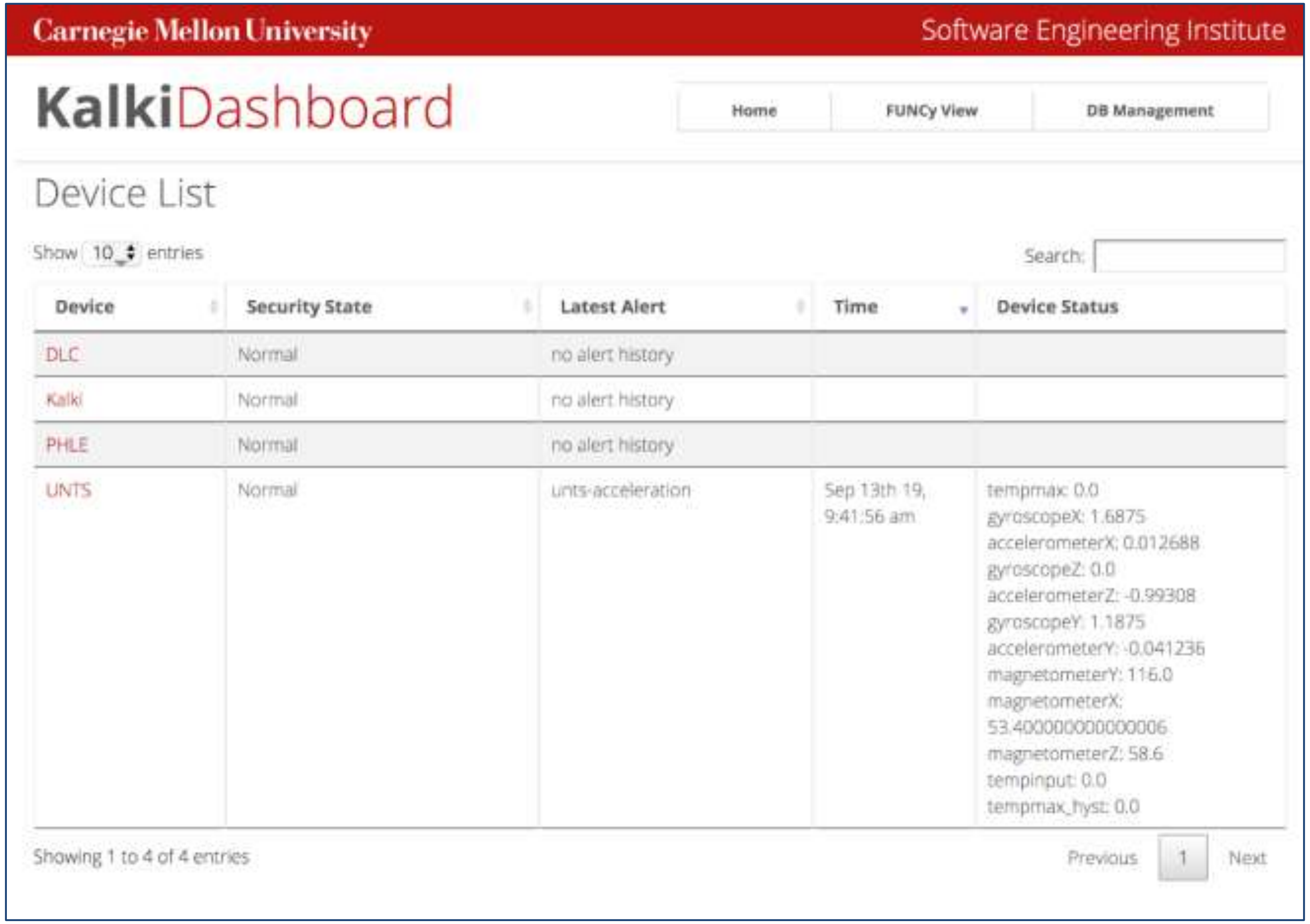
IoT Security Platform prototype implemented (software-defined part)

- Able to monitor device-specific vulnerabilities
- Supports different policies for each security state
- Runs on commodity hardware/software

Year 2 Accomplishments – Dashboard Update

Real-time monitoring of security state, easy configuration of security policies



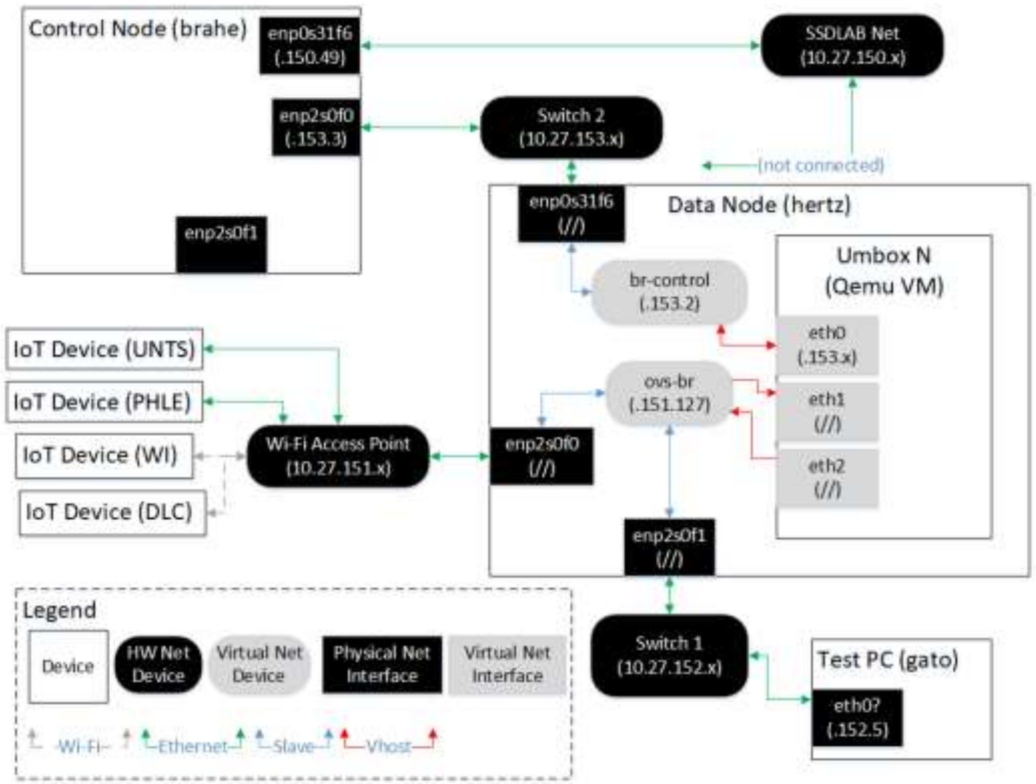


Year 2 Accomplishments – Policies and µmboxes

Creation of policies and µmboxes for four representative IoT devices

Smart Plug	Temperature Sensor	IP Camera	Smart Light
			

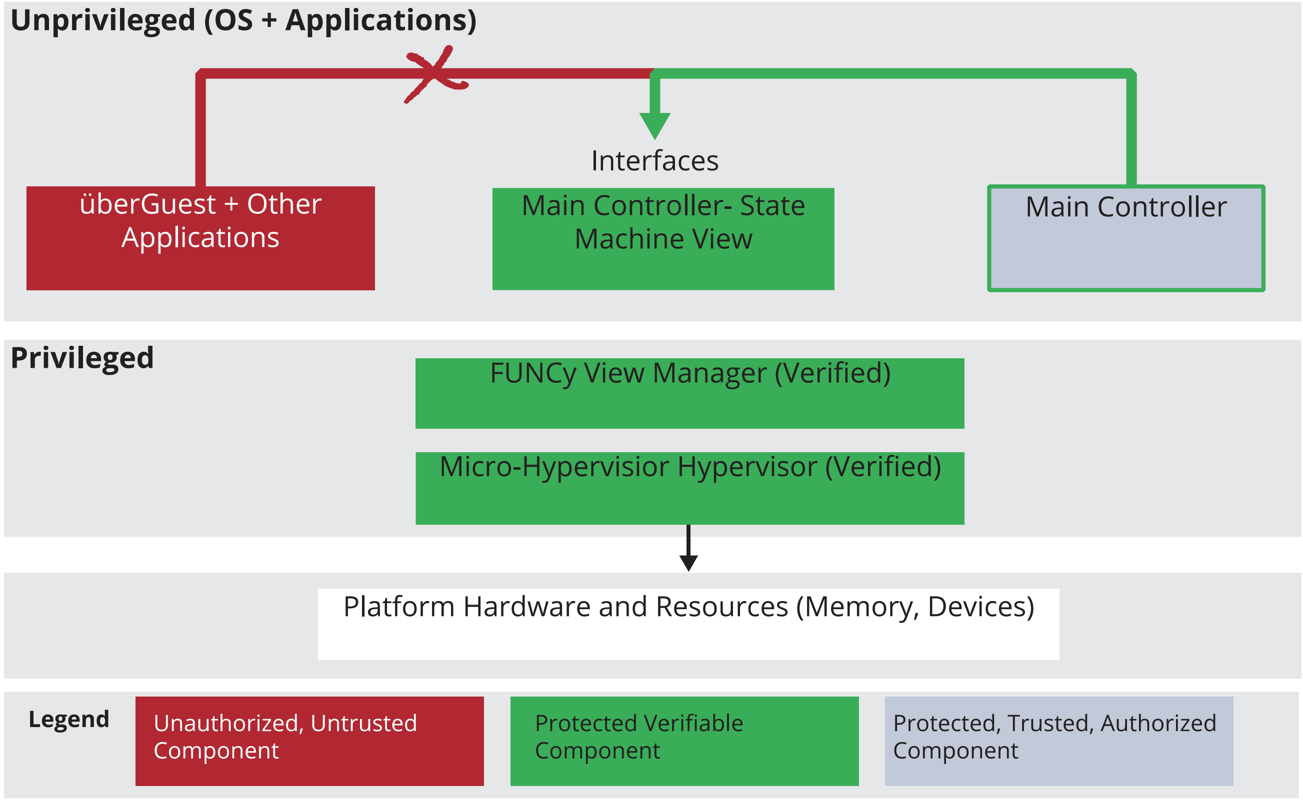
Year 2 Accomplishments – Experiment + Red Team Attacks



Executed multiple test scenarios to measure:

- Resiliency to attacks
- Performance (time to react to threats)
- Scalability (effect of the number of devices in performance)

Year 2 Accomplishments – überXMHF Extensions



Added support to protect state machines using überObjects via FUNCy views

- Verified, lightweight micro-hypervisor protects resource access
- Unauthorized applications can't access State Machines encapsulated as überObjects

Year 3 – Next Steps

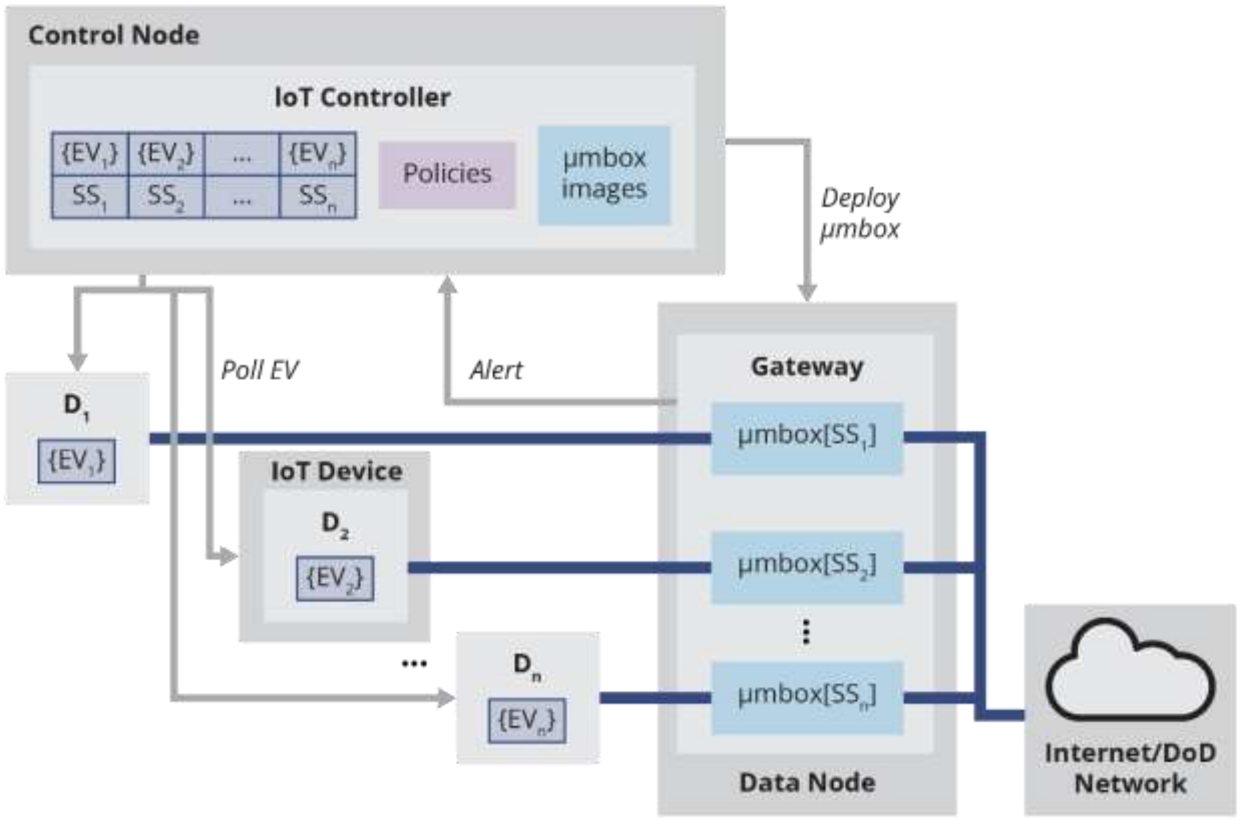
- Final platform development and optimizations
  - Integrate überXMHF security properties into prototype
  - Simplify integration of new devices and policies
  - Increase performance and reduce resource utilization
- Transition activities — identify transition partners for validation, testing, and adoption
  - Working with CMU liaisons for Navy (LCDR Christopher Lueken) and Marine Corps (LCDR Jeff Greenwald)
  - Establishing contacts with organizations leading IoT projects, including US Army Research Office (Durham), USAF Office of Scientific Research (Arlington), and Purdue University
- Publication of results and open source release of platform code

Looking Ahead

NEAR	MID	FAR
• Full platform tested with realistic IoT deployments • Results published	• Platform adapted and integrated into existing DoD networks	• AI techniques developed to automate and improve security policies and protections

KalKi IoT Security Platform - Summary

Enables the **secure integration of IoT devices** into DoD networks even though they are **not fully trusted**



- Has flexible **policies** to define states, transitions and actions
- Reacts using **network and environment** information
- Uses **different network defenses** for each device and state
- Adapts to **device-specific vulnerabilities** or limitations
- Secures critical areas through integration with **überSpark /überXMHF**