Lessons learned in IoT Threat Modelling

Paul Bottinelli, Névine Ebeid, Kevin Henry

escrypt, Embedded Security by ETAS, Canada

ETAS Embedded Systems Canada, Inc

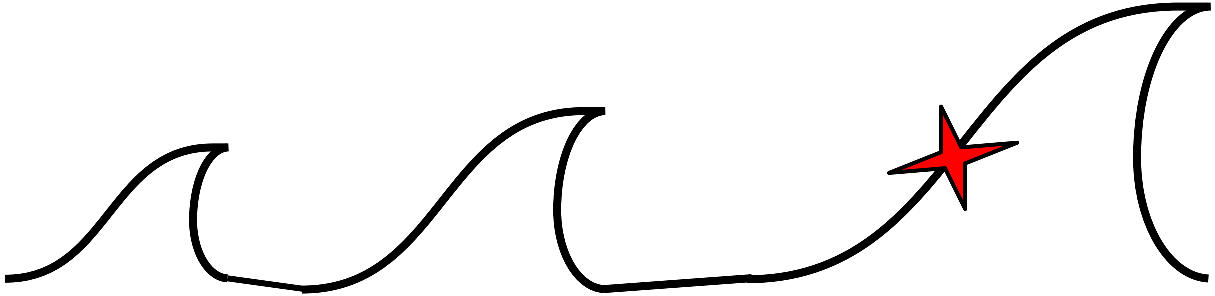
Outline

- Introduction – IoT, security and cryptographic modules
- Lessons learned in IoT threat modelling
- Methodology and examples
- Conclusion

Introduction

The Internet of Things

- IoT is the next (third) wave of Internet development



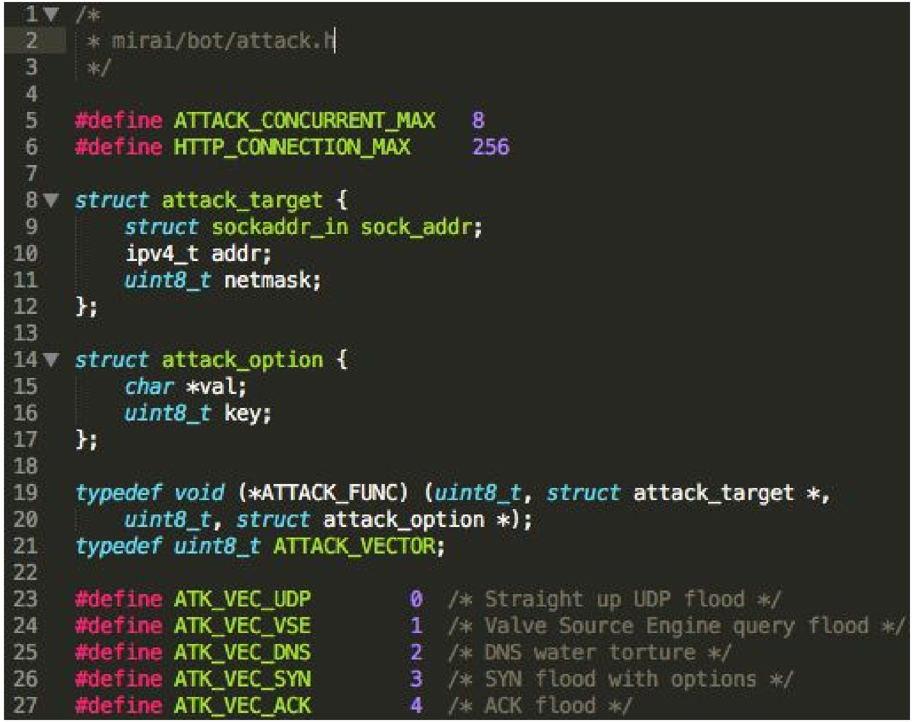
1990s 2000s 2020

- 1st wave – 1 billion users with fixed internet
- 2st wave – 2 billion additional users with mobile internet
- 3rd wave –up to 26 billion connected “things”
- **HP** study revealed 70% of IoT devices have inadequate security

Introduction

The Internet of **Insecure** Things

- Common security issues leading to large and very disruptive attacks
    - **Mirai**: malware converting IoT devices in botnet used in largest DDoS
    - **BrickerBot:** malware similar to Mirai, used in Permanent DoS (PDoS)
- *Lack of manufacturer security awareness*



```c
/*
 * mirai/bot/attack.h
 */

#define ATTACK_CONCURRENT_MAX   8
#define HTTP_CONNECTION_MAX     256

struct attack_target {
    struct sockaddr_in sock_addr;
    ipv4_t addr;
    uint8_t netmask;
};

struct attack_option {
    char *val;
    uint8_t key;
};

typedef void (*ATTACK_FUNC) (uint8_t, struct attack_target *,
    uint8_t, struct attack_option *);
typedef uint8_t ATTACK_VECTOR;

#define ATK_VEC_UDP         0  /* Straight up UDP flood */
#define ATK_VEC_VSE         1  /* Valve Source Engine query flood */
#define ATK_VEC_DNS         2  /* DNS water torture */
#define ATK_VEC_SYN         3  /* SYN flood with options */
#define ATK_VEC_ACK         4  /* ACK flood */
```

IoT and cryptographic modules

How do IoT devices and cryptographic modules relate?

- IoT devices can be viewed as extension of cryptographic modules
  - FIPS 140-2 description: *set of hardware, software, and/or firmware that implements Approved security functions and is contained within the cryptographic boundary*
  - Current certification is not adequate to provide the required assurance of the "faithfulness" of an IoT device

[ICMC2016 – David McGrew]

- But it is also much more!
  - Connected
  - Computing (not only cryptographic operations) and Data
  - Whole system that depends on it and functions in parallel to it

IoT and security

- What is unique about IoT and security?
  - Manufacturing and deployment process
  - Large attack surface
  - Hostile environment
- Identified some common insecurity that we used as groundwork for performing threat modelling

Lessons learned

- There is a Gap
  - Theory vs Practice
  - Design vs Implementation
- Existing threat modelling frameworks difficult to apply to the IoT
  - IoT systems are big and complex
  - Price of device has to be kept low
  - Fast paced environment: companies don't take time to invest in threat modeling during design phase

Lessons learned

- Certification valuable, but has limitation
  - IoT device is only a (small) part of the system
  - Might encourage bare minimum
  - Expensive
- Lessons learned: in order to achieve a minimum level of security in IoT, threat modelling has to be
  - Cheap
  - Simple and fast
  - Reiterated

Lessons learned

Answer

- Answer: A lightweight framework
    - Series of targeted questions
    - Tailored for IoT ecosystem
    - Based on OWASP's IoT Framework Security Considerations
    - Does not compete with certification

Methodology and examples

Examples from template

	Yes	No	Unk.	N/A
2.1.9 **Default credentials**				
2.1.9.1 No default credentials to access the device	○	○	○	○
2.1.9.2 No shared credentials	○	○	○	○
2.1.10 **Fail-safe defaults principle**				
2.1.10.1 Interfaces disabled by default	○	○	○	○

Methodology and examples

Real examples

	Yes	No	Unk.	N/A
2.1.9 **Default credentials**				
2.1.9.1 No default credentials to access the device	○	●	○	○
High Default (root, default) credentials for SSH				
High Default (root, default) credentials for web interface				
2.1.9.2 No shared credentials	○	●	○	○
High Same credentials for SSH and web interface				
2.1.10 **Fail-safe defaults principle**				
2.1.10.1 Interfaces disabled by default	○	●	○	○
Med Telnet port open for no reason				

Goal: *Address lessons learned*

- Raise awareness on customer's side
- Initiate dialogue instead of final binary outcome
- Drive best practices approach during design (blank template) and/or development (filled template)

➢ Allow to reiterate at lower cost

Methodology and examples

escrypt
Embedded Security by ETAS

Goal: *Address lessons learned*

- Template is series of simple targeted questions
- Broken down by components of a generic IoT system architecture
- No need to start from scratch for every new threat modelling or security assessment

➢ Allow to make it cheap and fast

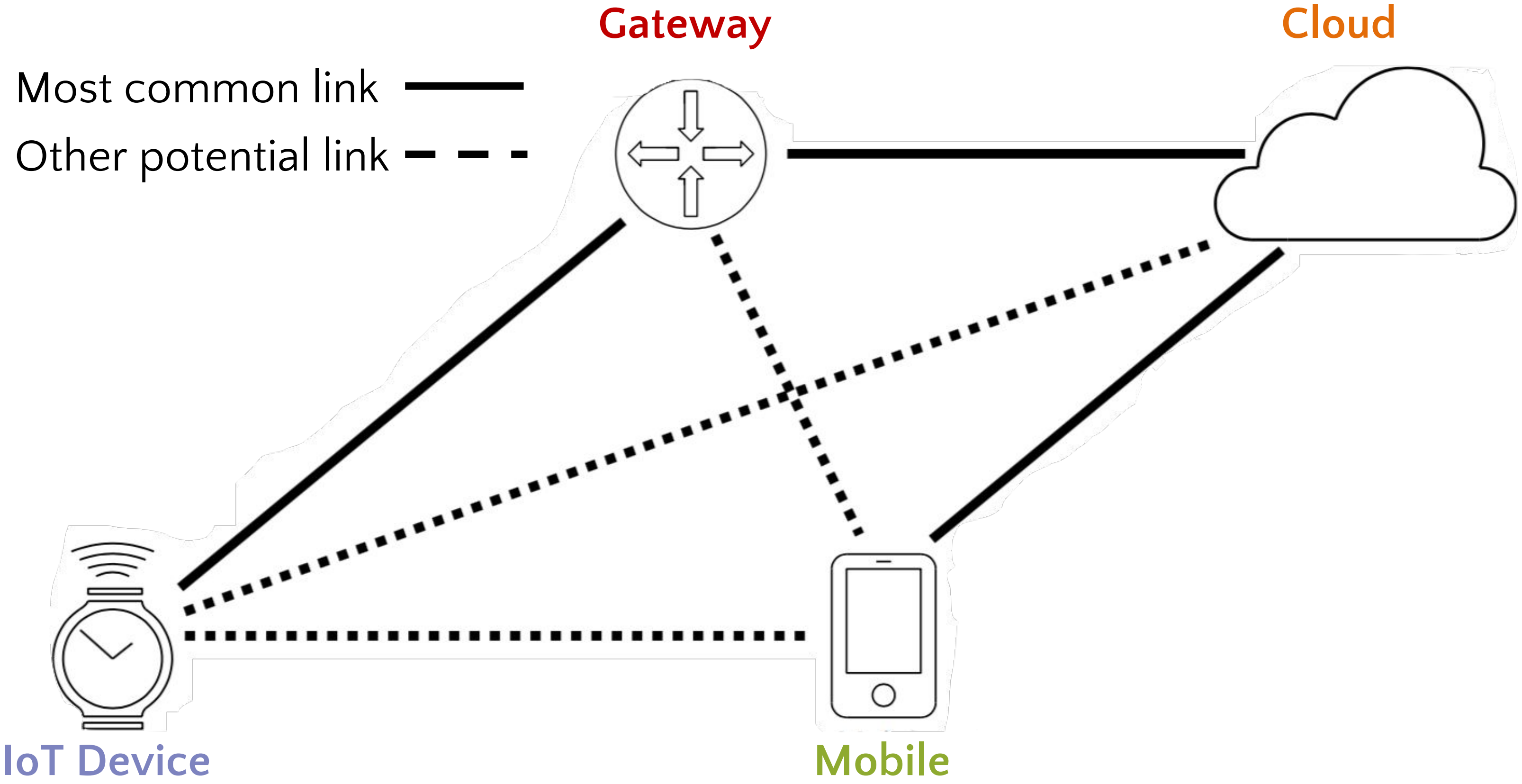
Methodology – Summary and comparison

Comparison with other existing industry standards threat modelling

Common Criteria	FIPS 199 CIA	Our approach
● Very generic ● Expensive ● Complex ● Documentation based ● Long process ● Certification	● Very generic ● Expensive ● Quite simple ● Documentation based ● Long process ● Certification	● Targeted ● Cheap ● Simple ● Adaptable ● Fast ● **Not intended to be a certification**

Methodology

Generic Architecture of IoT system



Most common link ——

Other potential link - - -

Gateway

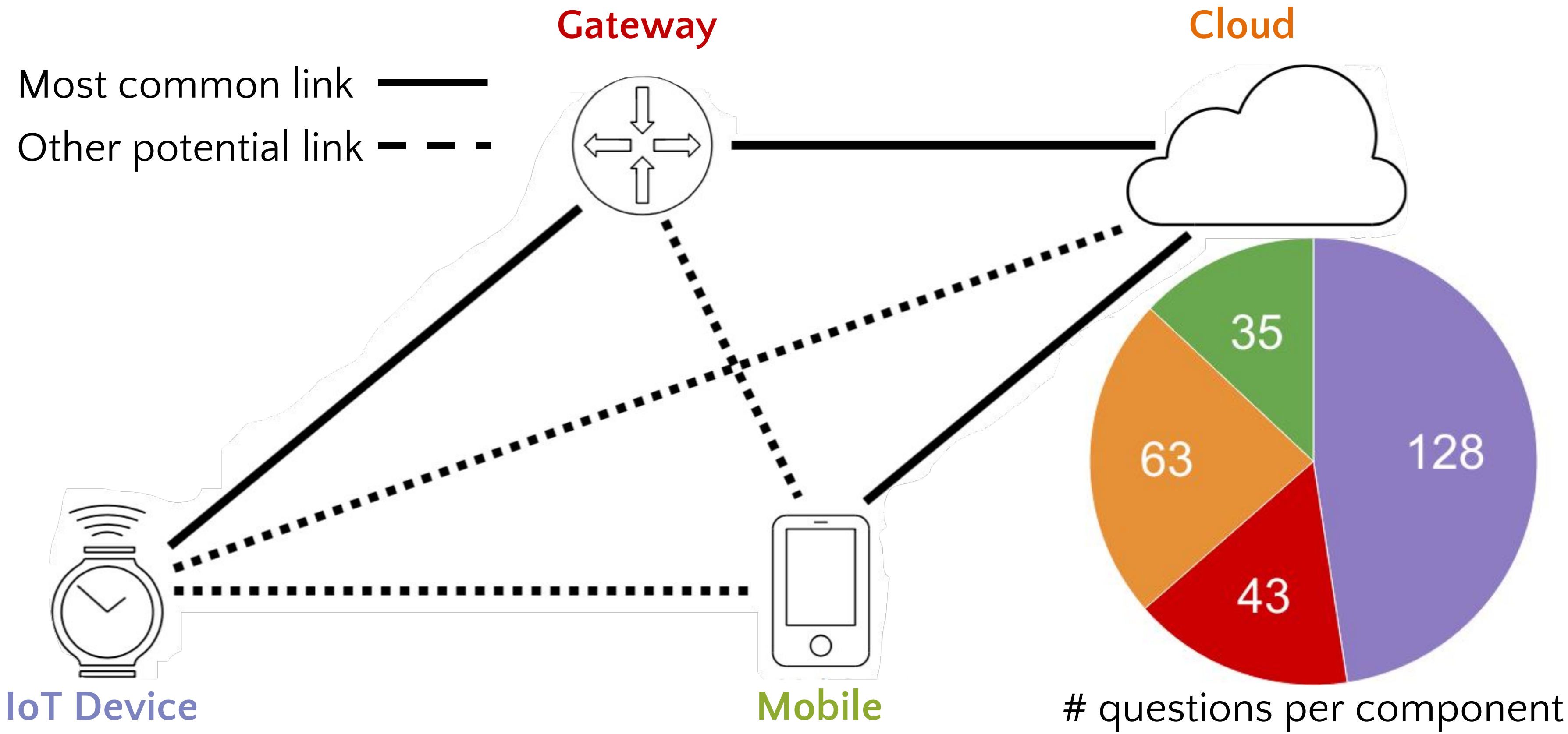
Cloud

IoT Device

Mobile

Methodology

Generic Architecture of IoT system



# questions per component

Methodology and examples

Template examples

2.1.7 **Update verification and software release process**	Yes	No	Unk.	N/A
2.1.7.1 Updates through secure channel	○	○	○	○
2.1.7.2 Integrity verified after download	○	○	○	○
2.1.7.3 Authenticity verified after download	○	○	○	○
2.1.7.4 Integrity verified before installation	○	○	○	○
2.1.7.5 Authenticity verified before installation	○	○	○	○
⋮ ⋮		⋮		

Methodology and examples

Template examples for Gateway

2.2.10 **Secure web interface**	Yes	No	Unk.	N/A
2.2.10.1 Web interface access to the Gateway	○	○	○	○
If Yes ○ :				
2.2.10.2 Limited access to web interface	○	○	○	○
2.2.10.6 Secure communication to web interface (e.g., with TLS)	○	○	○	○
2.2.10.7 Not using self-signed or invalid certificates	○	○	○	○

Methodology and examples

Template examples

2.1.2 **Channel security**	Yes	No	Unk.	N/A
2.1.2.1 Communication through a secure channel (encrypted and authenticated)	○	○	○	○
⋮ ⋮			⋮	
2.1.2.5 Key generation/distribution follows a process	○	○	○	○
⋮ ⋮			⋮	

Methodology and examples

Template examples

	Yes	No	Unk.	N/A
2.1.2 **Channel security**				
2.1.2.1 Communication through a secure channel (encrypted and authenticated)	●	○	○	○
Encrypted channel with WPA2-PSK				
2.1.2.5 Key generation/distribution follows a process	●	○	○	○
Critical WPA2 passkey generation is weak				
⋮ ⋮		⋮		

Conclusion

- Lack of security awareness in IoT
- Remedy, make threat modelling
  - Cheap
  - Fast and simple
  - Continuous, a part of development process
- Our answer
  - Threat modelling as targeted questions
  - E.g., Customer *A* thought their product was good enough
    - We quickly identified issues
    - This prompted a mindset change, dialogue and relationship