

IoT Wireless & Cloud Computing Emerging Technologies course projects

1. Bluetooth Scan project

In this project, I start by installing two Bluetooth applications BLE Scanner and Bluetooth 4.0 Scanner from android play store and test its functions by both apps in order to see how it performs scanning and collects information. When ready, and I conduct the scanning and record the measured RSSIs, device names, farthest estimated distance, Bluetooth packet types, PHY channel used while advertising, and checked if the message was connectable.

Below I attached the screenshots of the results in both Bluetooth Apps:

I. BLE Scanner app Result

Below are the screenshots of the results using the BLE Scanner application; the screenshots ordered according to the steps from **scanning** → **Near by available device** → **connection to a device** → **History**. One good thing that I observe on both Bluetooth scanner app is you can export all the scanning information as **csv** and **json** files, as it shows from the screenshot pic **Export Data** in the History part.

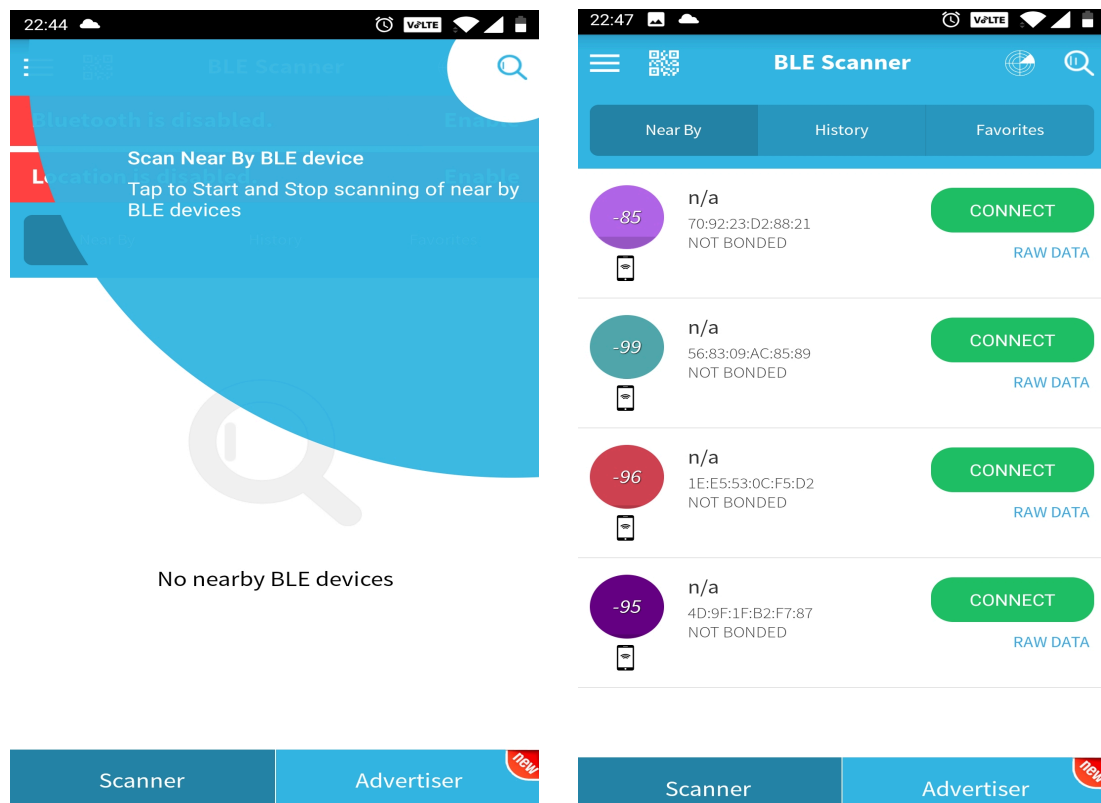


Fig1. Before the start of scanner and the second pic shows available, near by devices

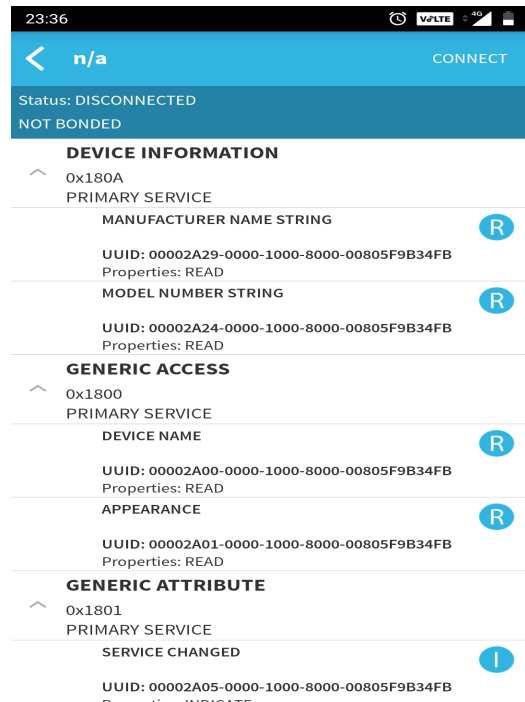
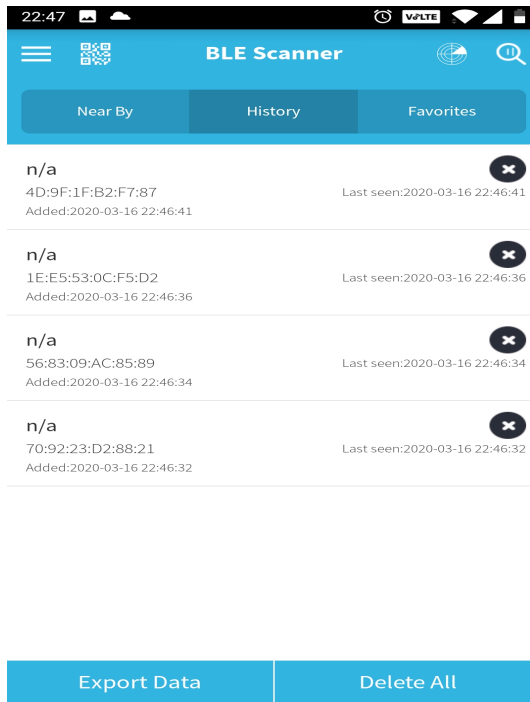


Fig2. First pic shows history of the last available devices and the 2nd pic shows connected device info.

II. Bluetooth 4.0 Scanner App Results

Below are the screenshots of the results using Bluetooth 4.0 Scanner App; the screenshots ordered according to the steps from **scanning** → **Scan Result**. Here once the scan result are done, the App allows you to download the obtained result information as **csv** and **json** files, which can be seen on Fig 4 below in excel form

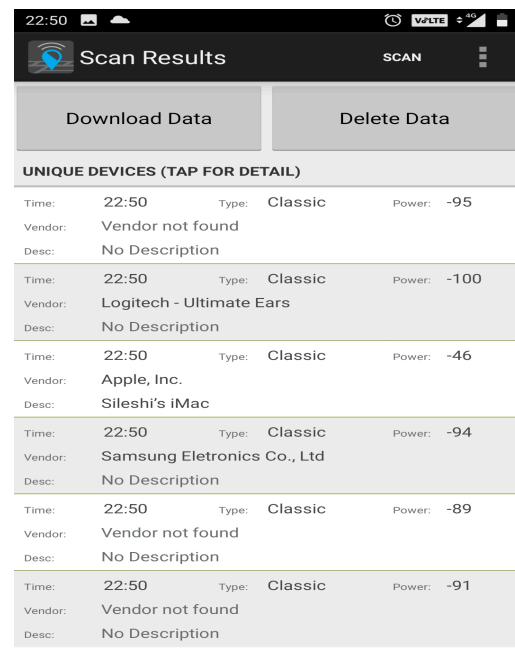
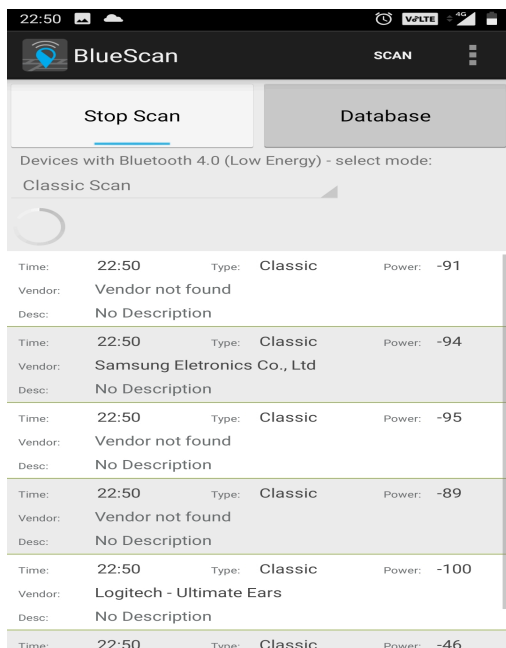


Fig 3. First pic shows the scanning process and the second one shows the result to be downloading.

Below is the result of all the required information data for the Bluetooth scan project as a screenshot from the excel data, on top of the required information there are some other additional data such as local date time, time stamps of the scan, device address and so on..

1	datetime_local	datetime_gmt	timestamp	deviceAddress	localAddress	localAndroid	deviceName	oui_name	scanMod	latitude	longitude	altitude	geo_provider	rsi
2	20200316 22:50:29	20200316 20:50:29	1.584E+12	4D:9F:1F:B2:F7	02:00:00:00:c7ceef29bd4	None			1	-1	-1	-1	bluescan_nu	-93
3	20200316 22:50:30	20200316 20:50:30	1.584E+12	70:92:23:D2:88	02:00:00:00:c7ceef29bd4	None			1	-1	-1	-1	bluescan_nu	-92
4	20200316 22:50:32	20200316 20:50:32	1.584E+12	C4:73:1E:63:98	02:00:00:00:c7ceef29bd4	None	Samsung Ele		1	-1	-1	-1	bluescan_nu	-90
5	20200316 22:50:37	20200316 20:50:37	1.584E+12	10:40:F3:E4:DC	02:00:00:00:c7ceef29bd4	Sileshis i	Apple, Inc.		1	-1	-1	-1	bluescan_nu	-46
6	20200316 22:50:38	20200316 20:50:38	1.584E+12	88:C6:26:90:7A	02:00:00:00:c7ceef29bd4	None	Logitech - Ult		1	-1	-1	-1	bluescan_nu	-100
7	20200316 22:50:43	20200316 20:50:43	1.584E+12	6C:E7:AD:4D:11	02:00:00:00:c7ceef29bd4	None			1	-1	-1	-1	bluescan_nu	-95

Fig 4, obtain result downloaded from the Bluetooth scan app

2. Wi-Fi Network Analyser experiments

For the Wi-Fi Network Analyser experiments, just like the same that I did in the Bluetooth project, I start with the installation of the network analyzer app and test its functions. After the installation of the network analyzer app I start conducting the measurements and record the device's internal and external IP addresses and gateway's IP address, link data rate that the AP provides, PHY channel and bandwidth you are using, number of detectable Wi-Fi channel signals and the strongest and weakest signal strength detected. Finally, I conduct a Ping to the gateway and record the results as it follows below.

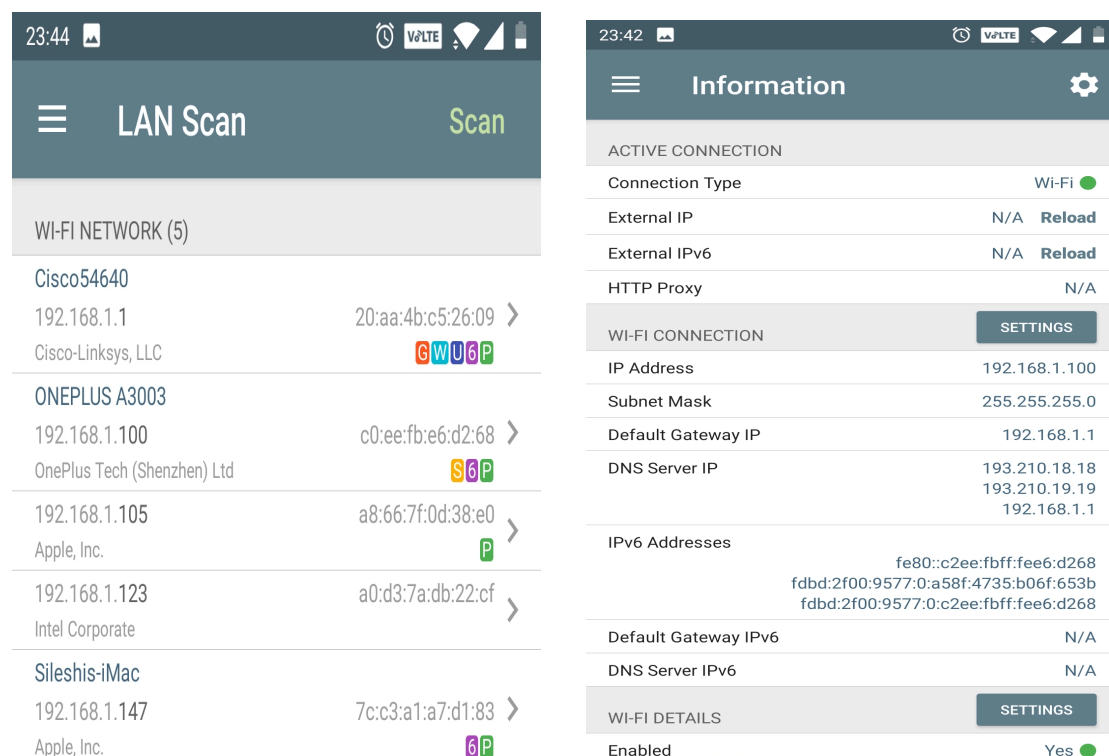


Fig 5. First pic shows the Wi-Fi Network Analyser app scan and the second one shows the obtained LAN scanning information

Information	
WI-FI DETAILS	
Enabled	Yes
Connection State	Completed
SSID	sile_wifi
BSSID	20:aa:4b:c5:26:0b
Vendor	Cisco-Linksys, LLC
Channel	11
MAC	c0:ee:fb:e6:d2:68
Speed	72 Mbps
Signal Strength	-59 dBm
CELL DETAILS	
Data State	Disconnected
Data Activity	Dormant
Roaming	No
SIM State	Ready
SIM Operator Name	Saunalahti
SIM MCC/MNC	244/05 (fi)
Operator Name	Saunalahti

Ping		
192.168.1.1 (10)		
10	192.168.1.1	4.3 ms
9	192.168.1.1	6.5 ms
8	192.168.1.1	4.2 ms
7	192.168.1.1	4.2 ms
6	192.168.1.1	4.1 ms
5	192.168.1.1	4.1 ms
4	192.168.1.1	3.5 ms
3	192.168.1.1	27.4 ms
2	192.168.1.1	4.7 ms
1	192.168.1.1	3.0 ms

Fig 6. Shows the screenshots of the obtain result information's such as; SSID, BSSID, he speed of the network and signal Strength and so on, the second pic shows while I am ping one LAN with IP ID 192.168.1.1.

Finally all LAN which was scanned by the Wi-Fi Network Analyser App for this Wi-Fi scan experiment project can be seen below with there Wi-Fi Signal properties below as a screenshot.

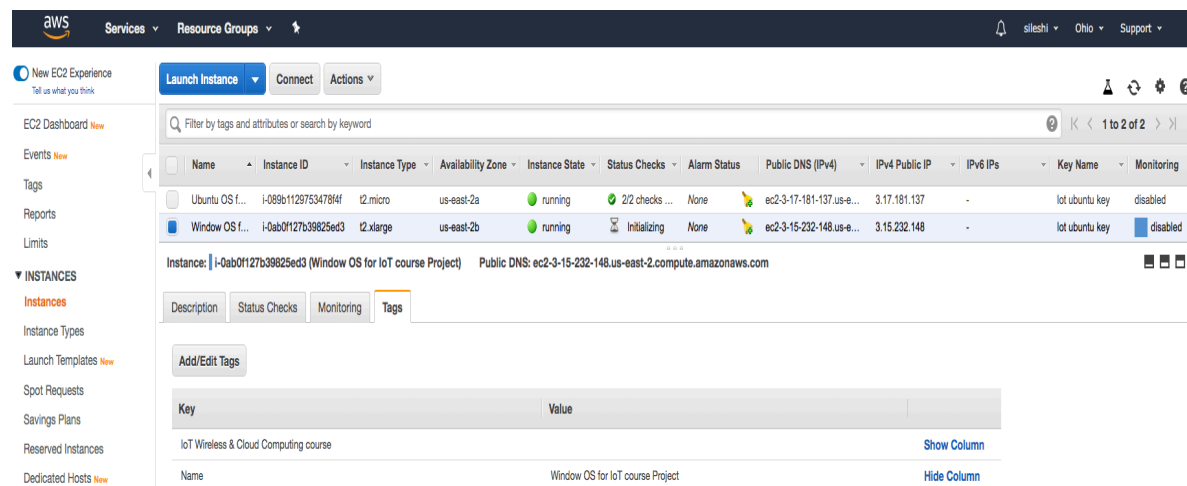
Wi-Fi Signal	
2.4 GHZ	
11 sile_wifi	-58 dBm
20:aa:4b:c5:26:0b	WPA2 (AES)
Cisco-Linksys, LLC	20 MHz, WPS
7 menwlan-02	-72 dBm
00:1e:ab:5c:d4:bb	WPA2 (AES)
TeleWell Oy	20 MHz
7 tony_guests	-76 dBm
00:1e:ab:5c:d4:bc	WPA2 (AES)
TeleWell Oy	20 MHz
11 Scrabble	-80 dBm
00:1d:73:8f:93:36	WPA2 (AES)
BUFFALO.INC	20 MHz
4 ZyxeL_3981	-84 dBm
54:83:3a:ed:39:81	WPA2 (AES)
ZyxeL Communications Corporation	20 MHz, WPS

3. Amazon EC2 projects

For the Amazon EC2 project, I created AWS account and after I get a one-year free trail. I created two EC2 cloud services one Ubuntu platform and the second one is Windows platform with SQL in it.

Generally, what I learn from this project, AWS is a secure cloud services platform, offering compute power, database storage, content delivery and other functionality to help businesses scale and grow. In simple words, AWS allows you to do the following things: -

- Running web and application servers in the cloud to host dynamic websites.
- Securely store all your files on the cloud so you can access them from anywhere.
- Using managed databases like MySQL, PostgreSQL, Oracle or SQL Server to store information.
- Deliver static and dynamic files quickly around the world using a Content Delivery Network (CDN).
- Send bulk email to your customers.



The screenshot displays the AWS Management Console interface. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and user information. The left sidebar shows the 'EC2 Dashboard' and various navigation links. The main content area shows a list of EC2 instances. Two instances are visible: 'Ubuntu OS f...' and 'Window OS f...'. The 'Window OS f...' instance is selected, and its details are shown below the list. The instance is in the 'running' state, with '2/2 checks' passed. The 'Tags' tab is active, showing a single tag: 'Name' with the value 'Window OS for IoT course Project'.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs	Key Name	Monitoring
Ubuntu OS f...	i-089b1129753478f4f	t2.micro	us-east-2a	running	2/2 checks ...	None	ec2-3-17-181-137.us-e...	3.17.181.137	-	iot ubuntu key	disabled
Window OS f...	i-0ab0f127b39825ed3	t2.xlarge	us-east-2b	running	Initializing	None	ec2-3-15-232-148.us-e...	3.15.232.148	-	iot ubuntu key	disabled

Instance: i-0ab0f127b39825ed3 (Window OS for IoT course Project) Public DNS: ec2-3-15-232-148.us-east-2.compute.amazonaws.com

Tags

Key	Value
Name	Window OS for IoT course Project

The above Fig shows the two EC2 cloud service instance that I created for this project