Lab6: Robot Bluetooth Sniffing

Pre-Requisite Knowledge and Skills

1. Understand basic knowledge of Sparki robot.
2. Be able to compile and upload a code.
3. Be able to run Python3

Learning Objective:

1. Understand how Bluetooth works in Sparki.
2. Understand Server and Client application model.
3. Understand the Malware.
4. Understand and explain the Sniffing attack.

Recommended Running Environment and Software:

1. Computers Running Windows OS, OSX, or Linux
2. Python3
3. SparkiDuino IDE

Instructional Material:

1. Sparki Robot
2. Instructions of this activity
3. Server and Client applications

Video Demonstration:

1. to be developed

Lab Assessment:

1. Exercises
2. Quiz

Lab Instructions

This lab requires two (2) PCs, running Window OS, OS X, or Linux, and one of two PCs should be Bluetooth capable. It also requires Sparki and Bluetooth card included in Sparki package. In addition, Python3 should be installed in both PCs to run the applications.

For the convenience, let’s label PCs as PC1 and PC2. PC1 is a Sparki robot user and PC2 is an attacker. Sparki robot user uses Bluetooth connection to controller the robot. For this purpose, Sparki user should create the Bluetooth connection and upload the code. First, we will need to connect PC1 to Sparki using Bluetooth. Bring the Bluetooth card from the box and plug it into the Bluetooth communication pins in the Sparki as shown in Figure 1.

The next step is Python 3 installation. Use following links to install the Python 3 for the corresponding Operating System (OS).

Woidnows 10: https://phoenixnap.com/kb/how-to-install-python-3-windows
Mac OS X: https://docs.python-guide.org/starting/install3/osx/
Linux: https://docs.python-guide.org/starting/install3/linux/

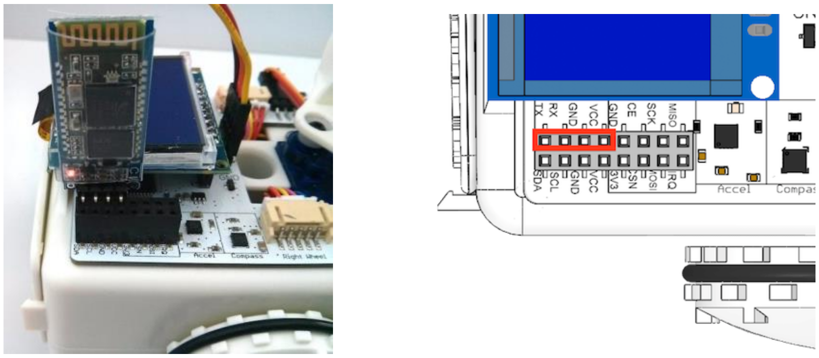


Figure 1. Bluetooth card setup

Now, we need to write a code for Sparki to check if there is any command received on Bluetooth connection and take an action if command is arrived. The structure of code is consisted of two part, read the communication channel and make a move based on reading.

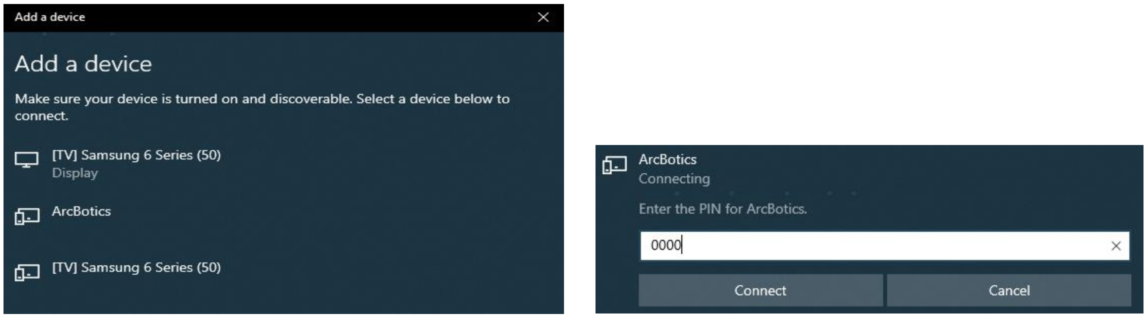
Note: Instructor can provide a code fully or partially depending on the level of student group. Sample code is included in the lesson package.

Step 1: Complete the code shown in *Appendix*. Write a code to move Sparki forward with “e”, left with “s”, right with “f”, stop with “d”. Instructor can ask student to make more moves such as backward, gripper gripping, and etc. for the challenge.

Step 2: Compile and upload the code. Then, disconnect the cable and turn the Sparki on.

Now we need to create the Bluetooth connection between Sparki robot and PC1.

Step 3: On PC1, open the Bluetooth setting. “Windows > Setting > Devices > Bluetooth & other devices”, then click “Add Bluetooth or other device”. Find the device called “ArcBotics” and enter the PIN, “0000” (four zeros) for ArcBotics as shown in Figure 2.
Note: This setup may differ based on OS.



Add a device

Add a device

Make sure your device is turned on and discoverable. Select a device below to connect.

[TV] Samsung 6 Series (50)
Display

ArcBotics

[TV] Samsung 6 Series (50)

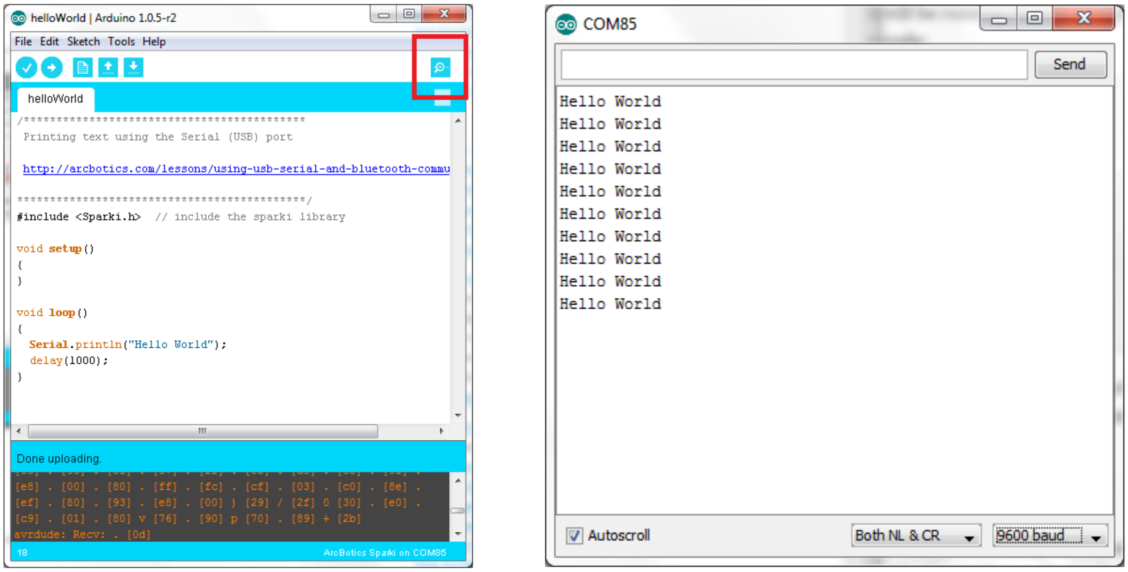
ArcBotics
Connecting

Enter the PIN for ArcBotics.

0000

Figure 2. Bluetooth setup in Windows 10

Step 4: Once the Sparki is connected through Bluetooth, Sparki receives a command using serial port. Open serial monitor by selecting “Tools > Serial Monitor” or clicking icon on the top right corner of the IDE as shown in Figure 3.


helloWorld | Arduino 1.0.5-r2
File Edit Sketch Tools Help
helloWorld

```
/*******************************************
 Printing text using the Serial (USB) port

 http://arcbotics.com/lessons/using-usb-serial-and-bluetooth-commu

*******************************************/
#include <Sparki.h>  // include the sparki library

void setup()
{
}

void loop()
{
  Serial.println("Hello World");
  delay(1000);
}
```

Done uploading.

```
[e8] . [00] . [80] . [ff] . [fc] . [cf] . [03] . [c0] . [8e] .
[ef] . [80] . [93] . [e8] . [00] ) [29] / [2f] 0 [30] . [e0] .
[c9] . [01] . [80] v [76] . [90] p [70] . [89] + [2b]
avrdude: Recv: . [0d]
```

18
ArcBotics Sparki on COM85

(a) Serial monitor icon.

```
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
Hello World
```

☑ Autoscroll   Both NL & CR   9600 baud

(b) Serial monitor

Figure 3. Serial monitor

Once the serial monitor is open, send a command to move Sparki.

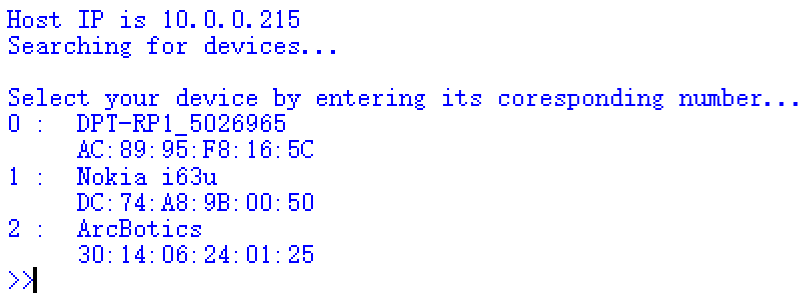
Step 5: In this step, we will run a Malware on PC1 to find the Sparki commands. Use following scenario for the rest of activities.

Scenario: PC1 is a controller unit of the Sparki using Bluetooth and PC2 is a sniffer to find the Sparki commands. PC1 and PC2 should be in the same network. If wireless router is available, connect PC1 and PC2 to wireless router in wired or wireless. Assuming that DHCP server is running on the router. If wireless router is not available, use switch to connect PCs. Connect PCs to the switch and assign IP address to each PCs. Go to “Window > Setting > Network & Internet > Change adapter options”. Right click on “Ethernet” and click Properties and select “Internet Protocol Version 4 (TCP/IP4)” and click “Properties”. Then, select “User the following IP address” and assign IP address and subnet mask properly. Use following information to assign IP address and subnet mask.

PC1:	IP address: 192.168.10.5	Subnet mask: 255.255.255.0
PC2:	IP address: 192.168.10.10	Subnet mask: 255.255.255.0

Then, click OK and click OK on “Ethernet Properties” window to make it effective. You may need to check if IP address is effective using Command Prompt. Press “window button + R” together and type “cmd” and click “OK”. And type “ipconfig” to check if they are all assigned correctly.

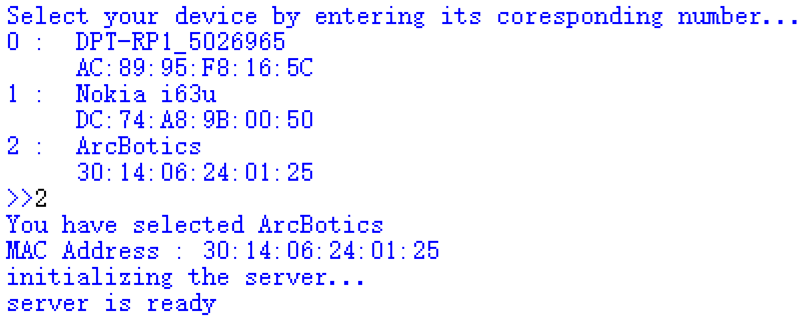
Run the Malware “server.py” on PC1. Once it is running, it will display server IP address and searching running devices on PC1. Select the number with ArcBotics and enter. Then, server is ready. As shown in Figure 4 and 5.



```
Host IP is 10.0.0.215
Searching for devices...

Select your device by entering its coresponding number...
0 :  DPT-RP1_5026965
     AC:89:95:F8:16:5C
1 :  Nokia i63u
     DC:74:A8:9B:00:50
2 :  ArcBotics
     30:14:06:24:01:25
>>
```

Figure 4. Device selection at server



```
Select your device by entering its coresponding number...
0 :  DPT-RP1_5026965
     AC:89:95:F8:16:5C
1 :  Nokia i63u
     DC:74:A8:9B:00:50
2 :  ArcBotics
     30:14:06:24:01:25
>>2
You have selected ArcBotics
MAC Address :  30:14:06:24:01:25
initializing the server...
server is ready
```

Figure 5. Display when server is ready

Step 6: Now we need to run “client.py” on PC2. Once you run it, all the command form PC1 to Sparki can be seen at PC2.

Step 7: Two students can be grouped in one team. One student writes a new code to move Sparki and hide the commands from the other team member. While student moves Sparki, the other team member is spying the PC1 and find the commands for Sparki moves.

Discussion

- **What is a malware and what are the examples of malware?**

  Malware is any type of software that can damage the device, steal data, and/or cause any harmful activities. Examples are Virus, Trojans, Spyware, Worms, Ransomware, Adware, and etc.

- **What does sniffing attack do and how does it work in this case?**

  Sniffing attack intercept the data by listening to the communication channel. All transmitted and/or received packets can be collected and any data included in the packets can be retrieved. In this lab, we use client and server model. Server is running in the target and it sniffs the data transmitting on the Bluetooth channel and sends it to the client.

- **How we can prevent sniffing attack and/or malware?**

Sniffing attack in the lab uses server-client model. Either one of them, server or client. Should be running on the target machine. In fact, all malware should be executing and running in the target machine. The first thing, you may need to do is not to run any suspicious code or click the link. Other type of sniffing attack is monitoring the packets in the network. In this case, it does not need malware running at the target machine. Then, encryption plays an important role to hid the data.

Appendix

Bluetooth code for Sparki

```
#include <Sparki.h> // include the sparki library

String inputString; //make an empty String called inputString
boolean returnFlag; //flag to check for carriage return
char commArray [10]; //array to store communication
int arrayCounter = 0; //integer to count through commArray

void setup()
{
 Serial1.begin(9600);
}

void loop()
{
 readComm();
 makeMove();
}

void makeMove()
{
 for(int i = 0; i <= 9; i++)
  {
  if(commArray[i] == 'f')
   {
   sparki.moveForward();
   delay(1000);
   sparki.moveStop();
   }
   else if (commArray[i] == 'r')
   {
   sparki.moveRight(90);
   }
   else if (commArray[i] == 'l')
   {
   sparki.moveLeft(90);
   }
   else if (commArray[i] == 's')
   {
   sparki.moveStop();
   delay(1000);
   }
   else if (commArray[i] != 0) //in case it's a character sparki doesn't know
   {
   Serial1.print("I'm sorry, I didn't understand the command- ");
   Serial1.println(commArray[i]); //send the character back
   }
 }
 memset(commArray, 0, sizeof(commArray)); //clear out commArray
}

void readComm()
{
 while (Serial1.available())
 {
 int inByte = Serial1.read();
 if ((char)inByte == 'n')
   {
   returnFlag = true;
   arrayCounter = 0;
   }
   else
   {
   if(inByte == 32) //if it's a blank space
   {
   arrayCounter ++; //increment array counter to store in new array space
   }
   else
   {
   //add the character to the arrayCounter space in commArray
   commArray[arrayCounter] = (char)inByte;
   }
  }
 }
}
```