HITBSecConf

Ghost Tunnel V2

Covert Data Exfiltration Channel to Circumvent Air Gapping

Yongtao Wang, Kunzhe Chai,Mingchuang Qin

PegasusTeam, 360 Security Technology

May 5, 2019

Who We Are

360 Security Technology is a leading Internet security company in Asia. Our core products are anti-virus security software for PC and cellphones.

Pegasus is a red team from 360 Security Technology focusing on wireless and IoT Security, we created 360SkyScan WIPS , we have achieved 100% success rate in our wireless pentest , our team was founded in 2015.

Agenda

- Introduction
- Previous research on Air-Gapped attack
- Ghost Tunnel V1 revision
- Ghost Tunnel V2 Introduction
- Ghost Tunnel V2 implementation

Introduction

- Air-Gapping
- Attack events

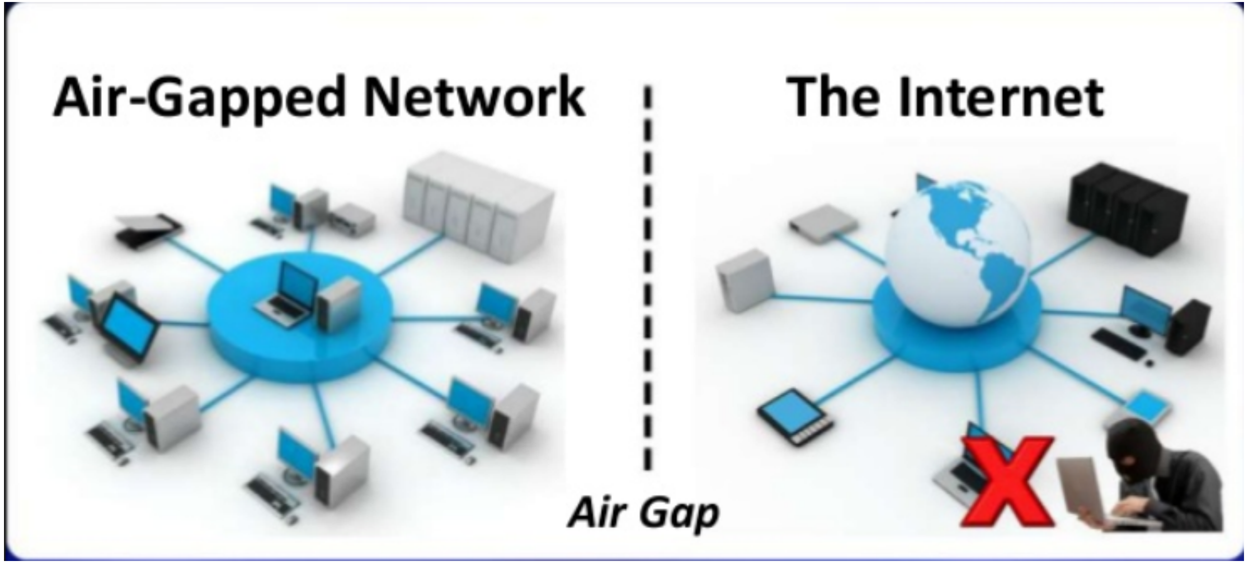
Air Gapping

- Air gapping
  - Wikipedia**:** “air gapping[1] is a network security measure employed on one or more computers to ensure that a secure computer network is physically isolated from unsecured networks, such as the public Internet or an unsecured local area network.[2] The name arises from the technique of creating a network that is physically separated (with a conceptual *air gap*) from all other networks.”

- Air gapping aims to avoid the intrusion and data leakage through network connections

Air-Gapped Network

- Considered to be the most secure

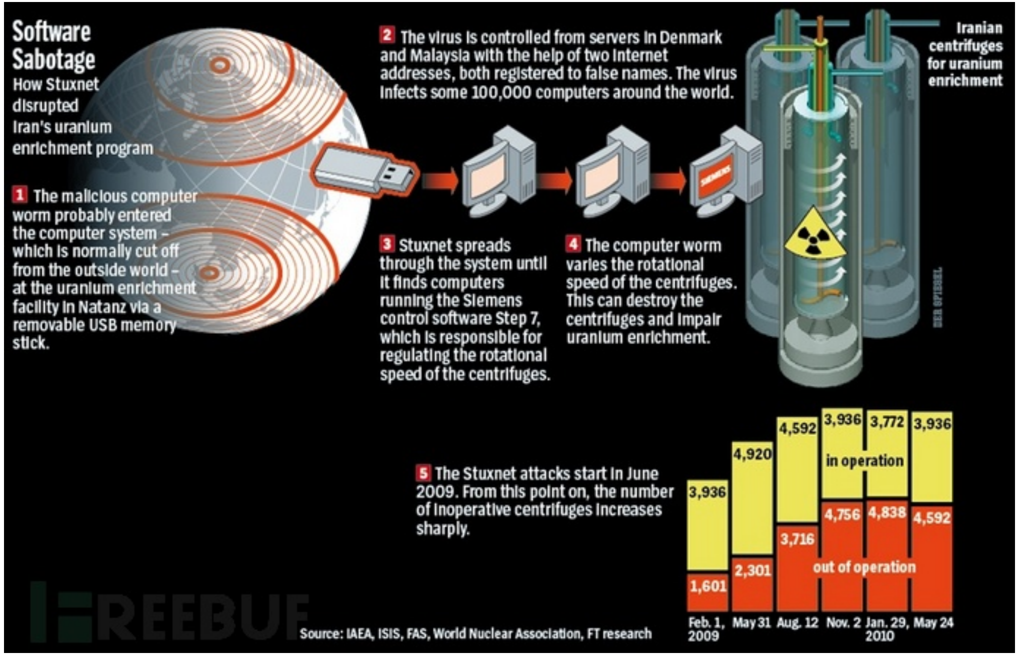


Nothing Is Impossible

- Attack Vectors
    - Malicious USB
    - Employee's laptop

Stuxnet Worm (2010)

- Attacking initiated via an infected USB drive
- Designed to sabotage centrifuges used at a uranium enrichment plant in Iran



Software Sabotage

How Stuxnet disrupted Iran's uranium enrichment program

1 The malicious computer worm probably entered the computer system – which is normally cut off from the outside world – at the uranium enrichment facility in Natanz via a removable USB memory stick.

2 The virus is controlled from servers in Denmark and Malaysia with the help of two Internet addresses, both registered to false names. The virus infects some 100,000 computers around the world.

3 Stuxnet spreads through the system until it finds computers running the Siemens control software Step 7, which is responsible for regulating the rotational speed of the centrifuges.

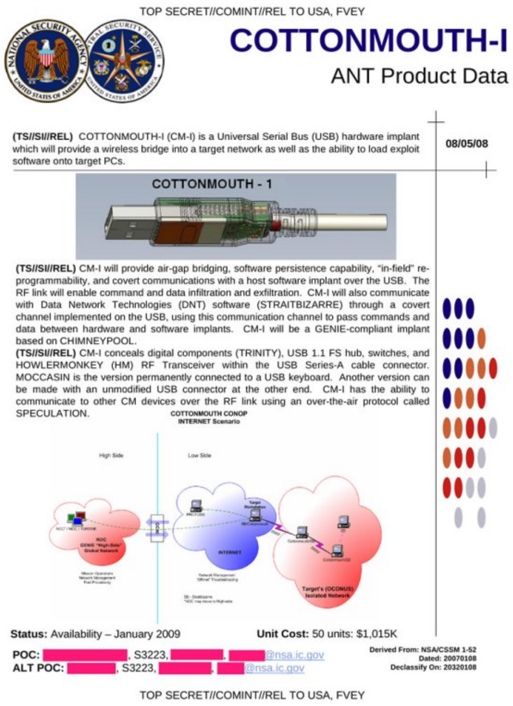
4 The computer worm varies the rotational speed of the centrifuges. This can destroy the centrifuges and impair uranium enrichment.

5 The Stuxnet attacks start in June 2009. From this point on, the number of inoperative centrifuges increases sharply.

Source: IAEA, ISIS, FAS, World Nuclear Association, FT research

NSA Leaks (2013)

- COTTONMOUTH-I
  - A USB hardware implant
  - Air-Gap bridging
  - Extracting data from targeted systems via RF signals



TOP SECRET//COMINT//REL TO USA, FVEY

COTTONMOUTH-I

ANT Product Data

08/05/08

(TS//SI//REL) COTTONMOUTH-I (CM-I) is a Universal Serial Bus (USB) hardware implant which will provide a wireless bridge into a target network as well as the ability to load exploit software onto target PCs.

(TS//SI//REL) CM-I will provide air-gap bridging, software persistence capability, "in-field" re-programmability, and covert communications with a host software implant over the USB. The RF link will enable command and data infiltration and exfiltration. CM-I will also communicate with Data Network Technologies (DNT) software (STRAITBIZARRE) through a covert channel implemented on the USB, using this communication channel to pass commands and data between hardware and software implants. CM-I will be a GENIE-compliant implant based on CHIMNEYPOOL.

(TS//SI//REL) CM-I conceals digital components (TRINITY), USB 1.1 FS hub, switches, and HOWLERMONKEY (HM) RF Transceiver within the USB Series-A cable connector. MOCCASIN is the version permanently connected to a USB keyboard. Another version can be made with an unmodified USB connector at the other end. CM-I has the ability to communicate to other CM devices over the RF link using an over-the-air protocol called SPECULATION.

Status: Availability – January 2009

Unit Cost: 50 units: $1,015K

POC: [redacted], S3223, [redacted], [redacted]@nsa.ic.gov
ALT POC: [redacted], S3223, [redacted], [redacted]@nsa.ic.gov

Derived From: NSA/CSSM 1-52
Dated: 20070108
Declassify On: 20320108

TOP SECRET//COMINT//REL TO USA, FVEY

Previous research on Air-Gapped attacks

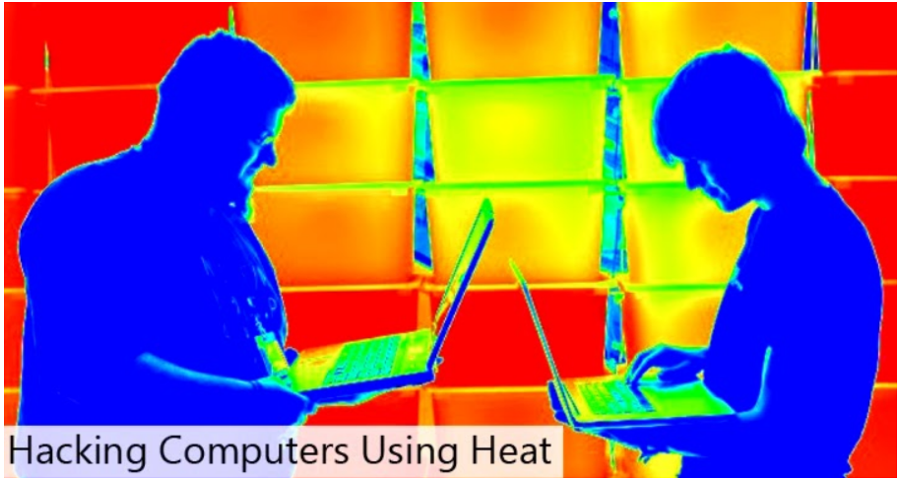
Previous research - 1

- Using radio frequencies to transmit data from a computer
  - Computer monitor
  - Mobile phone FM radio receiver

url: https://thehackernews.com/2014/10/airhopper-hacking-into-isolated.html

Previous research - 2

- A covert bi-directional communication channel between two close by air-gapped computers communicating via heat



url: https://thehackernews.com/2015/03/hacking-air-gapped-computer.html

Previous research - 3

- Data exfiltration via RF signal by attacking Siemens PLCs

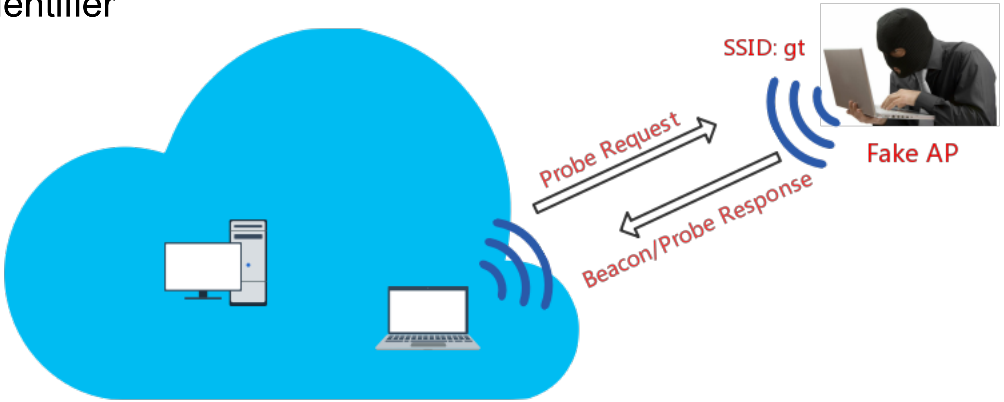
url: https://www.blackhat.com/eu-17/briefings.html#exfiltrating-reconnaissance-data-from-air-gapped-ics-scada-networks

Ghost Tunnel V1 Revision

A Covert Data Exfiltration Channel Using Wi-Fi

Ghost Tunnel V1 Revision

- A covert WiFi channel using Beacon, Probe Request, Probe Response
- A special SSID as the identifier



Ghost Tunnel V2

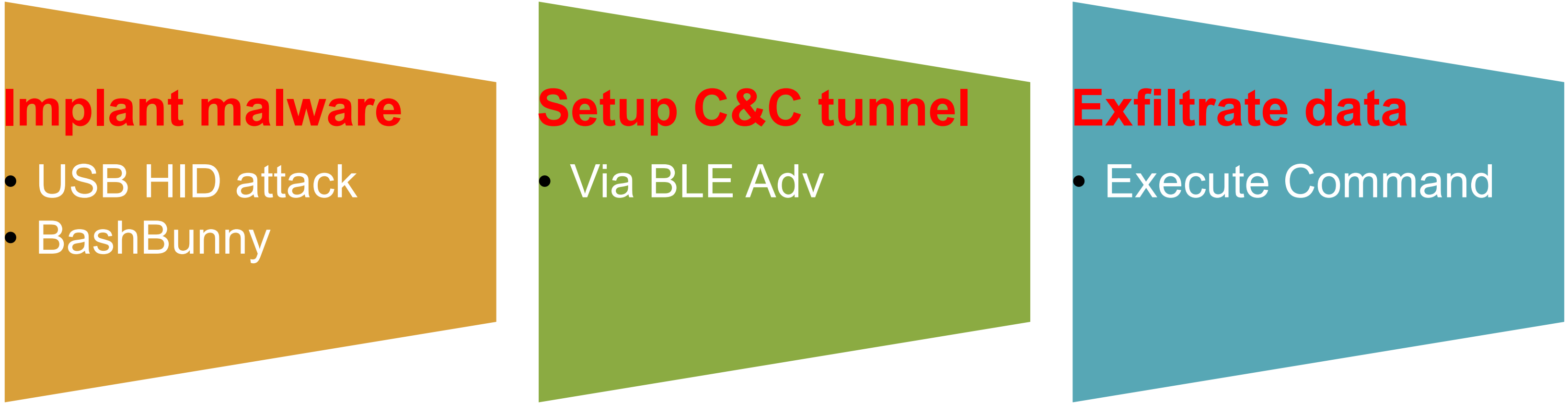
A Covert Data Exfiltration Channel Using Bluetooth Low Energy

Air-gapped Attack

- Implant
  - Malicious software/hardware

- A covert communication channel
  - Any medium that can carry data is possible

Ghost Tunnel V2



Implant malware

- USB HID attack
- BashBunny

Setup C&C tunnel

- Via BLE Adv

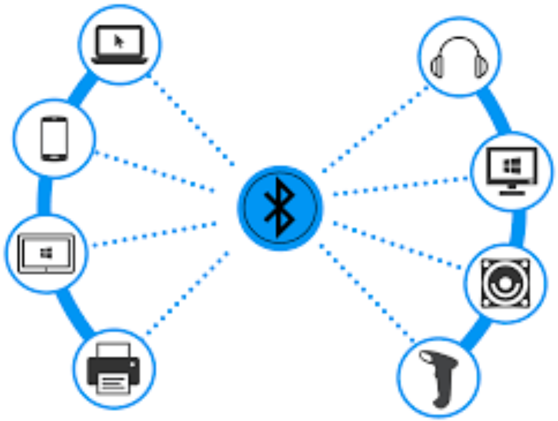
Exfiltrate data

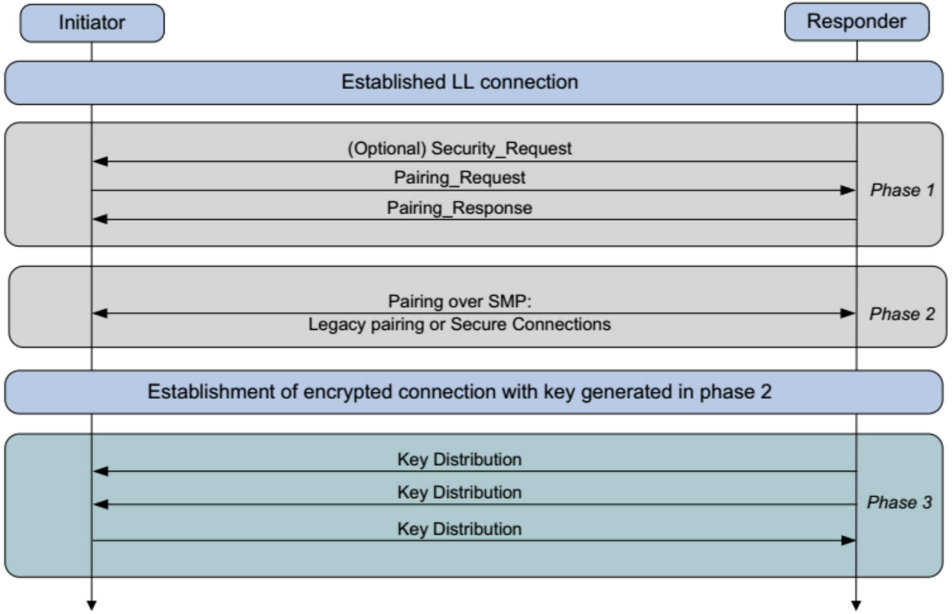
- Execute Command

Ghost Tunnel V2

- Can bypass firewalls
- Cross-Platform support
- Effective range up to 100 meters(@20dBm)

The Usual Bluetooth Connection Process





Initiator

Responder

Established LL connection

(Optional) Security_Request

Pairing_Request

Pairing_Response

Phase 1

Pairing over SMP:
Legacy pairing or Secure Connections

Phase 2

Establishment of encrypted connection with key generated in phase 2

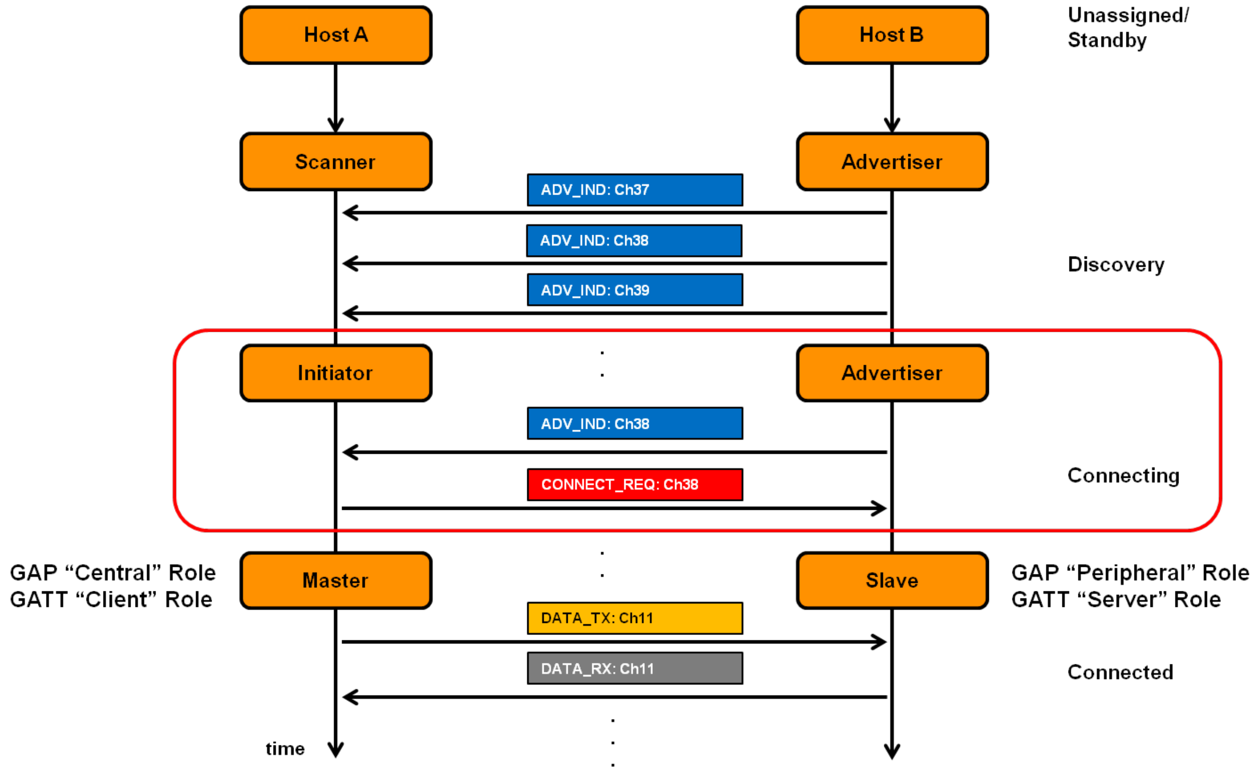
Key Distribution

Key Distribution

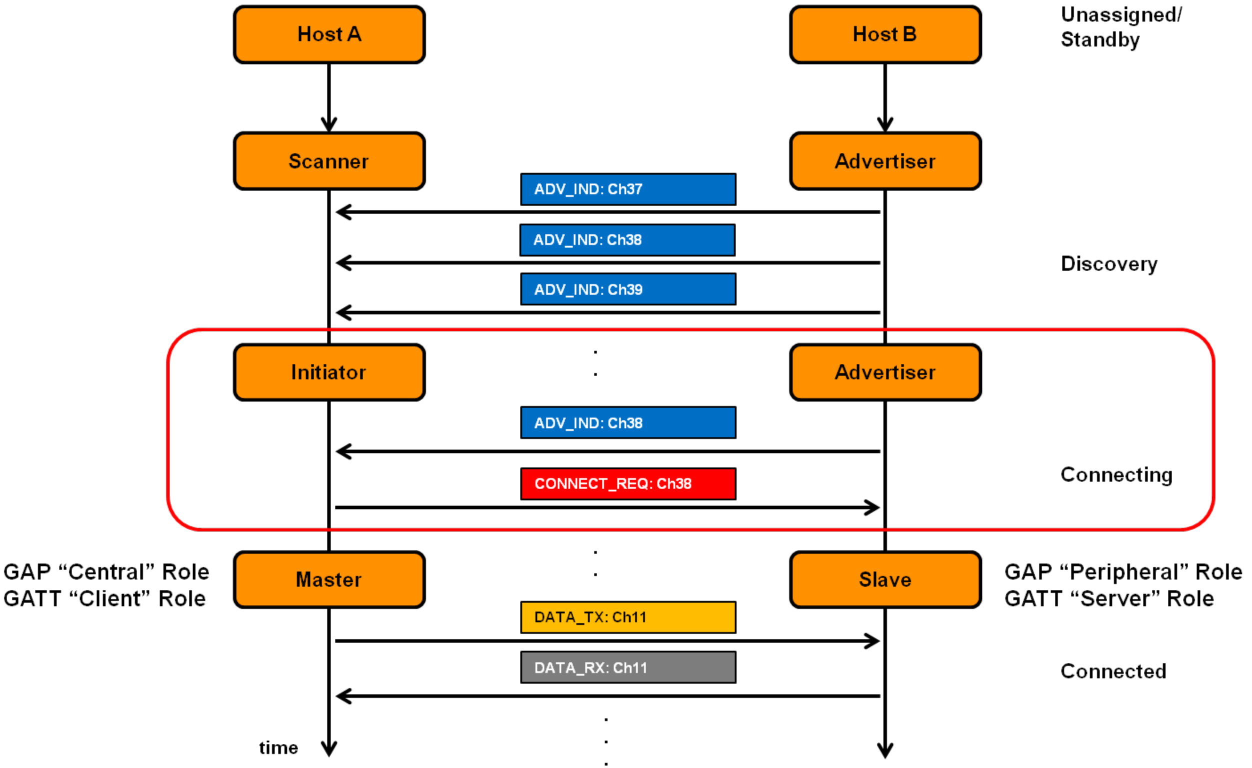
Key Distribution

Phase 3

Ghost Tunnel V2 – No Bluetooth Connection



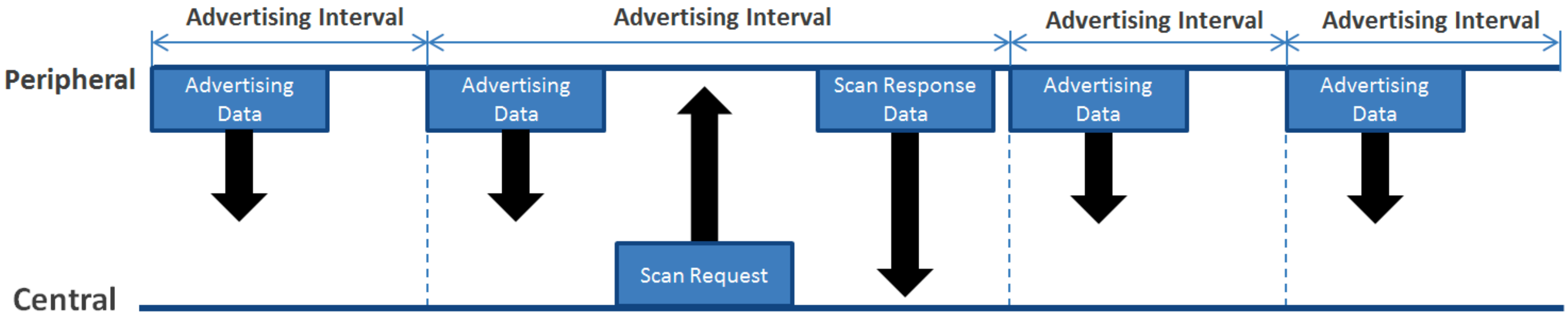
Bluetooth Low Energy State



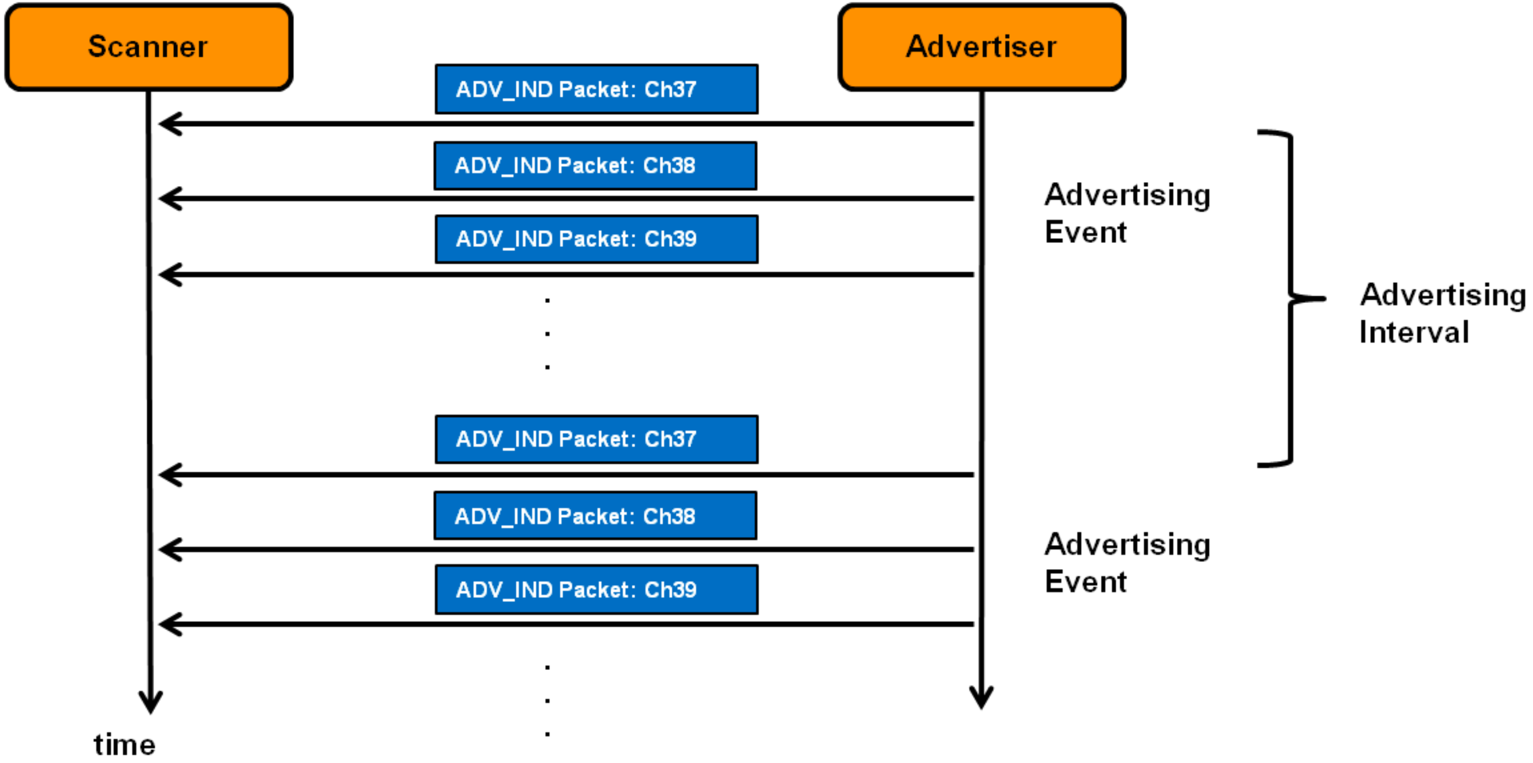
Bluetooth Low Energy Frames

Discovery	Connection	Data
ADV_IND	CONNECT_IND	DATA_TX
SCAN_REQ	CONNECT_REQ	DATA_RX
SCAN_RSP	CONNECT_RSP	……
……	……	

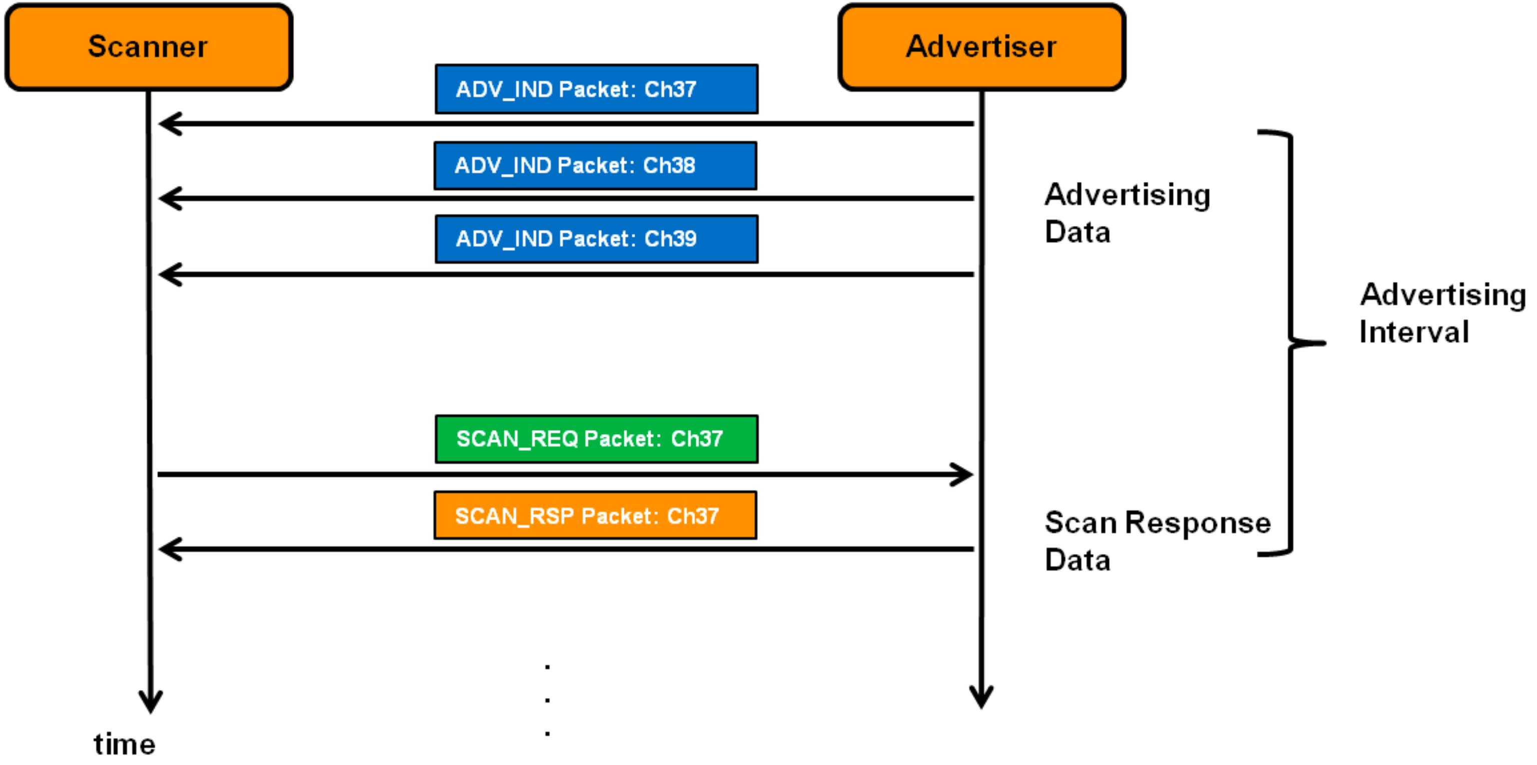
Scanning for BLE Networks



Passive Scanning for BLE Networks

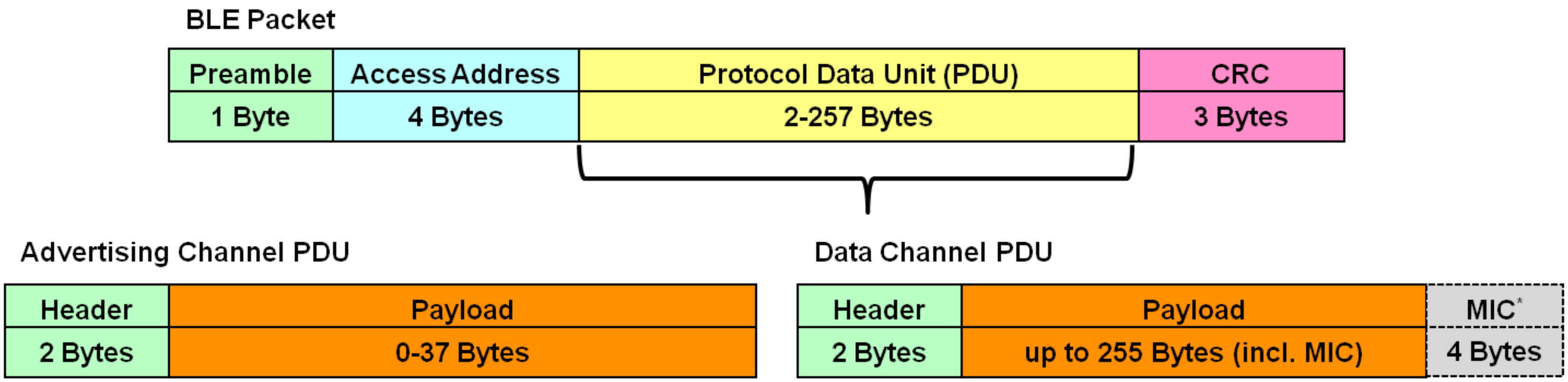


Active Scanning for BLE Networks



Ghost Tunnel V2 Implementation

Bluetooth Low Energy Packet



BLE Packet

Preamble	Access Address	Protocol Data Unit (PDU)	CRC
1 Byte	4 Bytes	2-257 Bytes	3 Bytes

Advertising Channel PDU

Header	Payload
2 Bytes	0-37 Bytes

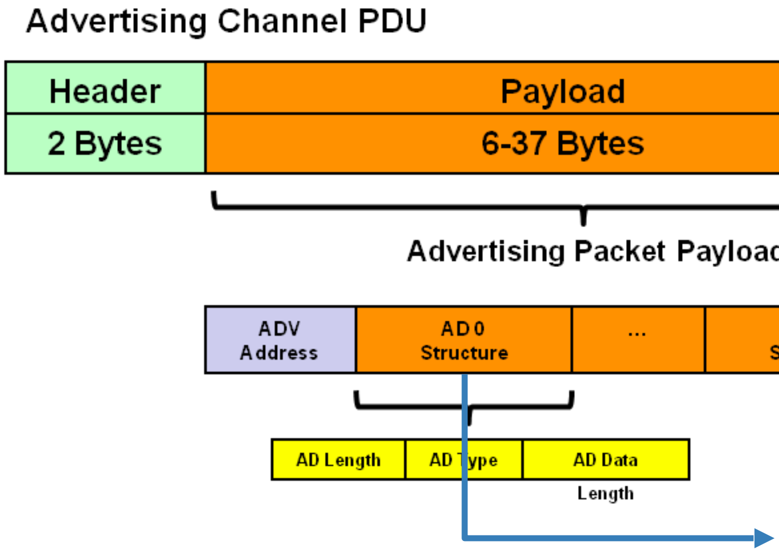
Data Channel PDU

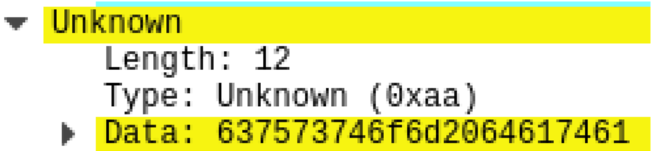
Header	Payload	MIC*
2 Bytes	up to 255 Bytes (incl. MIC)	4 Bytes

Ref: BT Specification v4.2, Vol. 6, Part B, Sec. 2.1

*Message Integrity Check: Included as part of Payload if used (for security)

Advertising Channel PDU

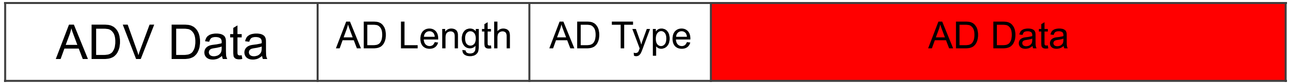




```
▼ Unknown
      Length: 12
      Type: Unknown (0xaa)
    ▶ Data: 637573746f6d2064617461
```

Advertisement Data Structures

Advertisement Data



Key Problem

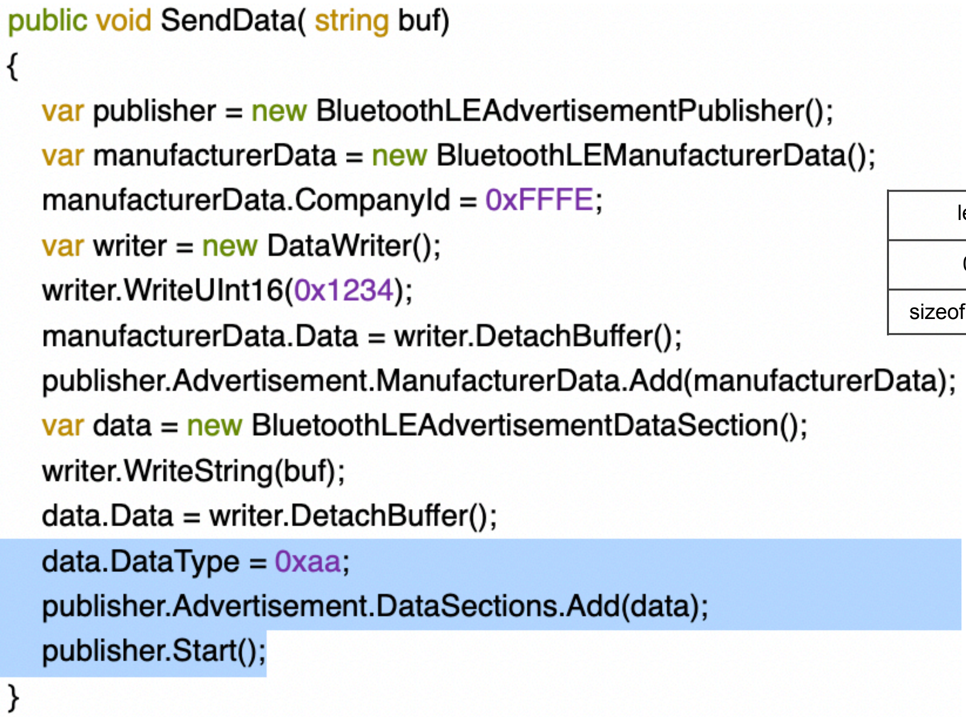
- How to send and receive Bluetooth Low Energy data frames through local Bluetooth interface in user space ?

- Bluetooth interface mode
  - BR/EDR (audio…)
  - BLE (IoT,wearable devices…)
  - …

Data Format

Identify	length	type	Company id	Data
	0x05	0xFF	0xFFFE	0x1234
Custom payload	len	type	Custom data	

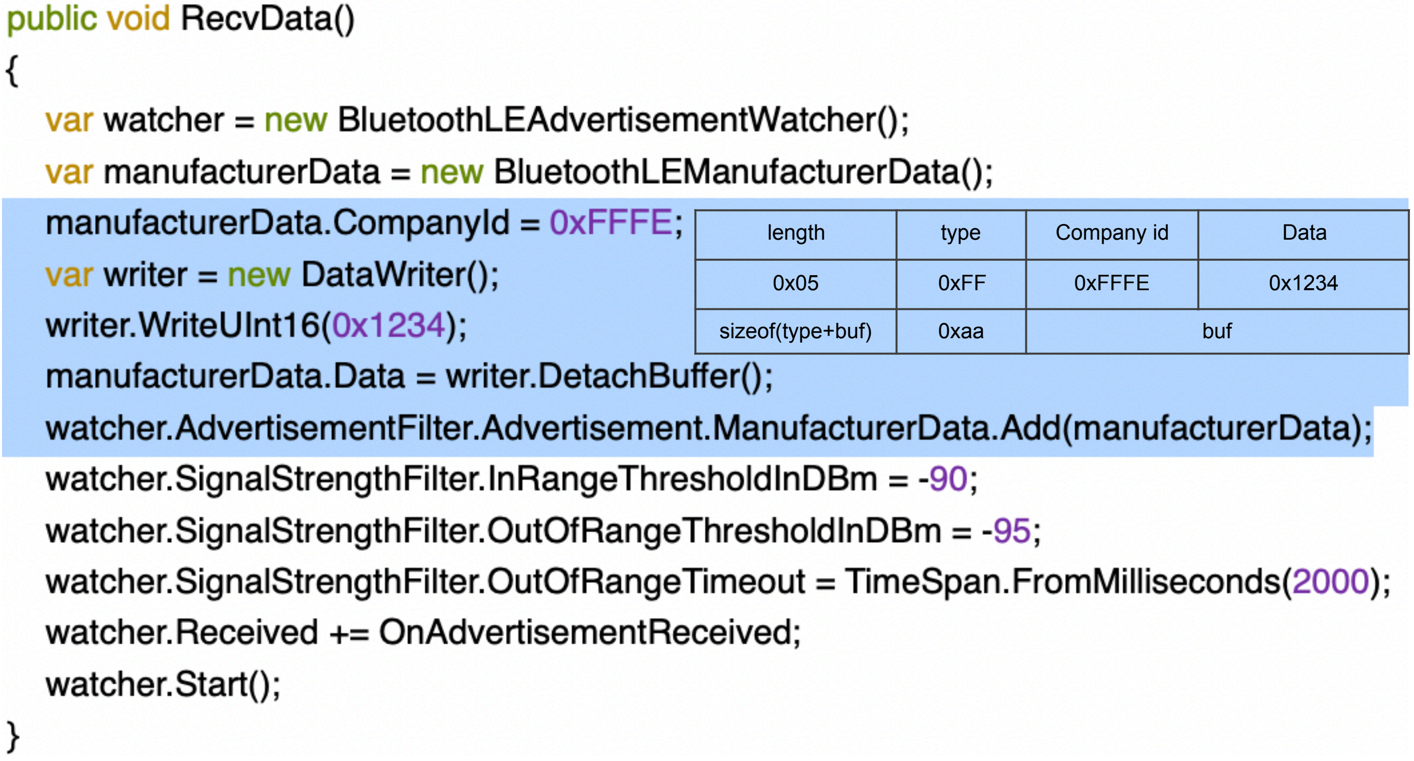
Send BLE Data



```
public void SendData( string buf)
{
    var publisher = new BluetoothLEAdvertisementPublisher();
    var manufacturerData = new BluetoothLEManufacturerData();
    manufacturerData.CompanyId = 0xFFFE;
    var writer = new DataWriter();
    writer.WriteUInt16(0x1234);
    manufacturerData.Data = writer.DetachBuffer();
    publisher.Advertisement.ManufacturerData.Add(manufacturerData);
    var data = new BluetoothLEAdvertisementDataSection();
    writer.WriteString(buf);
    data.Data = writer.DetachBuffer();
    data.DataType = 0xaa;
    publisher.Advertisement.DataSections.Add(data);
    publisher.Start();
}
```

length	type	Company id	Data
0x05	0xFF	0xFFFE	0x1234
sizeof(type+buf)	0xaa	buf	

Receive BLE Data

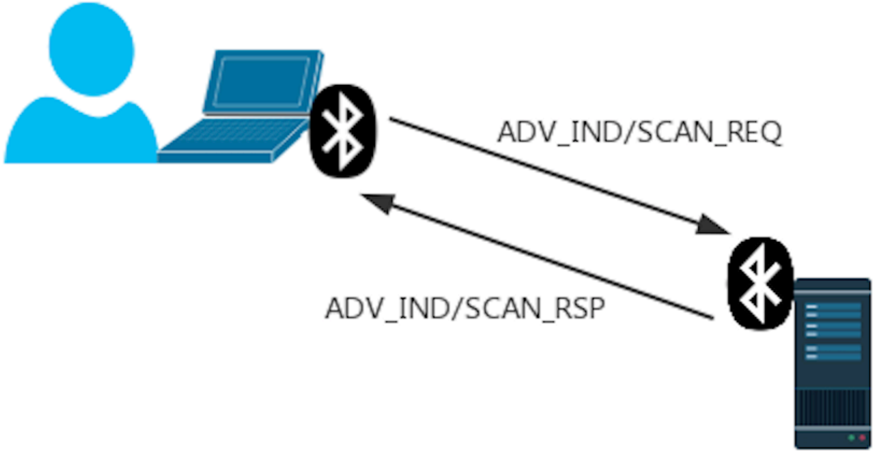


```
public void RecvData()
{
    var watcher = new BluetoothLEAdvertisementWatcher();
    var manufacturerData = new BluetoothLEManufacturerData();
    manufacturerData.CompanyId = 0xFFFE;
    var writer = new DataWriter();
    writer.WriteUInt16(0x1234);
    manufacturerData.Data = writer.DetachBuffer();
    watcher.AdvertisementFilter.Advertisement.ManufacturerData.Add(manufacturerData);
    watcher.SignalStrengthFilter.InRangeThresholdInDBm = -90;
    watcher.SignalStrengthFilter.OutOfRangeThresholdInDBm = -95;
    watcher.SignalStrengthFilter.OutOfRangeTimeout = TimeSpan.FromMilliseconds(2000);
    watcher.Received += OnAdvertisementReceived;
    watcher.Start();
}
```

length	type	Company id	Data
0x05	0xFF	0xFFFE	0x1234
sizeof(type+buf)	0xaa	buf	

Ghost Tunnel V2– No Connection

- A covert BLE channel using ADV_IND,SCAN_REQ,SCAN_RSP.
- A special Custom manufacture ID as the identifier



Thanks! & QA?

github@360pegasusteam