1. Министерство науки и высшего образования Российской Федерации
2. Санкт-Петербургский Политехнический Университет Петра Великого
3. —
4. Институт кибербезопасности и защиты информации

**Высшая школа кибербезопасности и защиты информации**

**ЛАБОРАТОРНАЯ РАБОТА № 1**

1. «**Разработка приложения для сканирования WIFI-сетей для ОС Android**»
2. по дисциплине «Введение в мобильные технологии и средства связи»
3. Выполнил
4. студент гр. 4851003/70801 Гасанов Э.А.

<*подпись*>

1. Преподаватель
2. ассистент Мясников А.В.

<*подпись*>

1. Санкт-Петербург
2. 2020

# Формулировка задания

Разработать приложение для ОС Google Android со следующим функционалом и требованиями:

* Сканирование точек доступа Wi-Fi в радиусе действия и вывод этой информации должен осуществляться в ListView.
* При нажатии на название сети должно открываться новое окно со всей дополнительной информацией о сети: SSID (имя сети), BSSID(MAC-адрес), уровень сигнала, частота, тип используемого шифрования.

# Ход работы

1. Запрос привилегий:

Согласно документации, прежде чем получить доступ к Wi-Fi модулю, необходимо запросить следующие привилегии:

android.permission.ACCESS\_WIFI\_STATE

android.permission.CHANGE\_WIFI\_STATE

Однако для версии Андроид 9 требуется включить дополнительные разрешения:

android.permission.ACCESS\_FINE\_LOCATION

android.permission.ACCESS\_COARSE\_LOCATION

Эти разрешения позволят получить объект WiFiManager как системный

сервис.

# Обработка нажатия кнопки сканирования сети

Создадим слушателя кнопки сканирования сети. Для этого в макете проекта присвоим кнопке идентификатор *scanBtn.* Найдем по заданному id эту кнопку с помощью функции findViewById. Для обработки нажатия клавиши начала сканирования воспользуемся setOnClickListner, где и будет вызываться функция сканирования WiFi.

1. **Сканирование сети**

Имея соответствующие привилегии, на данном этапе можно проверять состояние Wifi-адаптера и включать\отключать его в соответствии с задачами. Получим информацию для инициализации класса WiFi из сетевого контекста потока приложения: wifiManager=(WifiManager)

getApplicationContext().getSystemService(Context.WIFI\_SERVICE);

Нас интересуют именно результаты сканирования, таким образом можно задать фильтр с событием: SCAN\_RESULTS\_AVAILABLE\_ACTION.

После получения такого события можно обработать известные результаты. Сами результаты теперь доступны при помощи метода getScanResults, возвращающий список соответствующего типа.

1. **Связь полученных результатов с ListView**

Находим созданную в макете ListView по id. Свяжем ListView с адапетром. А адаптер свяжем со списком, куда записываются результаты сканирования. Так как это ListView,то выведем только имена сетей и уведомим об обновлении результатов адаптер.

1. **Информационное окно с дополнительной информацией**

Создадим обработчик нажатия пункта из ListView с помощью setOnItemClickListner и вызовем функцию getFullInfo. В ней по выбранной позиции выведем в отдельном окне класса AlertDialog выведем дополнительную информацию о сети.

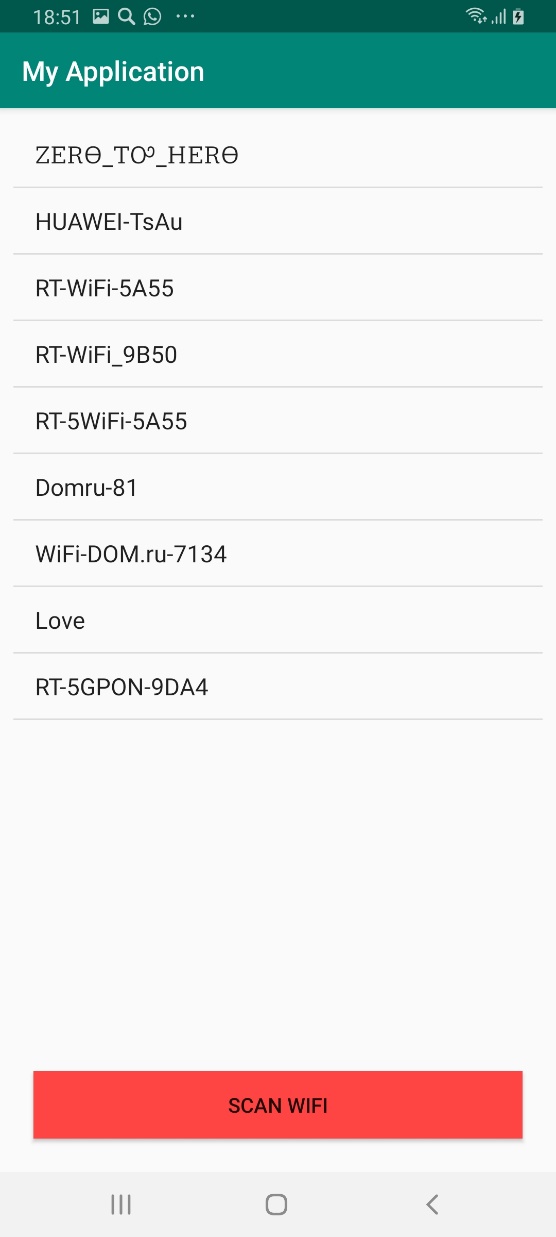


Рисунок 1, список сетей

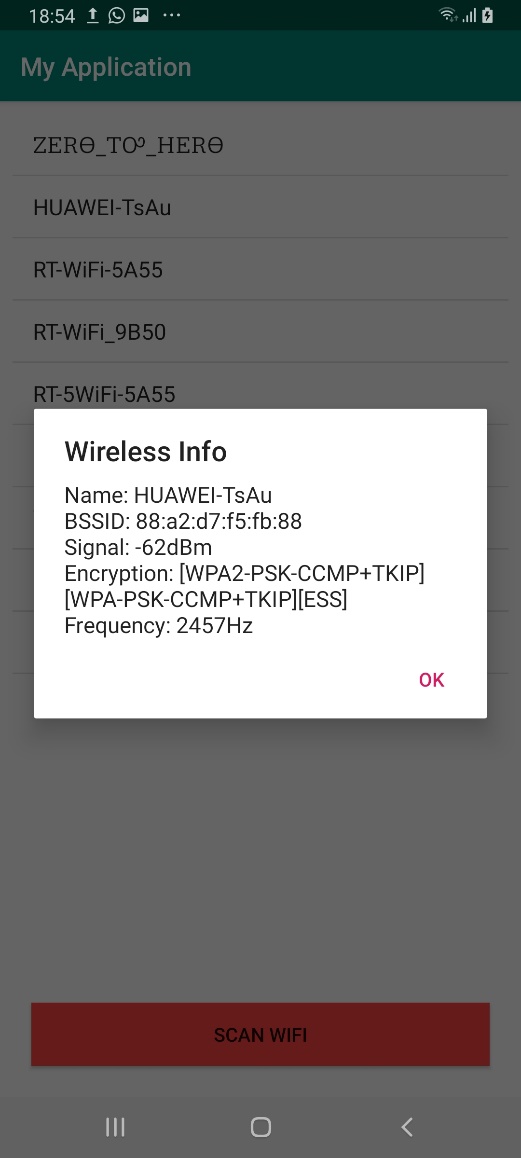


Рисунок 2, дополнительная информация о сети

# Вывод

В ходе работы были реализованы программы для ОС Android с использованием Wifi-модуля устройства.

Приложение позволяет сканировать окружение и обнаруживать доступные беспроводные точки доступа. Благодаря возможностям системы, получение более детализированных результатов также не представляет большой сложности. Полнота информации, полученной в результате сканирования, зависит от установленной версии ОС Android на устройстве, а также используемом для разработки API.

# Приложение

package com.example.myapplication;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.content.IntentFilter;

import android.net.wifi.ScanResult;

import android.net.wifi.WifiManager;

import android.os.Bundle;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.ListView;

import android.widget.Toast;

import android.app.AlertDialog;

import android.app.Dialog;

import androidx.appcompat.app.AppCompatActivity;

import java.util.ArrayList;

import java.util.List;

public class MainActivity extends AppCompatActivity {

private WifiManager wifiManager;

private ListView listView;

private Button buttonScan;

private int size = 0;

private List<ScanResult> results;

private ArrayList<String> arrayList = new ArrayList<>();

private ArrayAdapter adapter;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

buttonScan = findViewById(R.id.scanBtn);

buttonScan.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

scanWifi();

}

});

listView = findViewById(R.id.wifiList);

wifiManager = (WifiManager) getApplicationContext().getSystemService(Context.WIFI\_SERVICE);

if (!wifiManager.isWifiEnabled()) {

Toast.makeText(this, "WiFi is disabled ... We need to enable it", Toast.LENGTH\_LONG).show();

wifiManager.setWifiEnabled(true);

}

adapter = new ArrayAdapter<>(this, android.R.layout.simple\_list\_item\_1, arrayList);

listView.setAdapter(adapter);

listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {

@Override

public void onItemClick(AdapterView<?> adapter, View view, int pos, long id)

{

getFullInfo(pos);

}

});

//scanWifi();

}

private void scanWifi() {

arrayList.clear();

registerReceiver(wifiReceiver, new IntentFilter(WifiManager.SCAN\_RESULTS\_AVAILABLE\_ACTION));

wifiManager.startScan();

Toast.makeText(this, "Scanning WiFi ...", Toast.LENGTH\_SHORT).show();

}

BroadcastReceiver wifiReceiver;

{

wifiReceiver = new BroadcastReceiver() {

@Override

public void onReceive(Context context, Intent intent) {

results = wifiManager.getScanResults();

unregisterReceiver(this);

for (ScanResult scanResult : results) {

//arrayList.add(scanResult.SSID + " + " + scanResult.capabilities);

arrayList.add(scanResult.SSID);

adapter.notifyDataSetChanged();

}

}

};

}

void getFullInfo(int idx)

{

//String info ="kek";

android.net.wifi.ScanResult cur = results.get(idx);

String info = "Name: " + cur.SSID + "\n" +

"BSSID: " + cur.BSSID + "\n" +

"Signal: " + cur.level + "dBm" + "\n" +

"Encryption: " + cur.capabilities+"\n" +

"Frequency: "+ cur.frequency + "Hz";

AlertDialog.Builder builder = new AlertDialog.Builder(this);

builder.setTitle("Wireless Info");

builder.setMessage(info);

builder.setPositiveButton("OK", new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int id) {

// You don't have to do anything here if you just want it dismissed when clicked

}

});

builder.create().show();

}

}