



# iINTERCEPT + SDR for Signal Monitoring on Raspberry Pi

Created by Tim C



<https://learn.adafruit.com/intercept-and-sdr-for-monitoring-signals-on-raspberry-pi>

Last updated on 2026-05-01 03:45:20 PM UTC

# Table of Contents

<b>Overview</b>	<b>3</b>
<ul style="list-style-type: none"><li>• <a href="#">Parts</a></li></ul>	
<b>Install</b>	<b>5</b>
<ul style="list-style-type: none"><li>• <a href="#">Raspberry Pi OS</a></li><li>• <a href="#">Install iINTERCEPT</a></li></ul>	
<b>Use</b>	<b>13</b>
<ul style="list-style-type: none"><li>• <a href="#">Starting iINTERCEPT</a></li><li>• <a href="#">Access the Server</a></li><li>• <a href="#">Usage Documentation</a></li><li>• <a href="#">ADS-B Air Traffic</a></li><li>• <a href="#">Scan for Bluetooth Trackers</a></li><li>• <a href="#">Tune in to AM/FM and HAM Radio Frequencies</a></li></ul>	

---

# Overview

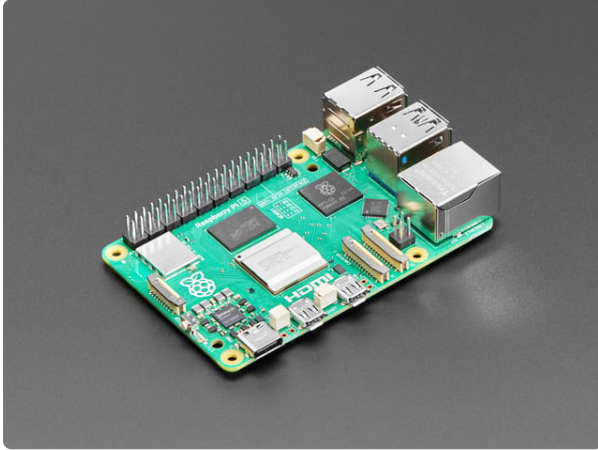


[iINTERCEPT \(https://adafru.it/1aD6\)](https://adafru.it/1aD6) is a free and open-source project that combines several signals intelligence tools together into a single web interface. The UI and dashboards resemble something you might see on a screen in the background of a scene in a spy movie. Even if you're not on a secret mission, this software offers a helpful visual GUI for interacting with a software defined radio device connected via USB.

Using this software on a Raspberry Pi with a cheap SDR, you can scan for BLE trackers, tune in to AM/FM and HAM radio frequencies, automatically decode morse code, receive ADS-B transmissions from airplanes, AIS from ships, and more. The system also supports the [HackRF \(https://adafru.it/1aD7\)](https://adafru.it/1aD7) device which covers the range 1 MHz to 6 GHz, much wider than a cheaper SDR.

iINTERCEPT can run on the Pi 3, 4, or 5. Available features vary based on built-in WIFI/Bluetooth device and the connected SDR.

## Parts



### Raspberry Pi 5 - 4 GB RAM

The Raspberry Pi 5 is the newest Raspberry Pi computer, and the Pi Foundation knows you can always make a good thing better! And what could make the Pi 5 better than the...

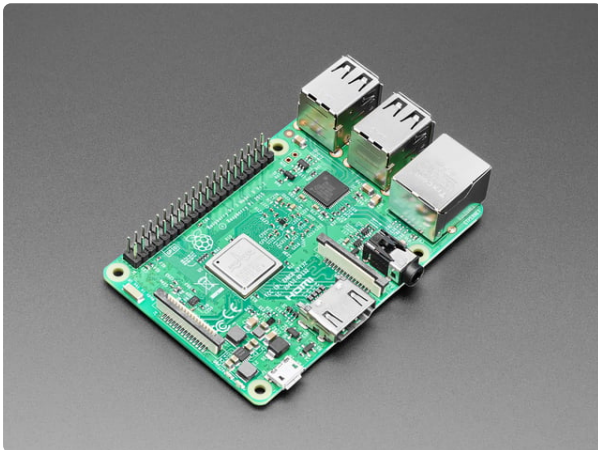
<https://www.adafruit.com/product/5812>



### Raspberry Pi 4 Model B - 4 GB RAM

The Raspberry Pi 4 Model B is the newest Raspberry Pi computer made, and the Pi Foundation knows you can always make a good thing better! And what could make the Pi 4 better...

<https://www.adafruit.com/product/4296>



### Raspberry Pi 3 - Model B - ARMv8 with 1G RAM

Did you really think the Raspberry Pi would stop getting better? At this point, we sound like a broken record, extolling on the new Pi's myriad improvements like we're...

<https://www.adafruit.com/product/3055>



### Software Defined Radio Receiver USB Stick - RTL2832 w/FC0013

If you've ever been curious about software defined radio (SDR), this USB stick is the easiest way possible to have fun with a powerful, configurable receiver. Packed with the...

<https://www.adafruit.com/product/1497>



### Official Raspberry Pi 45W USB-C Power Supply

If you want a general-purpose USB Power Delivery supply, the official Raspberry Pi 45W USB-C power supply makes for a good quality PD supply that provides high current at a large...

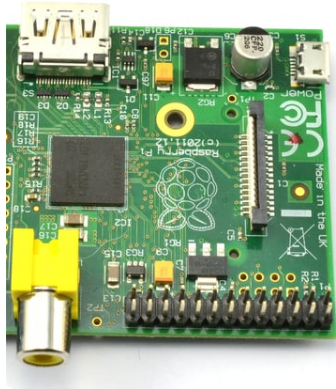
<https://www.adafruit.com/product/6320>

---

## Install

### Raspberry Pi OS

Begin by installing the latest full version of Raspberry Pi OS (64-bit) on your system using the steps outlined on this [Raspberry Pi Imager learn guide page \(https://adafru.it/ZDN\)](https://adafru.it/ZDN). If you intend to have the Raspberry Pi installed somewhere out of the way and wish to connect to it remotely, then be sure to enable SSH and configure your public key in the customization menu of the Raspberry Pi Imager. It's also convenient to configure the WiFi network details inside of the Pi Imager app so that the Pi will automatically connect to your WiFi when it boots up.



Adafruit's Raspberry Pi Lesson 1.  
Preparing an SD Card for your Raspberry Pi

By Simon Monk

▣ [Raspberry Pi Imager](#)

<https://learn.adafruit.com/adafruit-raspberry-pi-lesson-1-preparing-and-sd-card-for-your-raspberry-pi/raspberry-pi-imager>

After you boot up the Pi for the first time on a fresh Raspberry Pi OS image, run these commands to update the software preloaded in the system.

```
sudo apt update  
sudo apt upgrade
```

If these commands fail with network errors, use the WiFi settings in the OS to connect to your network and then try again.

## Install iINTERCEPT

To install the iINTERCEPT software, you need to clone the repo and run the **setup.sh** script inside of.



The iINTERCEPT install script can take a while to complete. It compiles several things from source which can be a slow process, especially on older Pi models. The full install will take 10-15 minutes on a Pi 5 or up to 30+ minutes on older models. During the process, there are several prompts that must be answered to continue.

```
git clone https://github.com/smittix/intercept.git  
cd intercept  
./setup.sh          # Interactive menu (first run launches setup wizard)
```

The setup wizard will begin and walk you through several steps to install and configure various components.

```
1: timc@raspi5-intercept:~/intercept ~
INTERCEPT

Welcome to INTERCEPT Setup!

[*] Detected OS: debian

This wizard will install SDR tools and set up the Python environment.
Choose which tool profiles to install below.

Continue with setup? [Y/n]: y

Install Profiles (space-separated for multiple, e.g. "1 3")
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
 1) Core SIGINT      - rtl_sdr, multimon-ng, rtl_433, dump1090, acarsdec, dumpvdl2, ffmpeg, gpsd
 2) Maritime & Radio - AIS-catcher, direwolf
 3) Weather & Space  - SatDump, radiosonde_auto_rx
 4) RF Security      - aircrack-ng, HackRF, BlueZ, hcxtools, Ubertooth, SoapySDR
 5) Full SIGINT     - All of the above
 6) Custom           - Per-tool checklist
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Select profiles: _
```

If you know that you're only interested in certain types of radio signals and don't care about the others, then select the profile that contains the tools for the signals you're interested in. If you don't know what they are, or just want to tinker with anything available for your hardware enter **5** for the "Full SIGINT" profile.

During the installation, you'll be presented with a several yes/no prompts about installing certain components. Choose yes if you're unsure and want to experiment, or no if you know that you don't need the capability the component provides or want to minimize the time it takes to install.

```
[*] Installation Summary:
    OS: debian
    Profiles: Full SIGINT

Proceed with installation? [Y/n]: y
[*] Updating APT package lists...
[*] Installing Python apt packages...
[1/25] ██████████ 4% - Installing RTL-SDR tools (rtl_fm, rtl_test)
[*] Removing broken rtl-sdr package...
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
[2/25] ██████████ 8% - Installing RTL-SDR Blog V4 drivers

[*] RTL-SDR Blog drivers add V4 (R828D tuner) support and bias-tee improvement
[*] They are backward-compatible with all RTL-SDR devices.
Install RTL-SDR Blog drivers? (recommended for V4 users, safe for all) [Y/n]
```

```
[7/25] ██████████ 28% - Installing ADS-B aircraft decoder
[*] Checking for dump1090 APT packages...
[*] dump1090 not available via APT. Building from source (this may take a while)
[*] Installing build dependencies for dump1090...
[*] Cloning FlightAware dump1090...
[*] Compiling FlightAware dump1090 (using 4 CPU cores)...
[✓] dump1090 installed successfully (FlightAware).
[8/25] ██████████ 32% - Installing ACARS aircraft message decoder
[*] acarsdec not available via APT. Building from source...
[*] Cloning acarsdec...
[*] Compiling acarsdec...
[✓] acarsdec installed successfully.
[9/25] ██████████ 36% - Installing VDL2 aircraft datalink decoder
[*] Building dumpvdl2 from source (with libacars dependency)...
[*] Cloning libacars...
[*] Compiling libacars...
[✓] libacars installed
[*] Cloning dumpvdl2...
[*] Compiling dumpvdl2...
[✓] dumpvdl2 installed successfully.
[10/25] ██████████ 40% - Installing Utility meter decoder (requires rtlamr)
[*] rtlamr is used for utility meter monitoring (electric/gas/water meters)
Do you want to install rtlamr? [y/N]:
```

```
[*] Compiling libacars...
[✓] libacars installed
[*] Cloning dumpvdl2...
[*] Compiling dumpvdl2...
[✓] dumpvdl2 installed successfully.
[10/25] ██████████ 40% - Installing Utility meter decoder (requires Go)
[*] rtlamr is used for utility meter monitoring (electric/gas/water meters)
Do you want to install rtlamr? [y/N]: y
[*] Installing rtlamr from source (requires Go)...
[*] Installing Go via apt...
[*] Building rtlamr...
[✓] rtlamr installed successfully
[11/25] ██████████ 44% - Installing HackRF tools
[12/25] ██████████ 48% - Installing AIS vessel tracker
[*] AIS-catcher not available via APT. Building from source...
[*] Cloning AIS-catcher...
[*] Compiling AIS-catcher...
[✓] AIS-catcher installed successfully.
[13/25] ██████████ 52% - Installing APRS packet radio decoder
[14/25] ██████████ 56% - Installing Weather satellite decoder (NOAA APT & Meteor LRPT)
[*] SatDump is used for weather satellite imagery (NOAA APT & Meteor LRPT)
Do you want to install SatDump? [y/N]: y_
```



```
Or for quick local dev:
sudo -E venv/bin/python intercept.py

Then open http://localhost:5050 in your browser

-----
[+] All REQUIRED tools are installed.
Configure environment settings (.env)? [y/N]: y

Environment Configurator
=====
1) Server
2) SDR Defaults
3) ADS-B
4) Observer Location
5) Weather Satellite
6) Radiosonde
7) Logging & Updates
8) Back
=====
Select category: 1
```

```
=====
Select category: 1

Server Settings
=====
Host to bind
INTERCEPT_HOST [0.0.0.0]:
Port
INTERCEPT_PORT [5050]:
Debug mode (true/false)
INTERCEPT_DEBUG [false]:
Enable HTTPS (true/false)
INTERCEPT_HTTPS [false]:

[+] Settings saved to .env
```

```
Set up PostgreSQL for ADS-B history tracking? [y/N]: y

Database Setup - ADS-B History (PostgreSQL)
=====
[+] PostgreSQL client (psql) not found.
Install PostgreSQL via apt? [y/n]: y
[+] Installing PostgreSQL (this may take a moment)...
[+] PostgreSQL installed
[+] PostgreSQL service running

Database host [localhost]:
Database port [5432]:
Database name [intercept.adsb]:
Database user [intercept]:
Database password [intercept]:
[+] Creating database user and database...
DO
GRANT
[+] Database and user configured
```

The default server configurations are fine if you intend to access the system remotely from a browser on different computer on the same network as the Pi. If you're only going to access it locally in the browser on the Pi itself then you can change the **HOST** IP address to **127.0.0.1** instead of **0.0.0.0**.

See the [Security.md](https://adafru.it/1aD8) (<https://adafru.it/1aD8>) documentation file for more details and information about the implications of these configurations. If you want to keep the system running long term and accessible over the network, it would be wise to configure the firewall and authentication as shown in this document.

Next, the wizard gives you the option to set up PostgreSQL to store historical ADS-B data from aircraft. If you opt not to do this, you can still listen to ADS-B traffic but it will only be stored temporarily and lost when you navigate away from the system.

To enable history, enter **Y** and the desired database configurations. The default values are fine.

Once you've completed all of the prompts you're ready to launch the system using the start script.

---

# Use

## Starting iINTERCEPT

The repo provides a `start.sh` script that can be used to start up the web server and begin listening on the configured port. The instructions say to run it with `sudo` for easiest operation so that it has permission to automatically disable drivers that conflict with its usage of the SDR device. If you have more experience with Linux, you can handle this manually with `modprobe` and then you would not need to use `sudo`.

```
cd intercept # go into the directory if not already there

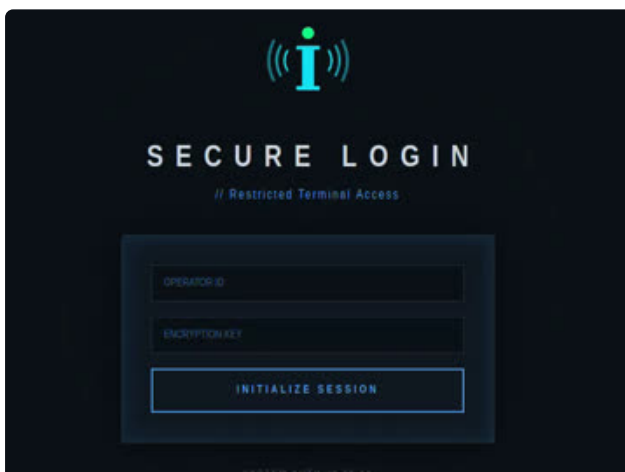
sudo ./start.sh
```

```
tim@raspi3-intercept:~$ cd intercept/
tim@raspi3-intercept:~/intercept $ sudo ./start.sh
[INTERCEPT] Starting production server (gunicorn + gevent)...
[INTERCEPT] Listening on http://192.168.1.160:5050
[2026-04-22 11:44:01 -0500] [41823] [INFO] Starting gunicorn 25.3.0
[2026-04-22 11:44:01 -0500] [41823] [INFO] Listening at: http://0.0.0.0:5050 (41823)
[2026-04-22 11:44:01 -0500] [41823] [INFO] Using worker: gevent
[2026-04-22 11:44:01 -0500] [41838] [INFO] Booting worker with pid: 41838
[2026-04-22 11:44:01 -0500] [41823] [INFO] Control socket listening at /root/.gunicorn/gunicorn.ctl
[2026-04-22 11:44:03 -0500] [41823] [INFO] Handling signal: winch
[2026-04-22 11:44:03 -0500] [41823] [INFO] Handling signal: winch
```

## Access the Server

If you're accessing the system locally on the Pi itself, open the browser and enter `http://localhost:5050` in the address bar.

If you're accessing from another computer on the network, open the browser and use the IP address from the "Listening on \_\_\_" section of the startup output or hostname of the Pi to reach the server, for example `http://192.168.1.160:5050` or `http://raspi5:5050`.

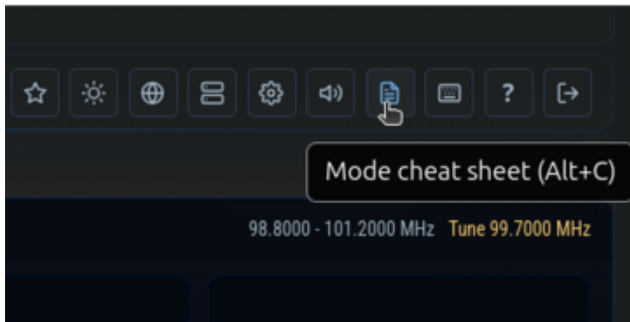


The login screen is purely decorative. The default credentials are `admin / admin` and it does not appear to be possible to change them from the UI.

It has a fun sweeping bar animation to fit the spy movie vibe that is inherent to the whole interface.

# Usage Documentation

The system provides access to lots of different tools and capabilities, the UI can seem complex and a bit overwhelming with panels and controls strewn all over the place.



Along the top right of the page is a row of navigation icons. Amongst them is a **Mode Cheat Sheet** that has a small document with folded corner icon. This will pop up a dialog with high level information about the current mode. It's good to figure out what each mode does.

The repo contains a **docs/** folder with some helpful markdown files.

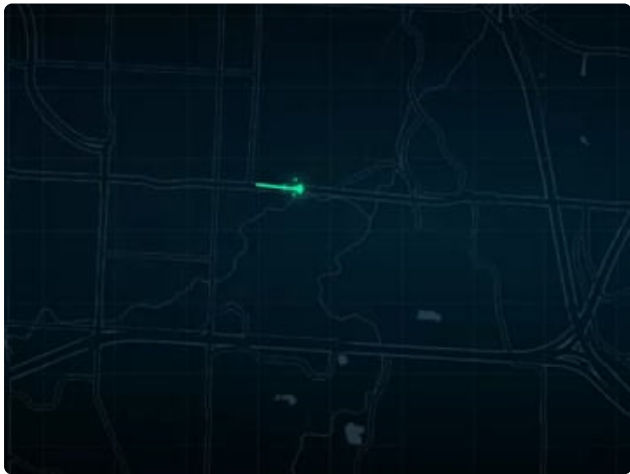
[Features.md \(https://adafru.it/1aD9\)](https://adafru.it/1aD9)

contains a comprehensive list of the systems capabilities

and [Usage.md \(https://adafru.it/1aDa\)](https://adafru.it/1aDa)

contains brief instructions for using each mode.

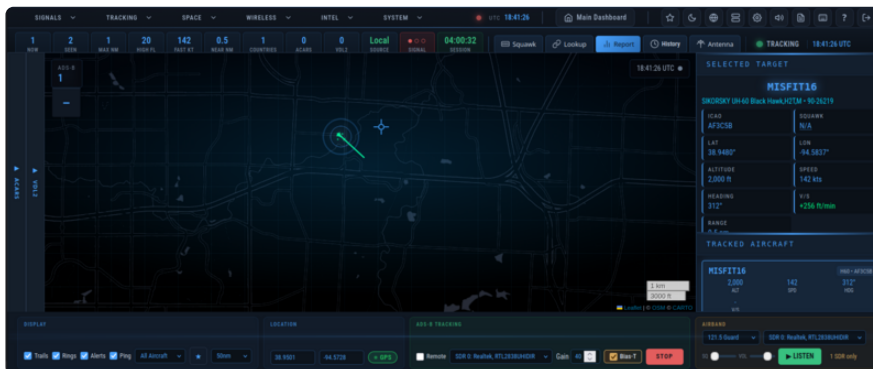
# ADS-B Air Traffic



To access the ADS-B monitoring dashboard, click **Tracking** -> **Aircraft**.

To start monitoring, click the green **Start** button in the panel labeled **ADS-B Tracking** below the bottom right corner of the map.

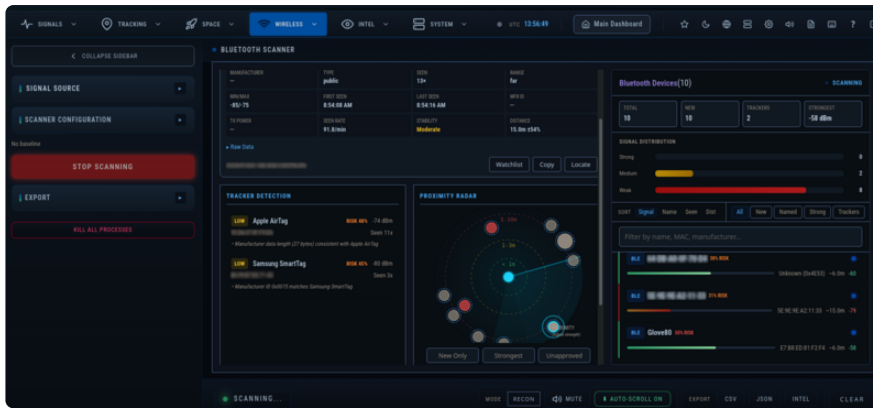
When a transmission is received, the page will play a beep sound and populate data in the transmission into the right side panels. If there is location data, it will plot the aircraft on the map and start showing a historical trail behind it of different way points received so far.



## Scan for Bluetooth Trackers

The Bluetooth dashboard can be accessed by clicking **Wireless** -> **Bluetooth**.

Click the green **Start Scanning** button on the left side of the page to initiate the scan. The system will list all nearby BLE devices that are found with their MAC address or other information that they are broadcasting. Trackers like Apple AirTags and Samsung SmartTags get identified and put into a separate panel to the left of the proximity radar. If you see a tracker and want to narrow down its location further, you can add filters to the monitor and move the Pi around to different locations, taking note of the estimated distance from each location to play a game of digital Marco Polo.



## Tune in to AM/FM and HAM Radio Frequencies

The SDR tuner page is accessed by clicking **Signals** -> **Waterfall**. On this page you can tune to and scan specified frequencies and ranges. The page contains a waterfall visualizer to see a representation of audio recently received.

Type the desired frequency into the box labeled **Center** on the left side of the main panel and choose the appropriate band to the right of that, then press the green **Monitor** button on the left side of the screen to hear audio from the tuned frequency.

In the left hand navigation column there are various panels for bookmarking stations, scanning, handoff, identification and more.

