Automated IoT Security

IETF103 - Bangkok
09/11/2018

Oscar Garcia-Morchon (Philips)
Thorsten Dahm (Google)

Goal of the Draft

https://datatracker.ietf.org/doc/draft-garciamorchon-t2trg-automated-iot-security/

Solving the mismatch between

- The security capabilities and settings with which IoT devices are designed / manufactured / deployed
- The actual security requirements of the IoT devices in different environments over time

Work derived from the “State-of-the-Art and Challenges for the Internet of Things Security” document: https://datatracker.ietf.org/doc/draft-irtf-t2trg-iot-seccons/

Problems to solve

Problem 1: Different environments

- Deploying in a home is not the same as in an office or in the Department of Defense

Problem 2: Evolving threats

- Algorithms become insecure
- Bugs in software are found
- Users change their preferences

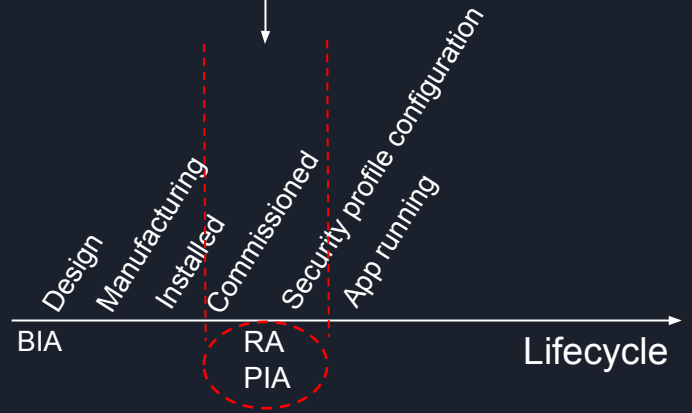
Problem 3: Pre-configuration is not always right

- a product owner doesn’t know they should disable a protocol;
- a developer doesn’t remove all of the off ending code (just some uses of it);
- the documentation doesn’t mention the protocol, even though the device implements it;

PASC - Protocol for Automatic Security Configuration

High-level idea of PASC

- Thing to publish its usage profile to a Gateway
- Gateway gathers additional information about the Thing, the usage and expected interactions of the smart object with other devices in the deployment environment (e. g. via MUD, portscan)
- Gateway performs an automated risk assessment
  - Determines potential threats on the device and on deployment environment
  - Determines security profile containing mitigations
- Deploy updated security profiles
  - to the Thing itself
  - to other devices already present in the deployment environment (other smart objects, Firewalls)

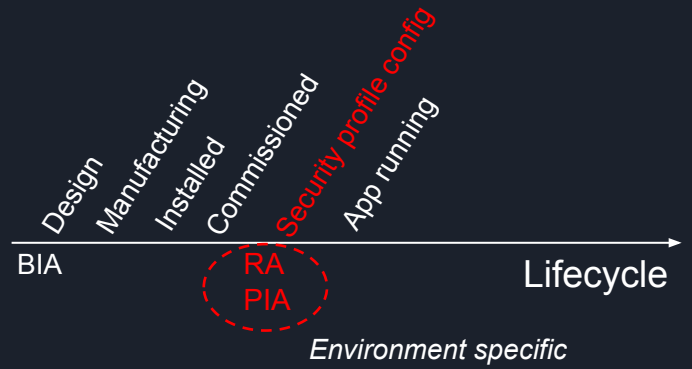


PASC - Protocol for Automatic Security Configuration

Enabling automatic security configuration of Things by shifting methodologies for risk management from the tailored product design and implementation phases to the onboarding phase

Current practice

PASC: 1st protocol in our draft



PAVA - Protocol for Automatic Vulnerability Assessment

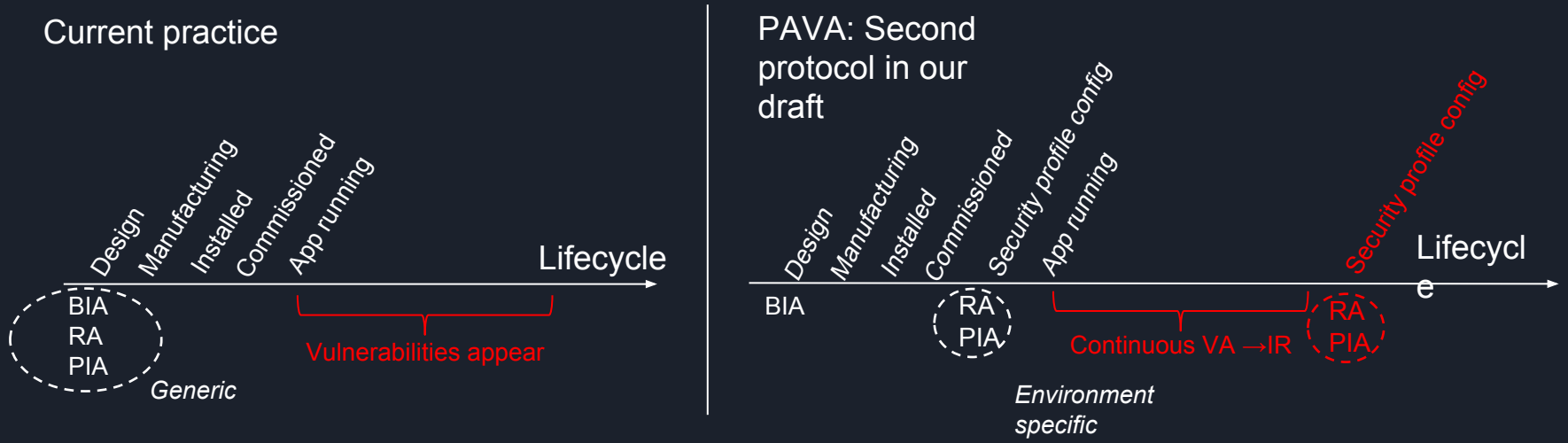
High-level idea of PAVA

- Thing to send standardized reports of potential vulnerabilities to a Gateway via Syslog
- Gateway to analyse the reports and decide regarding the existence of a vulnerability, its origin and its impact
- Gateway to run additional and continuous analysis of each Thing based on Security Profile

Enabling updates of security profiles in real time and automatic incident reporting towards

- the user
- the manufacturer
- the deployment environment provider

PAVA - Protocol for Automatic Vulnerability Assessment



Benefits

- Benefits for manufacturers
  - no need to decide which security mitigations are required for each product
  - simply describe the expected usage of the Thing

- Benefits for system operators
  - minimize operational cost while ensuring that the system remains secure at any moment
  - enabling automation for security configuration in deployment environments with potentially millions of smart Things

- Benefits for end users
  - security configuration is done in an automatic way
  - users “don’t need to do anything”

Work to do

- Definition of IoT use cases, overall architecture for IoT security automation, and applicable techniques (e.g., MUD, SDN, ACE,...) to realize PASC & PAVA

- Define minimum viable PASC & PAVA protocols
  - information required during onboarding
  - describing the required input for the automation part
  - defining the output required or desired by users, routing infrastructure and end devices
  - standardizing the PASC Messages (including transport protocol for PASC and PAVA messages)
  - creating the Risk Analysis and Privacy Impact assessment logic to generate the (SDN) security configuration
  - standardizing the PAVA policy and messages for vulnerability assessment & messages/Information required from services to perform PAVA