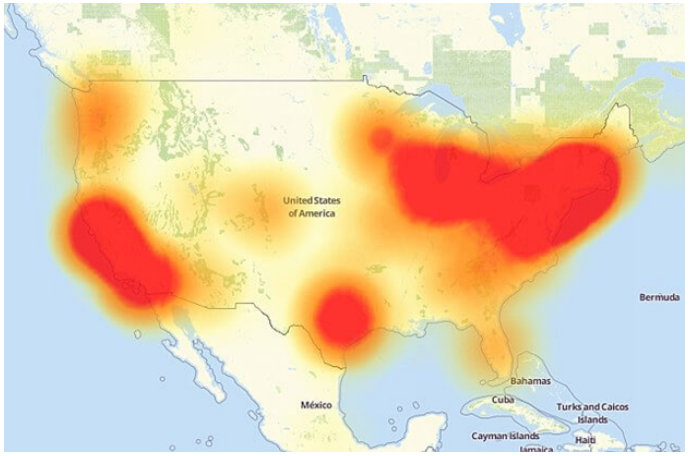
IoTrain-Sim

IoT Training System Using the Cooja Network Simulator

Motivation

- The growth rate of IoT devices is exponential, with 50 billion devices being predicted for 2020
- Design and implementation issues will cause more severe problems regarding IoT safety/security
- People lack knowledge and awareness of IoT security, hence IoT security education and training are extremely urgent



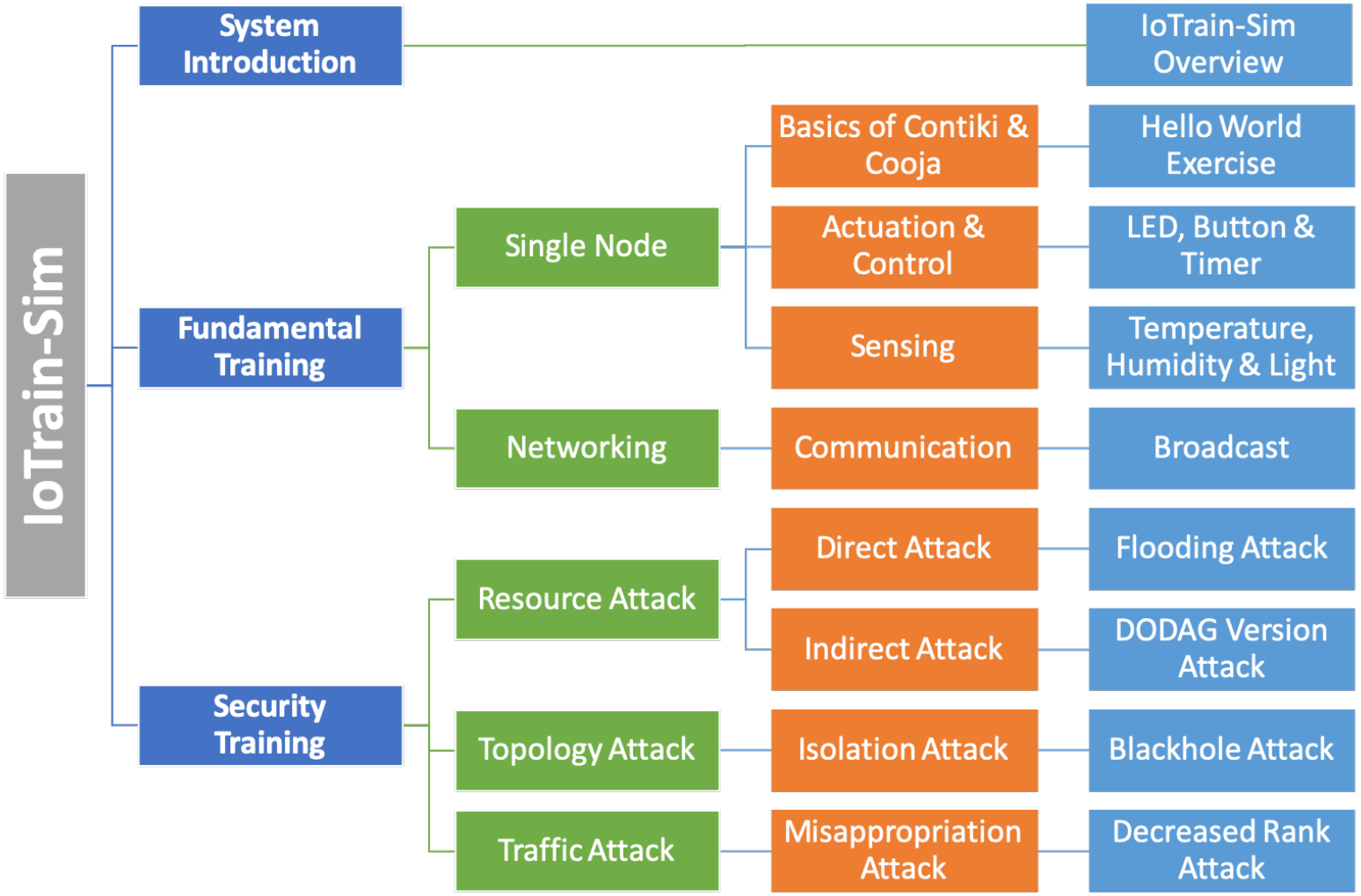
Effects of Mirai DDoS attacks in the USA

Source: https://www.theverge.com/2016/10/21/13362354/dyn-dns-ddos-attack-cause-outage-status-explained

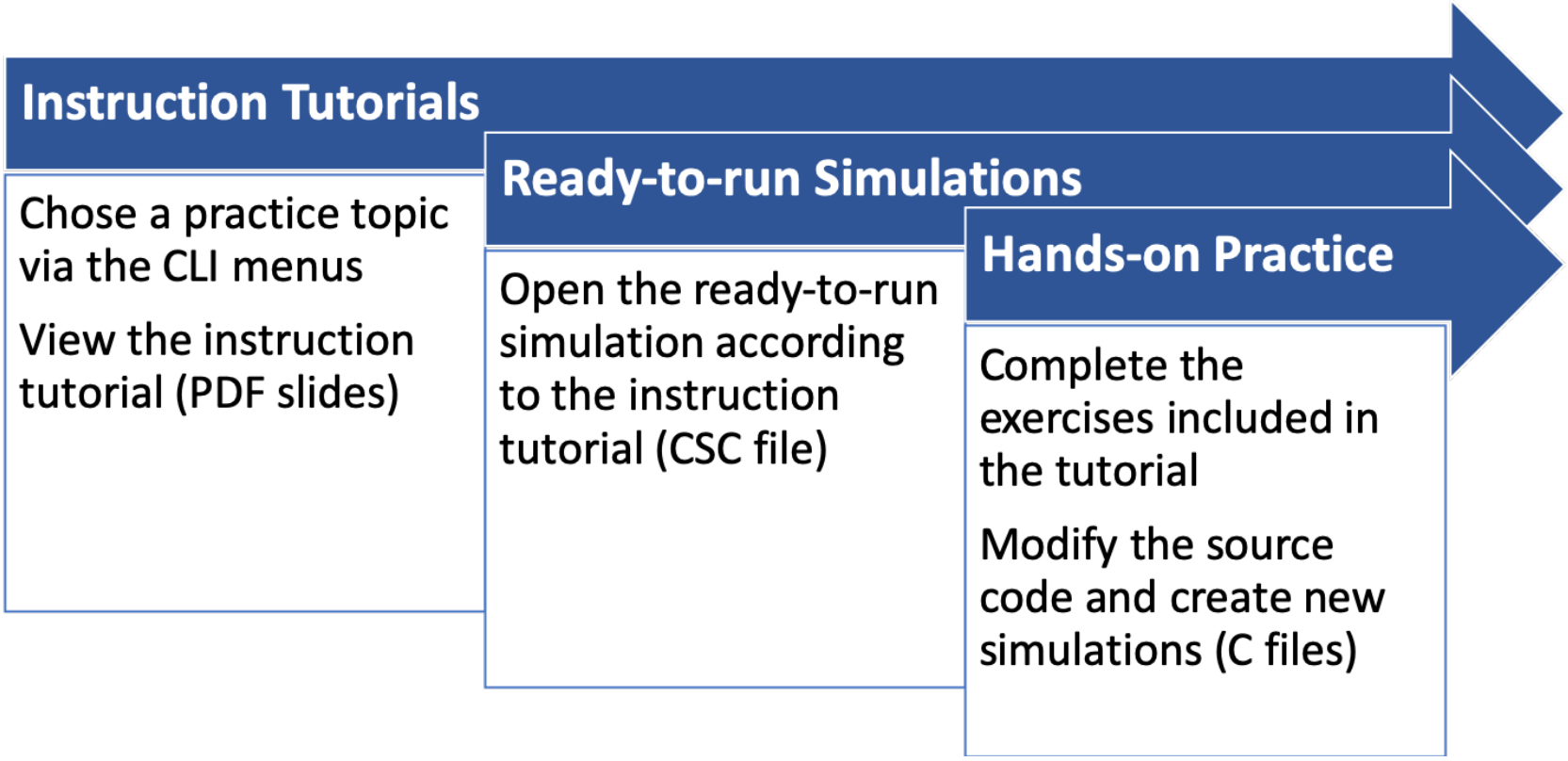
What is IoTrain-Sim

- IoTrain-Sim is a prototype system for IoT security training and education
  - Provides training tutorials, simulation examples, and hands-on exercises to users
  - Content is divided in fundamental and security training
  - Contiki OS and Cooja simulator are employed as tools to set up the simulation environment
- Due to the characteristics of the tools used, the training scope is currently limited to Wireless Sensor Networks (WSN) and RPL-based IoT networks

Content Overview



Training Flow



What is Contiki OS

- Contiki is an open source operating system for the Internet of Things
  - Helps connects tiny low-cost, low-power microcontrollers to the Internet
  - Powerful toolbox for building complex wireless systems
  - Supports fully standard IPv6 and IPv4, along with the recent low-power wireless standards, 6LoWPAN, RPL, CoAP
- Contiki applications are written in standard C, and with the Cooja simulator Contiki networks can be emulated before being written into hardware
- As there are plenty of examples in the Contiki source code tree to help users get started with their own code, and most have a corresponding Cooja simulation available, Contiki & Cooja are very suitable as tools for our training system
  - For more information, see http://www.contiki-os.org/

What is Cooja

- Cooja is a Contiki OS network simulator
  - An extensible Java-based simulator capable of emulating Tmote Sky (Z1 or other) nodes
  - Compiles Contiki for the native platform as a shared library, and loads the library into Java using Java Native Interfaces (JNIs)
- The code to be executed by the simulated nodes is the exact same firmware you may upload to physical nodes
- Provides a simulation environment that allows developers to see their applications run in large-scale networks or in extreme detail on fully emulated hardware devices

Fundamental Training Content

- Single Node
  - Basics of Contiki & Cooja
  - Actuation & Control
    - LED, button, timer programming in Contiki & Cooja
    - Hands-on exercises
  - Sensing
    - Temperature, humidity and light intensity sensor simulation
    - Sensor programming in Contiki & Cooja
- Networking
  - Communication
    - Broadcast tutorial

Security Training Contents

- Introduction
- Resource attacks
  - Direct attacks
    - Flooding – RPL DIS attack
  - Indirect attacks
    - Version number modification – RPL DODAG version attack
- Topology attack
  - Isolation
    - Blackhole attack
- Traffic attack
  - Misappropriation
    - Decreased rank attack

IoT Background Knowledge

IoT Overview

- What is IoT?
  - IoT (Internet of Things) is a network of physical devices, vehicles, home appliances, etc., embedded with electronics, software, sensors, actuators, and connectivity which enables these things to connect, collect and exchange data [1]
- Three-layer IoT architecture [2]
  1) Perception Layer → Gathers environment data
  2) Network Layer → Wired and wireless systems that process and transmit the input obtained by the perception layer supported by corresponding communication technologies
  3) Application Layer → Abstracted solutions that interact with the end users in order to satisfy their needs

IoT Elements

IoT Elements		Examples
Identification	Naming	EPC, uCode
	Addressing	IPv4, IPv6
Sensing		Smart Sensors, Wearable sensing devices, Embedded sensors, Actuators, RFID tag
Communication		RFID, NFC, UWB, Bluetooth, BLE, IEEE 802.15.4, Z-Wave, WiFi, WiFiDirect, LTE-A
Computation	Hardware	SmartThings, Arduino, Phidgets, Intel Galileo, Raspberry Pi, Gadgeteer, Z1, Tmote Sky
	Software	OS (Contiki, TinyOS, LiteOS, Riot OS, FreeRTOS, Android); Cloud (Nimbits, Hadoop)
Service		Identity-related (shipping), Information Aggregation (smart grid), Collaborative-Aware (smart home), Ubiquitous (smart city)
Semantic		RDF, OWL, EXI

IoT Common Standards

Application Protocols		DDS	CoAP	AMQP	MQTT	MQTT-SN	XMPP	HTTP REST
Service Discovery		mDNS				DNS-SD		
Infrastructure Protocols	Routing Protocols	RPL						
	Network Layer	6LoWPAN				IPv4/IPv6		
	Link Layer	IEEE 802.15.4						
	Physical/Device Layer	LTE-A	EPCglobal			IEEE 802.15.4		Z-Wave
Influential Protocols		IEEE 1888.3, IPSec				IEEE 1905.1		

IoT Challenges

- Availability
  - Hardware and software can be provided anywhere and anytime to users
- Reliability
  - Increase the success rate of IoT service delivery
  - Implemented in software and hardware at all the IoT layers
- Mobility
  - Connect users with their desired services continuously while on the move
- Performance
  - Continuously develop and improve service to meet customer requirements

IoT Challenges (cont.)

- Management
  - Efficient protocols needed to handle the management issues that will stem from the deployment of IoT in the coming years
- Scalability
  - Add new devices, services and functions for customers without negatively affecting the quality of existing services
- Interoperability
  - Handle a large number of heterogeneous devices that belong to different platforms
- Security and privacy
  - IoT devices require specific mechanisms to protect user privacy, as well as detect and block malicious activities

References

[1] https://en.wikipedia.org/wiki/Internet_of_things

[2] K. Zhao, L. Ge, “A survey on the Internet of Things security,” 9th International Conference on Computational Intelligence and Security (CIS), December 14-15, 2013, pp. 663–667.