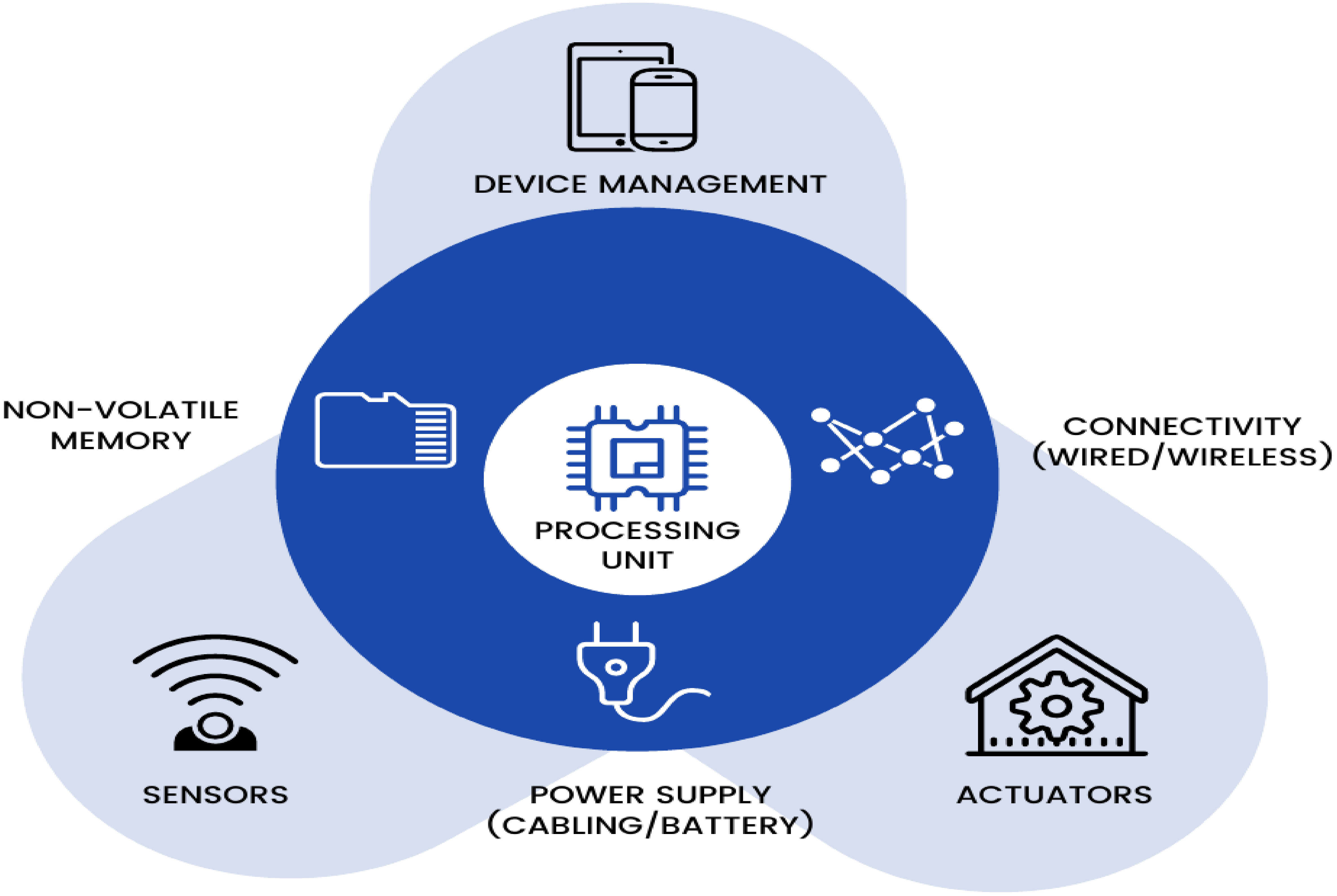
HACKSTUDENT

Internet of Things (IoT) & Security

What is IoT?

- Internet of Things.
- Any device that can send and receive data through the internet.
- Examples include phones, smart devices (fridge, camera, lights, TV), industrial applications like smart city devices (traffic monitoring).


DEVICE MANAGEMENT
NON-VOLATILE MEMORY
PROCESSING UNIT
CONNECTIVITY (WIRED/WIRELESS)
SENSORS
POWER SUPPLY (CABLING/BATTERY)
ACTUATORS

Why is IoT Security so Important?

- IoT devices are typically not very secure.
- They use simple default passwords.
- Infrequent patches - if any.
- Left on the corporate or home network visible to other devices/servers/computers.

Infrequent Patching is especially prevalent on cheaper brands – like a lot of the off brand ones from China

Default Passwords

- A lot of devices come preconfigured with simple default username/passwords.
- It is important to change these default passwords to new secure ones as soon as you get the device.

Remember, a criminal only needs one unsecure entrance to access your home. Your network and data are no different.

How Easy is it to Crack Default Passwords?

- Search up a device make/model followed by "default password“.

- http://open-sez.me - This is a website that keeps a database of default credentials for all sorts of vendors – home and enterprise.

Remember, a criminal only needs one unsecure entrance to access your home. Your network and data are no different.

Network Segmentation

- One of the most important things when it comes to IoT devices is making sure to keep them on a different network from your home or business.

- Companies that have been breached through IoT devices, often had them attached to their regular network which allowed the attacker access to other areas once they got in through the IoT device.

Keeping them separate will mitigate the risk of a more vulnerable device

Real World Examples: Casino Breach

- A casino was breached using an internet connected fish tank.
- The tank was connected to a PC with IoT connected devices like thermometer.
- The thermometer was the point of entry which then allowed them to scan for vulnerabilities across the network resulting in 10GB of data being stolen.

Real World Examples: Mirai

- Botnet was created using IoT devices (Cameras, printers, refrigerators, doorbells, baby monitors, etc.).
- Hundreds of thousands of devices infected.
- DDoS against DYN (DNS service provider).
- DNS translates an IP address to the website name (Netflix, Twitter, AWS, Etsy, Paypal, etc.).

Real World Examples: Mirai Explained

- Found devices by scanning the internet for devices who have telnet port open, it then runs.
- Ran those devices against password "dictionaries" of commonly used and/or default passwords to gain access.
- Once elevated permissions were gained on these devices, they were connected to a C2 server.

C2 stands for Command& Control.

A C2 server commands a Botnet.

Kali Linux

- Kali Linux is a popular open-source operating system specifically designed for penetration testing and ethical hacking.
- It provides a wide range of tools and resources for assessing the security of computer systems, networks, and IoT devices.
- Kali Linux is often used by security professionals and researchers to identify and address vulnerabilities in IoT deployments.

Why
Hackers Use
Kali Linux

Why Kali Linux

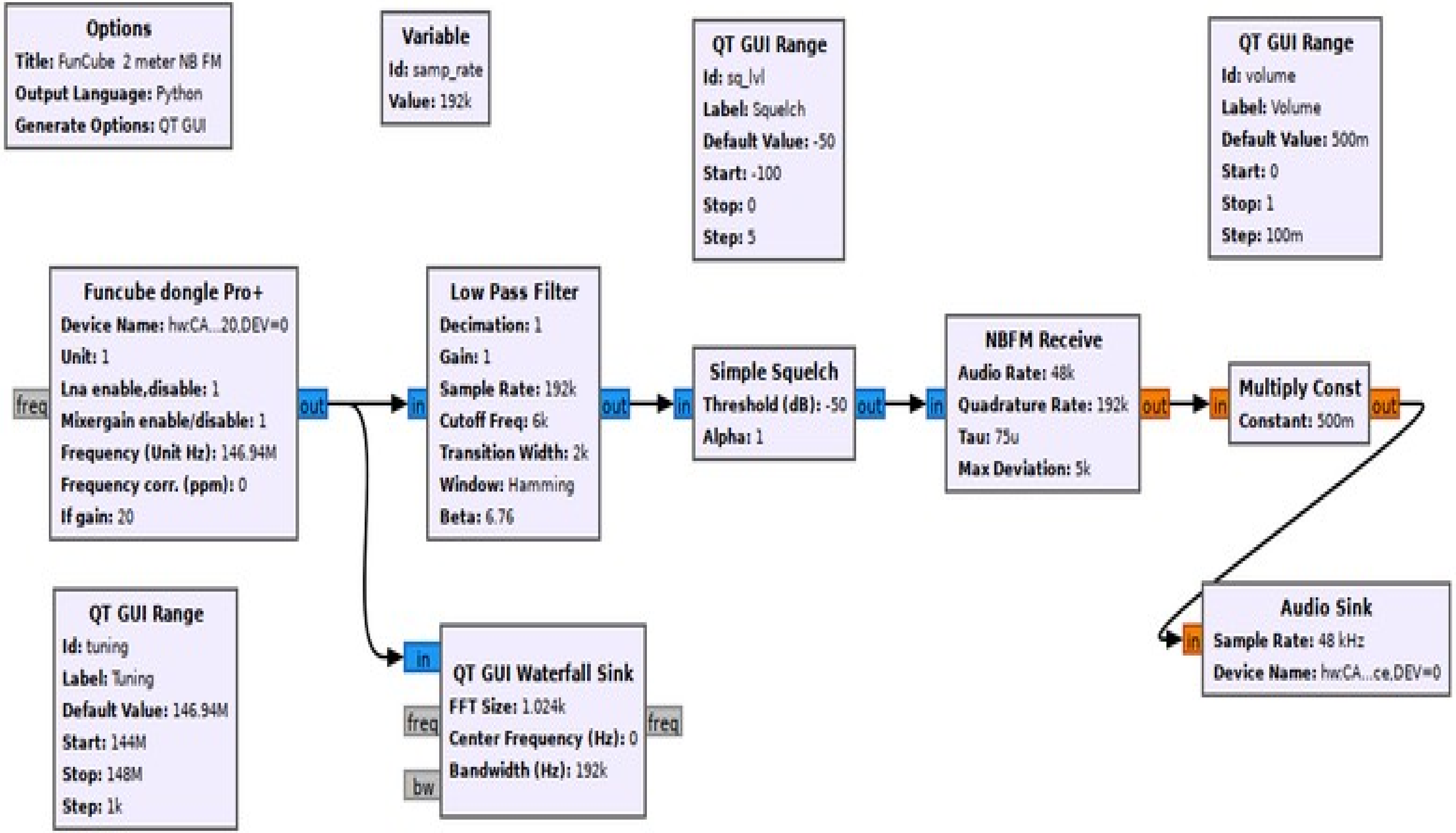
- Protects your Privacy
- Legal Globally
- Works Well at Minimum System Requirements
- Feature Rich

SDR (Software-Defined Radio)

- Software-Defined Radio is a technology that allows flexible and programmable radio communication systems.
- In the context of IoT security, SDR can be utilized to analyze and monitor wireless communication channels, identify vulnerabilities, and assess the security of IoT devices and networks.

GNU Radio

- GNU Radio is an open-source software development toolkit that provides a platform for designing and implementing software-defined radios (SDRs) and signal processing systems. It offers a range of signal processing blocks and tools that can be used to build custom radio systems and perform various digital signal processing (DSP) tasks.



Options
Title: FunCube 2 meter NB FM
Output Language: Python
Generate Options: QT GUI

Variable
Id: samp_rate
Value: 192k

QT GUI Range
Id: sq_lvl
Label: Squelch
Default Value: -50
Start: -100
Stop: 0
Step: 5

QT GUI Range
Id: volume
Label: Volume
Default Value: 500m
Start: 0
Stop: 1
Step: 100m

Funcube dongle Pro+
Device Name: hw:CA...20,DEV=0
Unit: 1
Lna enable,disable: 1
Mixergain enable/disable: 1
Frequency (Unit Hz): 146.94M
Frequency corr. (ppm): 0
If gain: 20

Low Pass Filter
Decimation: 1
Gain: 1
Sample Rate: 192k
Cutoff Freq: 6k
Transition Width: 2k
Window: Hamming
Beta: 6.76

Simple Squelch
Threshold (dB): -50
Alpha: 1

NBFM Receive
Audio Rate: 48k
Quadrature Rate: 192k
Tau: 75u
Max Deviation: 5k

Multiply Const
Constant: 500m

Audio Sink
Sample Rate: 48 kHz
Device Name: hw:CA...ce,DEV=0

QT GUI Range
Id: tuning
Label: Tuning
Default Value: 146.94M
Start: 144M
Stop: 148M
Step: 1k

QT GUI Waterfall Sink
FFT Size: 1.024k
Center Frequency (Hz): 0
Bandwidth (Hz): 192k

GNU Radio: Features

- **Protocol Implementation:** GNU Radio allows the implementation of various communication protocols used in IoT, such as Zigbee, Bluetooth Low Energy (BLE), LoRa, and more. With the flexibility of GNU Radio, developers can create custom signal processing flowgraphs to decode and encode these protocols, enabling communication with IoT devices.
- **Spectrum Analysis:** GNU Radio provides a rich set of tools for spectrum analysis, allowing researchers and developers to analyze and monitor the radio frequency (RF) spectrum used by IoT devices. This capability can be used to identify and troubleshoot interference issues, perform signal characterization, and assess the performance of IoT networks.
- **IoT Security Research:** GNU Radio's flexibility and extensibility make it an excellent tool for IoT security research. Security professionals can utilize GNU Radio to analyze the security of wireless communication protocols used in IoT, identify vulnerabilities, and develop countermeasures. This can include activities such as signal eavesdropping, packet injection, and protocol analysis.
- **Rapid Prototyping:** GNU Radio's graphical interface and code generation capabilities facilitate rapid prototyping of IoT systems. It allows developers to quickly build and test signal processing chains for IoT applications, enabling faster development cycles and iterative design processes.
- **Integration with SDR Hardware:** GNU Radio supports integration with a wide range of SDR hardware devices. This allows IoT developers to connect SDR devices to their GNU Radio flowgraphs and interact directly with IoT devices operating in various frequency bands, facilitating experimentation, testing, and analysis.