An Overview of Practical Attacks on BLE Based IOT Devices and Their Security

Publisher: IEEE

Sode Pallavi ; V Anantha Narayanan All Authors

Abstract:
BLE is used to transmit and receive data between sensors and devices. Most of the IOT devices employ BLE for wireless communication because it suits their requirements su... **View more**

Metadata

Abstract:
BLE is used to transmit and receive data between sensors and devices. Most of the IOT devices employ BLE for wireless communication because it suits their requirements such as less energy constraints. The major security vulnerabilities in BLE protocol can be used by attacker to perform MITM attacks and hence violating confidentiality and integrity of data. Although BLE 4.2 prevents most of the attacks by employing elliptic-curve diffie-Hellman to generate LTK and encrypt the data, still there are many devices in the market that are using BLE 4.0, 4.1 which are vulnerable to attacks. This paper shows the simple demonstration of possible attacks on BLE devices that use various existing tools to perform spoofing, MITM and firmware attacks. We also discussed the security, privacy and its importance in BLE devices.

Published in: 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)

References

Citations

Keywords

Metrics

More Like This

Date of Conference: 15-16 March 2019

DOI: 10.1109/ICACCS.2019.8728448

Date Added to IEEE *Xplore*: 06 June 2019

Publisher: IEEE

ISBN Information:

Conference Location: Coimbatore, India

ISSN Information:

Contents

SECTION I.
INTRODUCTION

Bluetooth Low Energy

BLE popularly known as Bluetooth Smart or Bluetooth 4 technology was designed by the Bluetooth Special Interest Group (Bluetooth SIG) for the applications in the areas such as health care, fitness, beacons, security, home and entertainment industries. References [5], [13], [14] shows the applications of BLE in agriculture, health care. It is a wireless technology that operates at 2.4GHz [12].

It works for shorter ranges and consumes very less energy than the other wireless protocols. The BLE protocol is similar to the Bluetooth. BLE is not compatible with classic Bluetooth. The recent devices and mobile operating systems such as Android, IOS, Windows Phone and Blackberry, as well as macOS, Linux, Windows 8 and Windows 10 supports both BLE and Bluetooth.

In this paper, section 2 and section 3 discusses the pairing methods and pairing process in BLE 4.0, 4.1. Section 4 and 5 presents the MITM attacks, firmware attacks that we performed on "Nordic Thingy 52" and a fitness tracker. Section 6 presents precautions to be taken by manufactures and users to prevent these attacks. And finally, section 7 gives the conclusion and future work.

SECTION II.
PAIRING METHODS IN BLE-4.0

BLE devices mainly supports three types of pairing mechanisms. They are:

1. Just Works
2. Passkey
3. Out of Band

1) Just Works:

This is the pairing mechanism in which the devices doesn't ask for anything to authenticate to other device, Which means that anyone can easily connect to the device. Hence it is the weakest pairing method with no security against the Man-In The-Middle attacks.

- The devices that don't have display or keypad will use this "Just Works" as pairing method.
- The default temporary key (TK) is zero.

2) Passkey:

In this method one device will display the 6-digited passkey and the other device that wants to connect to has to enter that displayed key. When both are matched the connection will be established and this key will be used as temporary key(TK). Hence this method can partly prevent the MITM attacks by employing authentication. However, the devices that use this method are still vulnerable to attacks.

3) Out of Band:

This is the pairing mechanism in which TK will be shared between the devices by other wireless technologies for example, NFC. The length of TK can be less than or equal to 128 digits. Hence the TK used is strong and also provides strong security if the OOB channel is secure as it prevents attacker from eavesdropping. Hence when compared with the other two the OOB pairing method is more secure.

SECTION III.
BLE PAIRING PROCESS AND KEYS USED

The pairing process in the 4.0 and 4.1 devices, is also known as LE Legacy Pairing. The devices that use LE Legacy Pairing exchange a key known as Temporary Key (TK). The value of the temporary key and its exchange depends on the employed pairing method. This temporary key is again used to generate a" Short Term Key" shortly known as STK. STK is the key used to encrypt the connection and messages exchanged.

- Thus the "confidentiality" of the messages/data mainly "depends on TK and STK" in the LE Legacy Pairing.
- Hence if TK or STK is compromised or known to the attacker they can decrypt the data being exchanged.

PAIRING PROCESS

1) Phase One:
In this phase initially, the device that wants connection will send a pairing-request to the device to which it wants to communicate with. Then both devices will exchange their requirements regarding authentication, link key size, their Input-output capabilities (availability of display, keypad) and some requirements for bonding. Hence based on the data exchanged in this phase the device will choose pairing methods suitable for their I/O capabilities. But all the data exchanged here is not encrypted. Hence if the attacker begins eavesdropping from this phase then it will lead to MITM attacks.

2) Phase Two:
After the completion of phase 1, the devices will start generating and/or exchanging the Temporary key(TK). Then they exchange the mrand and confirm values. Using this values both the devices will ensure/verify if both are using are same TK. After this, they start generating the Short-Term Key(STK) using the values exchanged. Then encryption is done using STK.

3) Phase Three:
This is an optional phase and is started after completion of phase2 and if the bonding requirements

are exchanged in first phase. Here, many keys specific to transport were shared between each other. The following figure Fig. 1 shows the BLE pairing process

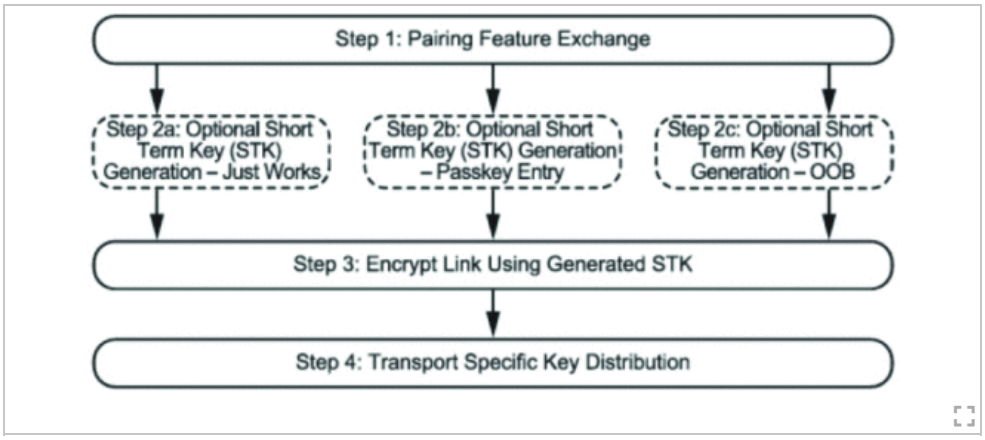

Fig. 1.
LE Legacy pairing

SECTION IV.
ATTACKS ON NORDIC THINGY 52

For the attack purpose, the device "Nordic Thingy 52" with BLE 4.0 was used. This has many sensors that collect the temperature, humidity, Pressure, Air quality, Color intensity, motion of device, percentage of CO_2 etc.

- Since it doesn't have I/O capabilities, it uses "Just Works" as pairing mechanism.

A. Passive Interception

If the transmission between sender and receiver is not encrypted, then it can be known to the eavesdroppers. Now a-days we don't need any expensive hardware to intercept the BLE communication.

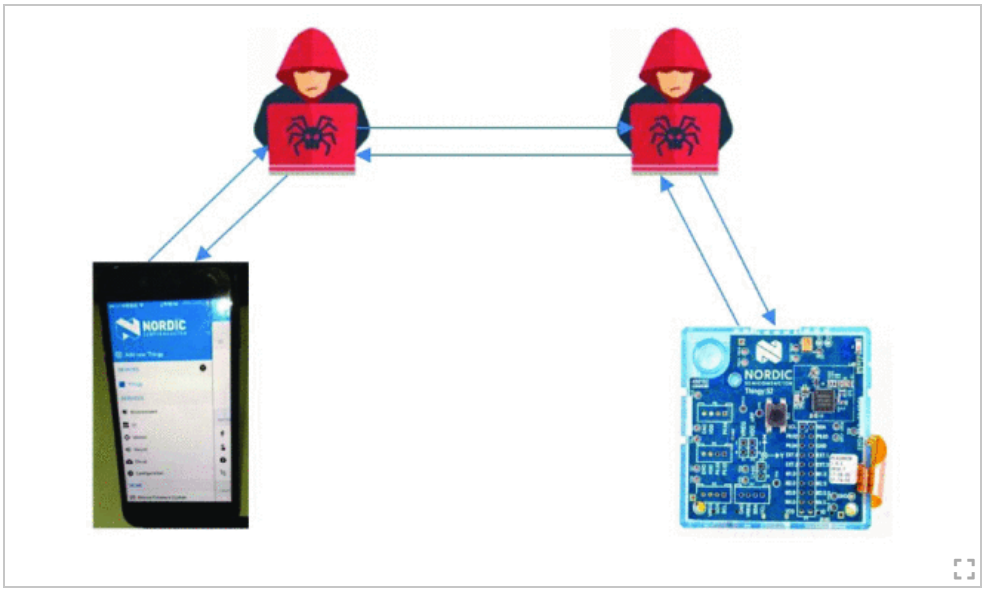

Fig. 2.
MITM Attack

The device will send the values collected by sensors to the mobile application "NORDIC THINGY".

The figure Fig. 2 shows the basic idea behind the Attack.

1) No Authentication:
As per the observation, Nordic thingy 52 doesn't have any authentication for user verification. Anyone with the NORDIC THINGY mobile application can directly connect to the device if it is powered ON and not already connected to another device. If the Attacker connects to the device, they can get and read all the sensor values either in the app or using packet analyzer tools such as "UBERTOOTH" or "WIRESHARK". since the transmission between the sender and the receiver was improperly encrypted, we were able to intercept all the data.

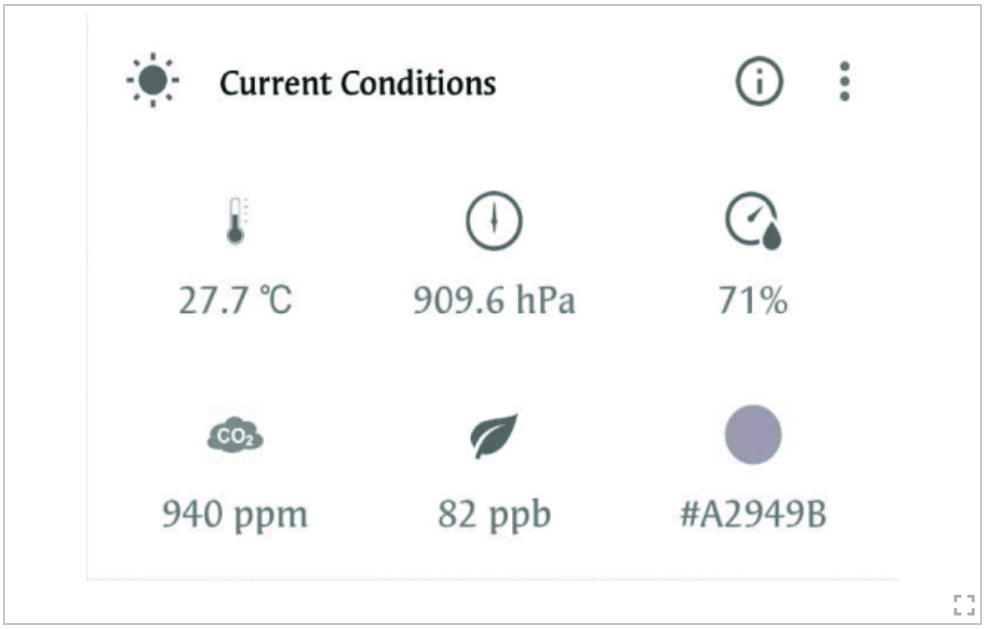

Fig. 3.
Sensor Values

2) Capturing the Data during transfer:
The passive attack and active attack was implemented using an open source tool named BtleJuice[12] . It clones the original device and advertises more frequently than the original device and hence the user can first see the cloned (duplicate) device set by attacker with the same name and characteristics as of original device which makes the user believe that it is the device and connect to it instead of intended one. Next, the tool helps us to setup a proxy between device and application thus forwarding and presenting the data exchanged. Fig. 2 gives the complete view of attack.

Before cloning the MAC address of the original device we need to "clone" all of the service and characteristics including the handle numbers. Else, the GATT structure of the mobile OS will no match to it. Resulting to improper communication between the mobile application and the device.

The figure Fig. 4 shows the sample subset of values that are sent by the device to the attacker's mobile application.

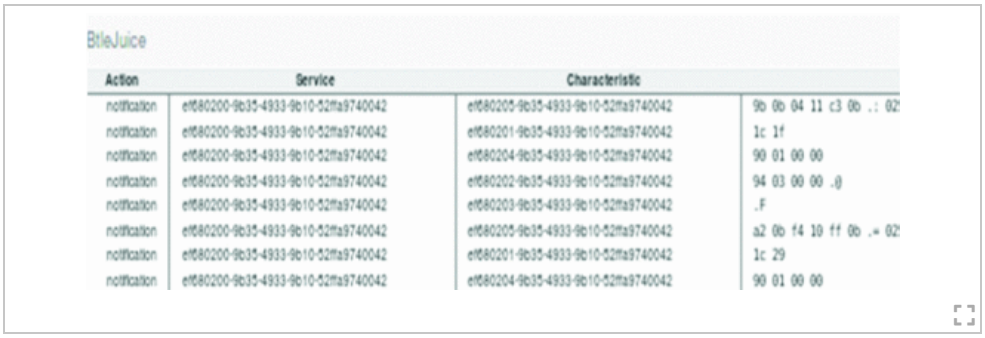

Fig. 4
Intercepted Data

B. Active interception

Active interception of BLE communication means that the attacker is able to alter the data transmission between the user device and its application. The ability of an attacker to modify and include malicious data that is shared between device and application can lead to many attacks. This type of attacks are called as Man in the middle (MiTM) attacks.

The attacks depend on the type of data and the way it is getting transmitted, and also the actions that can be made by them once they receive that particular data.

1) Data manipulation:
A Nordic Thingy 52 device was connected to a mobile application with BLE 4.0. Even though the data transmitted was properly encrypted with AES, still the device is able to allow the attacker to inject the malicious instructions.

Here the attacker can connect to the device and receives the original values from it, alter it and send the false data to the mobile application i.e., the intruder acts like a proxy between the device and application. BtleJuice provides an interface to write the values to the device. Hence, we performed the attack by changing the original text sent by device, it was possible to display any text on the app and vice versa.

The figure Fig. 5 shows changing of color to "RED".

Fig. 5.
Data Manipulation

If the similar device is used in case of medical or scientific applications the attacker can easily modify the critical values as heart rate, temperature etc. which will lead to severe problems such as improper diagnosis and treatment.

2) Replay:
In the same example, once BtleJuice captures the data from the device the attacker can analyze them and replay the same values. For example, Initially the device will show blue color. The app can send the command to the device to change its color to green if the device is connected.

The figure Fig. 6 shows the replay attack using BtleJuice. By analyzing the data, the attacker can find the command that includes the hexa decimal value of color and replay the command that changes the color from green to blue which indicates the disconnection of the device.

3) Command injection:
For example, Thus Man-in-the-Middle attack allowed us to intercept all the data transmission between authenticated devices. Then, we detached the original command sent by the Thingy application, and instead added some other false commands. The command injection includes changing number of milliseconds after which the date has to be captured and sent to app.

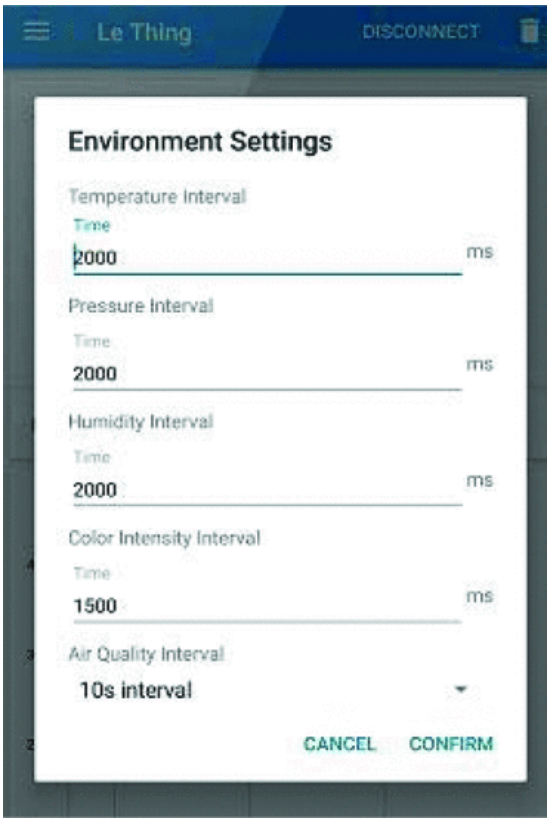

Fig. 7.
Command Injection

The figure Fig. 7 shows the manipulation of command instructing to collect the data for every given ms values.

Example: If it used in an application that collects data for every hour and alerts if temperature/gas is higher than threshold. In this case if the attacker injects the command that instructs the device to collect the data for every 4 hours which will lead to delay in alerts and may lead to severe damage.

SECTION V.
ATTACK ON FITNESS TRACKER

The fitness trackers that use BLE for communication are used for attacks. In order to ensure security of users the device used here, and full firmware versions are not mentioned. The vulnerabilities that are exploited are being stated to device manufacture companies and they took the steps to prevent these attacks. Before performing the attack, the firmware details of device have to be known. The attack works, provided that there is an official update and it also requires social engineering.

We used an android app called "GADGET BRIDGE" to perform Firmware downgrading and firmware modification attacks. The figure Fig. 8 shows the firmware details before attack.

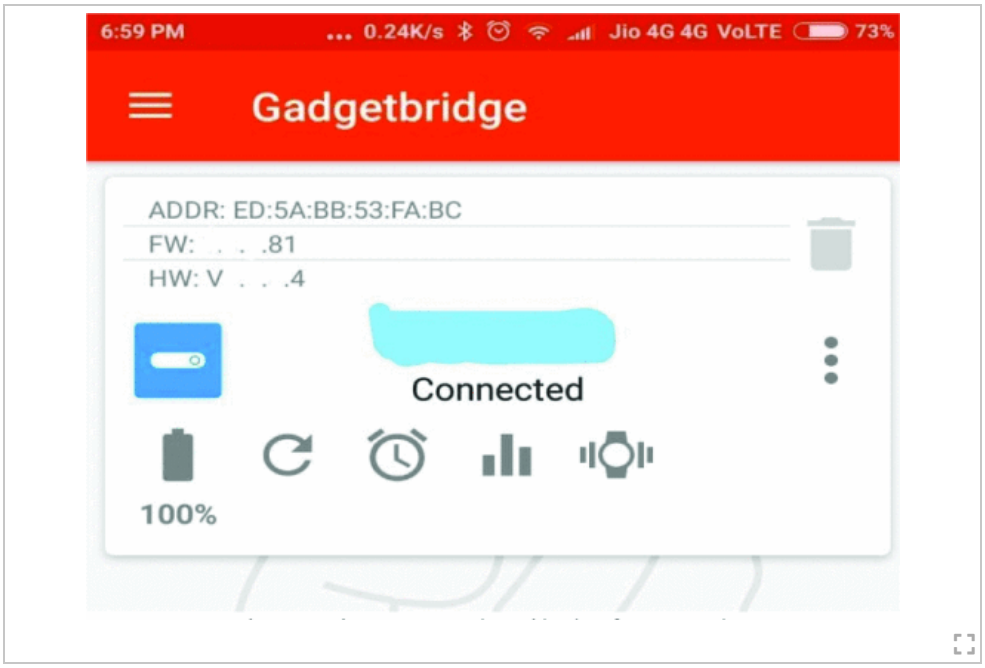

Fig. 8.
Firmware details before update

In this demonstration we show how the attacker can use this app maliciously.

As part of it the first step is to get firmware from the official mirror APK. The current firmware version is x.x.x.81. we downgraded it to x.x.x.53 hence the firmware degradation attack is also possible. The figure Fig. 9 shows that the firmware has been degraded to x.x.x.53 version.

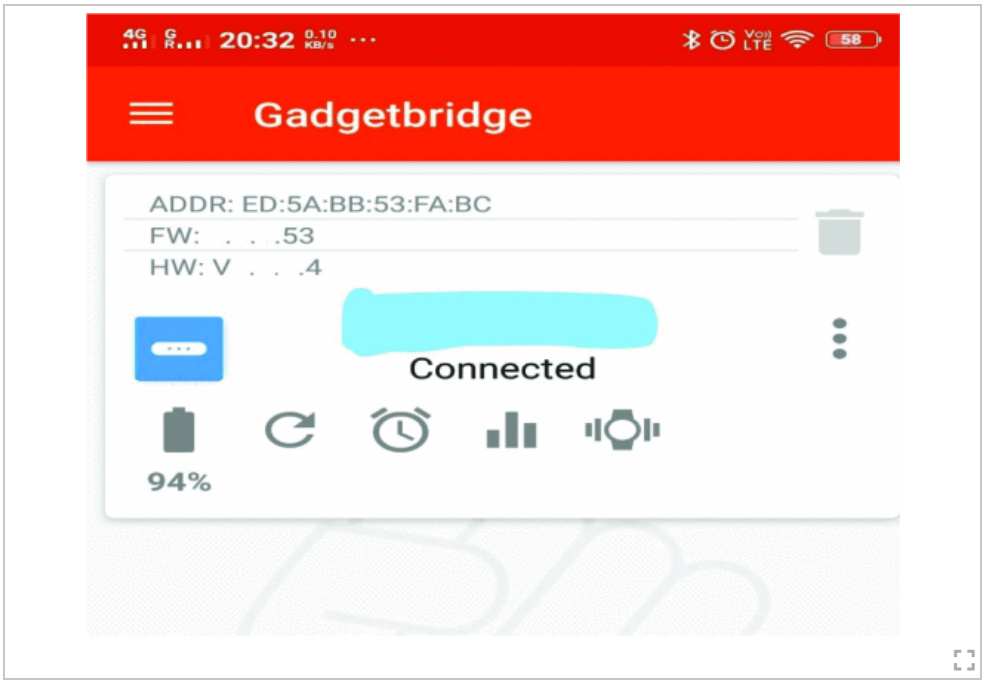

Fig. 9.
Firmware details after downgrading

- we performed reverse engineering on the firmware to get the basic details. Later we injected the malicious code by making changes to the original firmware into the device.

- The modified firmware has been flashed into the device successfully since the device has no check sum verification for firmware files as a result the device stopped working due to the malicious firmware in it.

Most of the wearable devices doesn't have much security checks in it. Instead, these checks are being implemented in the official app that the device connects to. Which means, the device is completely depending on the application it is connected to. Hence if the attacker can create a similar android application to fool the user by sending a connection request, firmware update request then it enables them to even inject malwares into device and alter the behavior of the tracker.

SECTION VI.
PRECAUTIONS TO PREVENT THE FIRMWARE ATTACKS

1) By manufacturers:

- Ensure that the device is connecting to the official intended application by using fingerprints.

- Prevent the devices from connecting to the malicious apps.

- Employ integrity checks before updating the firmware.

- The firmware should be updated only when there is an official update.

- Do not allow the user to downgrade or use the older firmware versions.

- Educate the user about security and privacy.

2) By Users:

- Use the devices that provide strong authentication.

- Do not install and use malicious applications for the devices that contains sensitive data.

- Wearables have less authentication. The attacker can send false connection request to the device to confuse. Hence ensure the connection and authenticate only if you are the one, sending the pairing request.

SECTION VII.
CONCLUSION AND FUTURE WORK

IOT devices mostly deal with the sensitive data such as temperature, pressure and also user's personal data such as heart-rate, sleep etc. Since most of the IOT devices that collects these sensitive data use BLE for transmission it is important to improve security of this protocol to ensure privacy of the users. [1], [2] and [7] to [10] presents a survey on security and privacy analysis of BLE and IOT devices.

Although many steps were taken to prevent these kinds of attacks we can still see that there are many devices that are still vulnerable to attacks easily. check the references section [4], [6], [11] for more attacks that can be performed on BLE device.

And with the firmware attack we can conclude that even if the BLE protocol is secure but the improper implementation of security checks makes the device vulnerable.

As shown in the attacks the user is getting fooled easily. So, the users must be aware of their activities. Hence it is responsibility of both the manufacturers and users to take the necessary precautions to avoid the attacks. As we also found that the length and value of TK is predictable by use of tools such as crackLE, which makes BLE even more vulnerable that the attacker can get STK that is used for encryption and hence get all the exchanged information. Hence, the future work includes proposing a protocol that adds an extra layer of security before the generation of TK, that provides a large and more random TK that is generated using the devices fingerprints and also provides strong authentication to prevent these kinds of attacks.

Authors
Figures
References
Citations
Keywords
Metrics