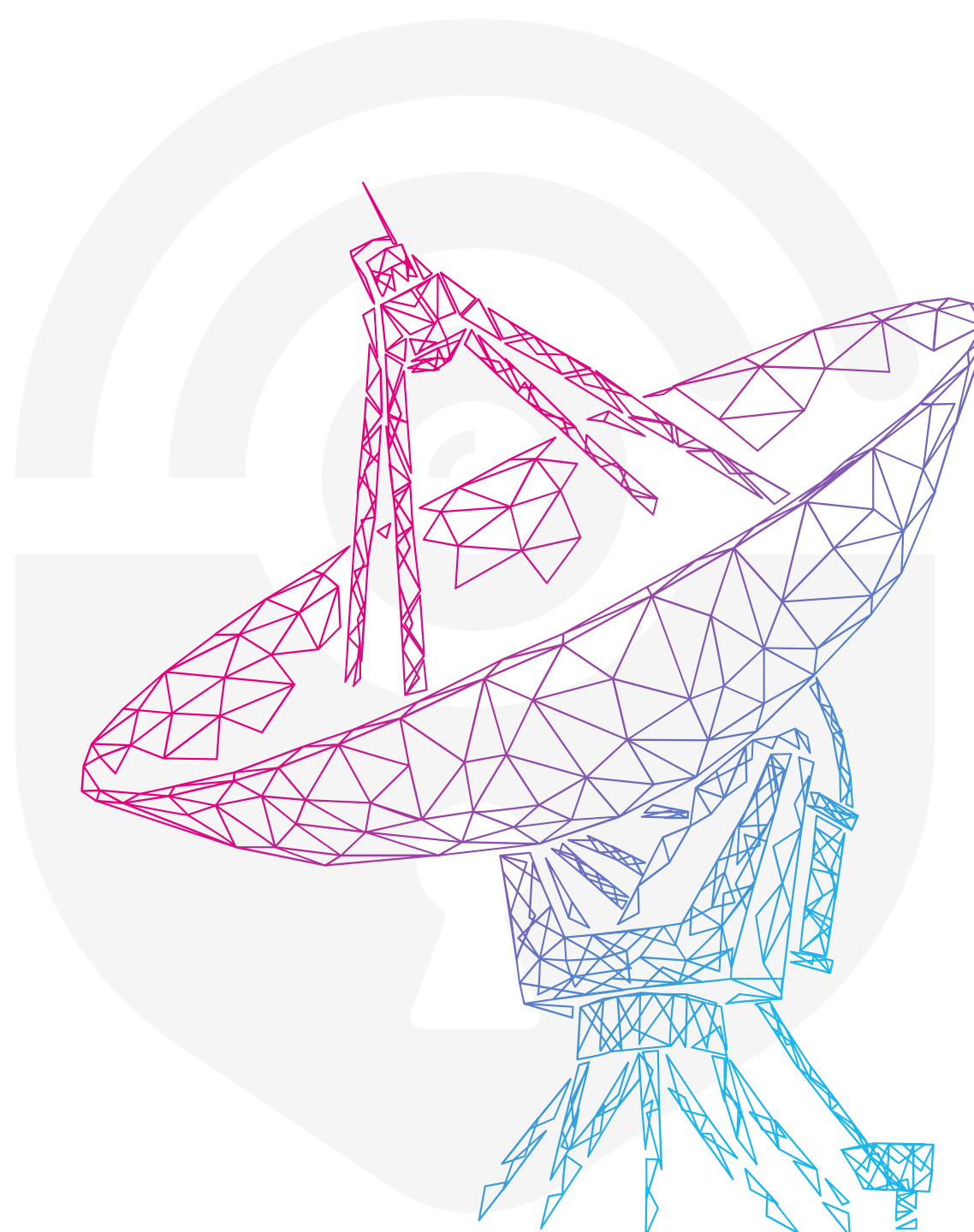
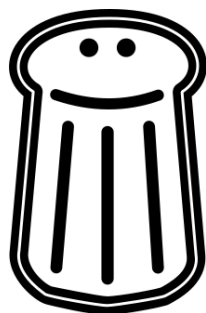




RF Swift: a swifty toolbox for all wireless assessments

By Sébastien Dudek



Founder of Penthertz

- Sébastien Dudek (@FIUxluS)
- CEO of Penthertz
 - Founded during COVID in 2020
 - Specialized in Wireless communications security
- > 10 years of experience in Software & Hardware security
 - Security researcher
 - Pentester & Red Team
 - Vulnerability researcher

Perfect mix to make
Penthertz!



Main activities



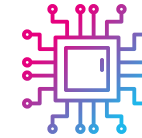
Security assessments

- Wireless communications (RFID, Wi-Fi, Mobile communications, Bluetooth, etc.)
- Embedded devices
- Backend servers
- Red Team



Trainings

- Software-Defined Radio Hacking
- Wi-Fi Red teaming
- RFID Hacking
- Mobile attacks (2G/3G/4G/5G), and more...



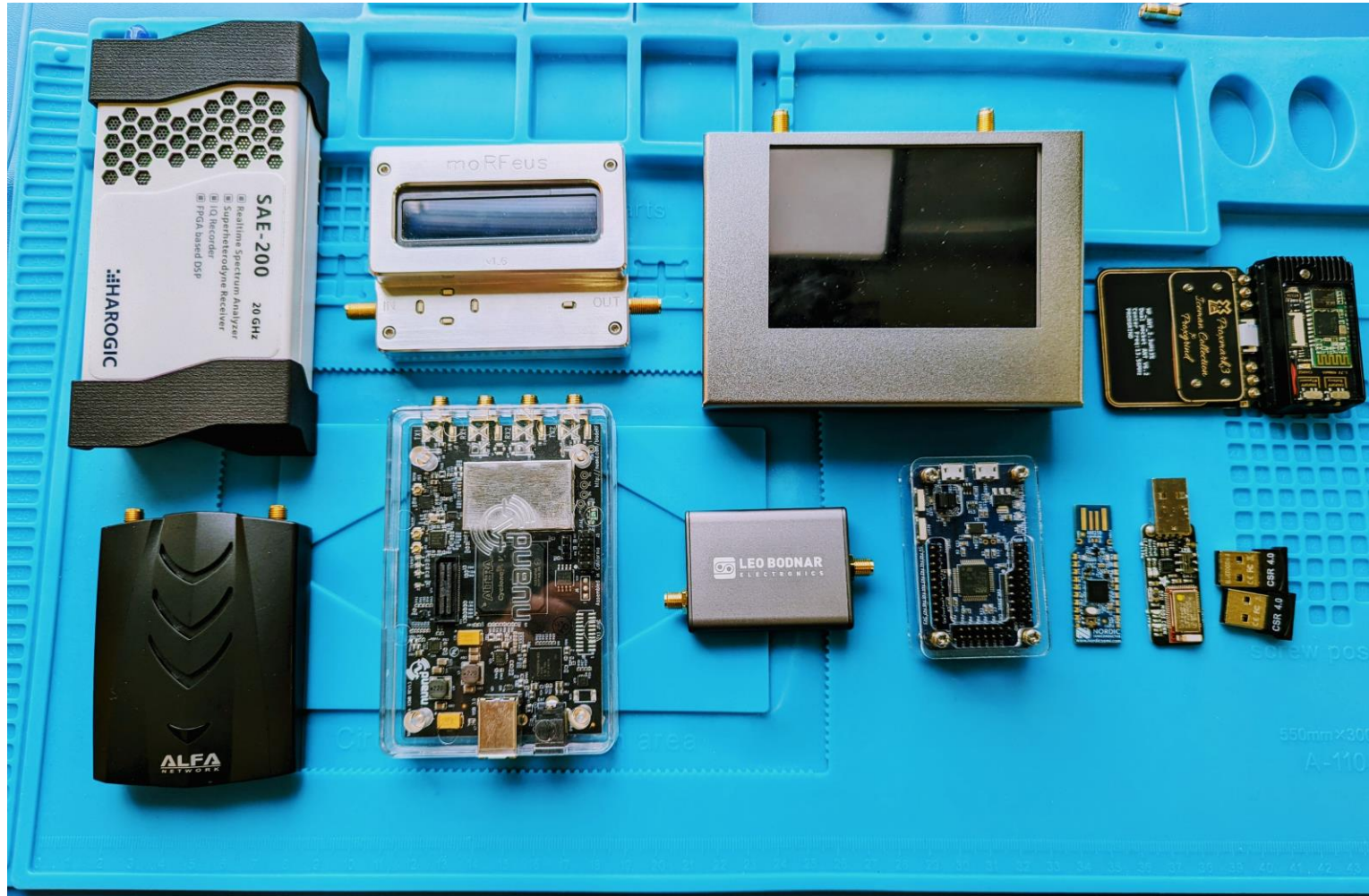
Hardware security

- Firmware extraction
- Chip off
- Secrets extraction
- Library's analysis
- Vulnerability hunting

RF Pentester 010: Having a good setup



A minimum setup for assessments



Software setup

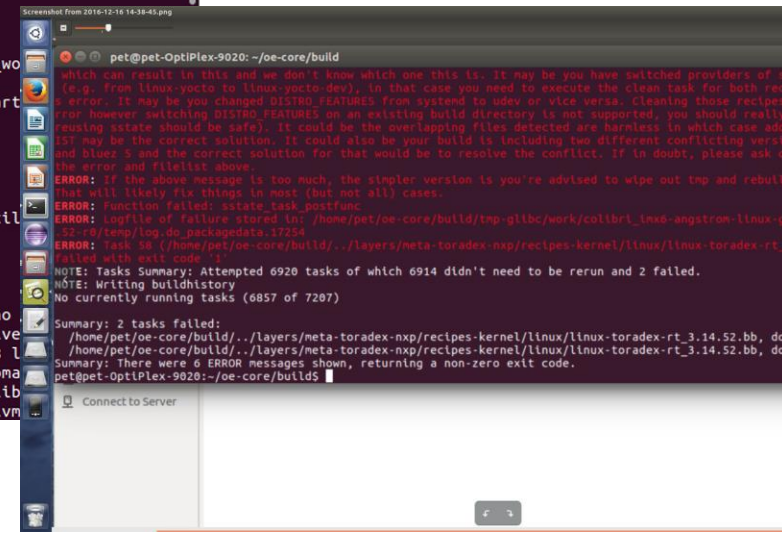
- We need all required pentests tools for different context:
 - Wi-Fi
 - RFID
 - Bluetooth Classic & LE 4/5
 - Telecom
 - And even exotic communications
- In addition: report generator, common network tools, web tools, etc.
- **But: takes at least 1-5 days to setup properly (depending on number of tools)**

Compile your tools

- Need to deal with:
 - Compilation issues
 - Dependencies
 - Collisions/conflicts
- A good setup can take a day to a week depending on needed tools
- Time is running
- **Not good when rushing on an assessment...**

```
CC [M] drivers/net/ethernet/mellanox/mlx5/core/dev.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/wq.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/lib/gid.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/diag/fs_tracepoint.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/diag/fw_tracer.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_main.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_common.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_fs.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_ethtool.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_tx.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_rx.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_dtm.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_txrx.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_xdp.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_stats.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_selftest.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en/port.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_arfs.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_fs_ethtool.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_dcbnl.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en/port_buffer.o
CC [M] drivers/net/ethernet/mellanox/mlx5/core/en_rep.o
gcc: fatal error: Killed signal terminated program cc1
compilation terminated.
make[5]: *** [scripts/Makefile.build:304: drivers/net/ethernet/mellanox/mlx5/core/en_rep.o] Error 1
make[5]: *** Deleting file 'drivers/net/ethernet/mellanox/mlx5/core/en_rep.o'
make[4]: *** [scripts/Makefile.build:544: drivers/net/ethernet/mellanox/mlx5/core] Error 2
make[2]: *** [scripts/Makefile.build:544: drivers/net/ethernet/mellanox] Error 2
make[1]: *** [scripts/Makefile.build:544: drivers/net/ethernet] Error 2
make: *** [scripts/Makefile.build:544: drivers/net] Error 2
```

```
can@can-VirtualBox: ~/reversing
can@can-VirtualBox:~$ pwd
/home/can
can@can-VirtualBox:~$ mkdir reversing
can@can-VirtualBox:~$ cd reversing/
can@can-VirtualBox:~/reversing$ nano hello_world.c
can@can-VirtualBox:~/reversing$ gcc -m32 hello_world.c hello_world.o
/usr/include/stdio.h:27:10: fatal error: bits/libc-header-start.h: No such file or directory
 27 | #include <bits/libc-header-start.h>
    |          ^~~~~~~~~~~~~~~~~~~~~~
compilation terminated.
can@can-VirtualBox:~/reversing$ gcc hello_world.c
can@can-VirtualBox:~/reversing$ sudo apt-get install gcc-multilib
[sudo] password for can:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi i965-va-driver intel-media-va-driver libaac0 libaom3 libass9 libavcodec58 libavutil56 libbdplus0 libblas3 libbluray2 libbs2b0 libchroma libcodec2-1.0 libdav1d5 libflashrom1 libflite1 libftdi1-2 libgstreamer-plugins-bad1.0-0 libigmpmm12 liblilv-0-0 liblvm
```



Alternative distributions

- Existing alternative distributions:
 - Kali: packages for Wi-Fi, Bluetooth, RFID, SDR and many other pentest tools
 - Pentoo: Like Kali with extra GNU Radio tools and modules, SDR tools as well (<https://github.com/pentoo/pentoo-overlay/tree/master/net-wireless>)
 - Dragon OS: Really focusing on radio tools and much more complete than other distributions
 - Others



Alternative distributions (2)

- **Pros:**

- Packages as much tools as possible --> reducing installation time
 - Tools not yet package can be installed after
- Less troubleshooting during our setup --> tools are ready to be used
- Perfect for less experienced people

- **Cons:**

- Need to reinstall the computer with the specialized distribution
- Dependencies issues with new installed tools --> breaking the setup

Alternative distributions

- Existing alternative distributions:
 - Kali: packages for pentesting, SDR, SDR, many other pentesting tools
 - Pentoo: Like Kali with extra modules, SDR tools as well (<https://github.com/pentestmonkey/pentoo-overlay/tree/master>)
 - Dragon OS: Really good SDR tools and more complete toolsets
 - Others



Breaking the setup

- **Need to reinstall everything! Sometimes until 5am during a pentest...**



Breaking the setup (2)

- **And doing that all the time, your turn like:**





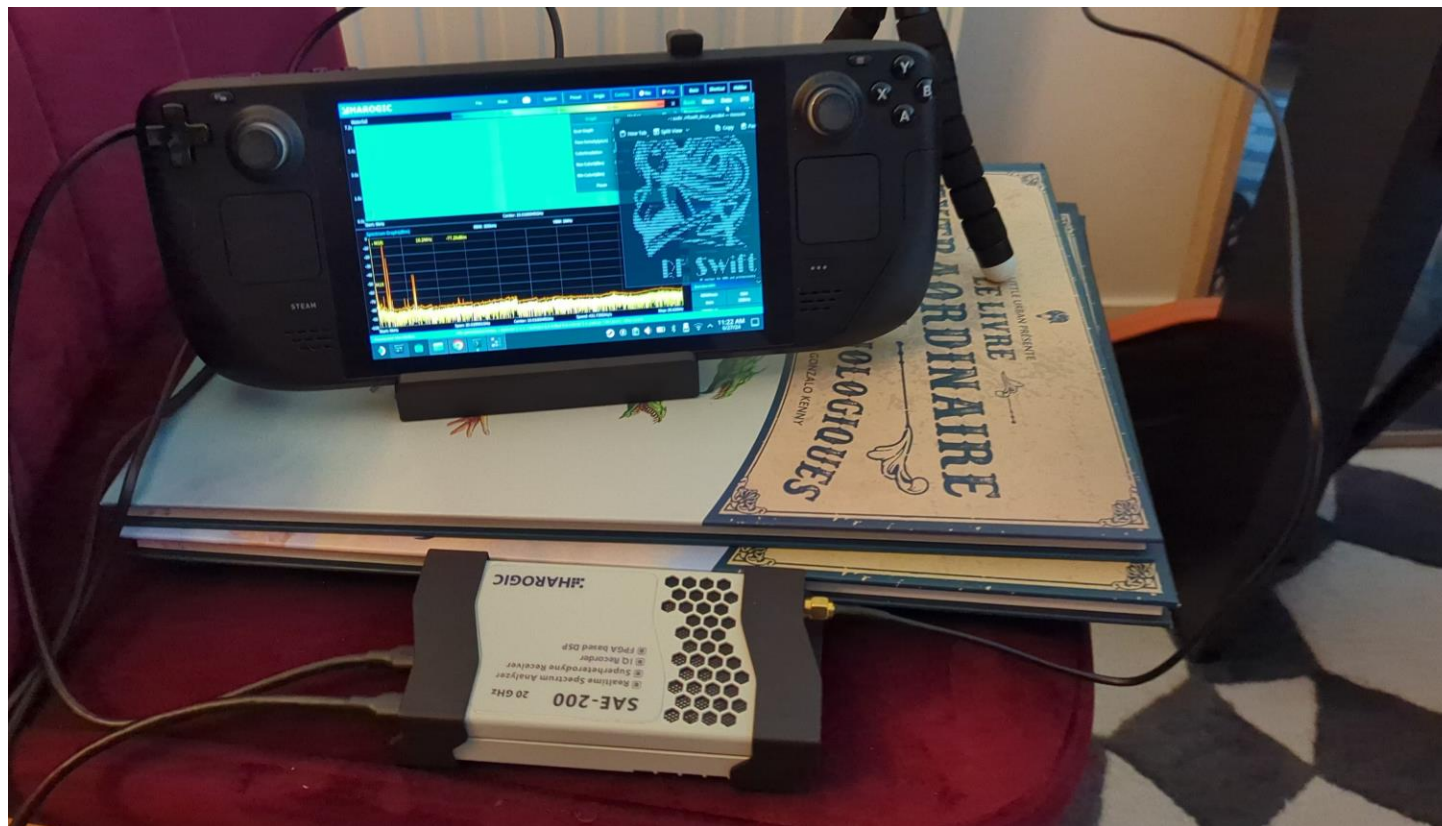
Meet RF Swift!

What is it?

- Tool made in Go --> Instrumenting Docker + host
 - Inspired from Exegol project ;)
- Docker files "recipes"
- Registry with built images
- Scripts for automating installations of various tools
- Supported and tested architectures: x86_64, ARM64, and RISC-V 64
- Supported and tested OSes: Linux and Windows



Assessments on a Steam Deck



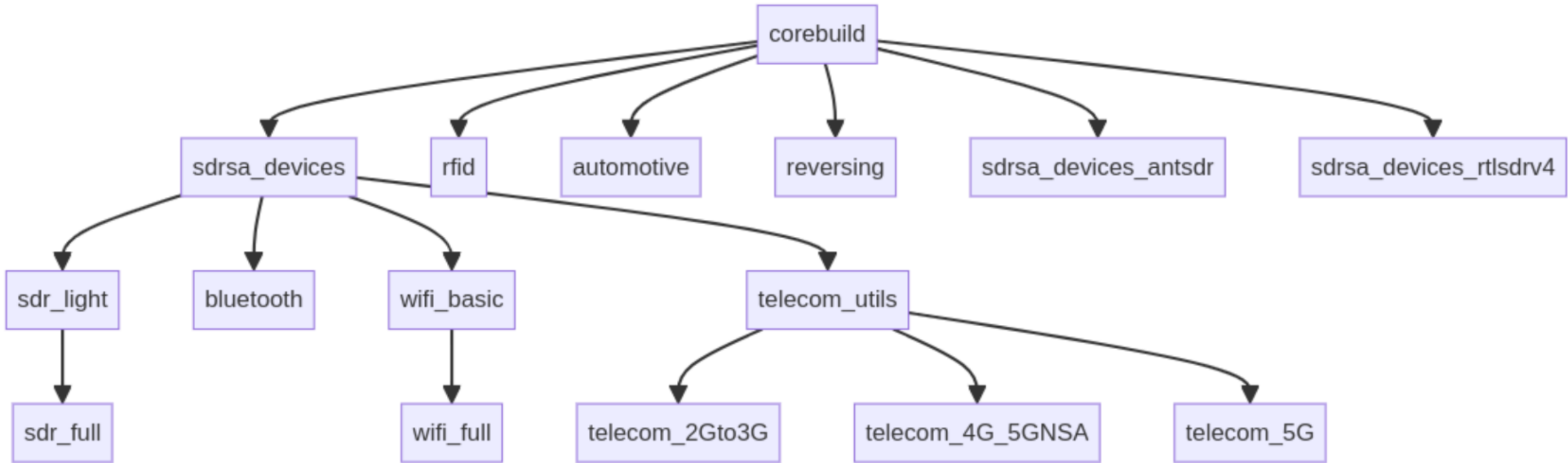
Windows GPRS stations (in few minutes)

The screenshot shows a Windows desktop environment. On the left, a Command Prompt window displays colorful ASCII art and text: "RF toolbox for HAMs and professionals", "You are running version: 0.4.8 (Up to date)", and the file path "C:\Users\fluxius\Desktop\New folder\". On the right, a Docker Desktop window shows the "Images" tab with a search bar and a table of images. Below the images, there are "Walkthroughs" for "How do I run" (6 mins) and "Run Docker H" (5 mins). In the foreground, a yate terminal window displays a disclaimer: "This program comes with ABSOLUTELY NO WARRANTY. Use of this software may be subject to other legal restrictions, including patent licensing and radio spectrum licensing. All users of this software are expected to comply with applicable regulations and laws. See the LEGAL file in the source code for more information." followed by logs for a GPRS station, including release information and initialization steps like "Starting MBTS...", "Yate engine is initialized and starting up on docker-desktop", and "bladeRF detected, attached to serial=bd7fff8efb4de4ba08d94bd5958b06".



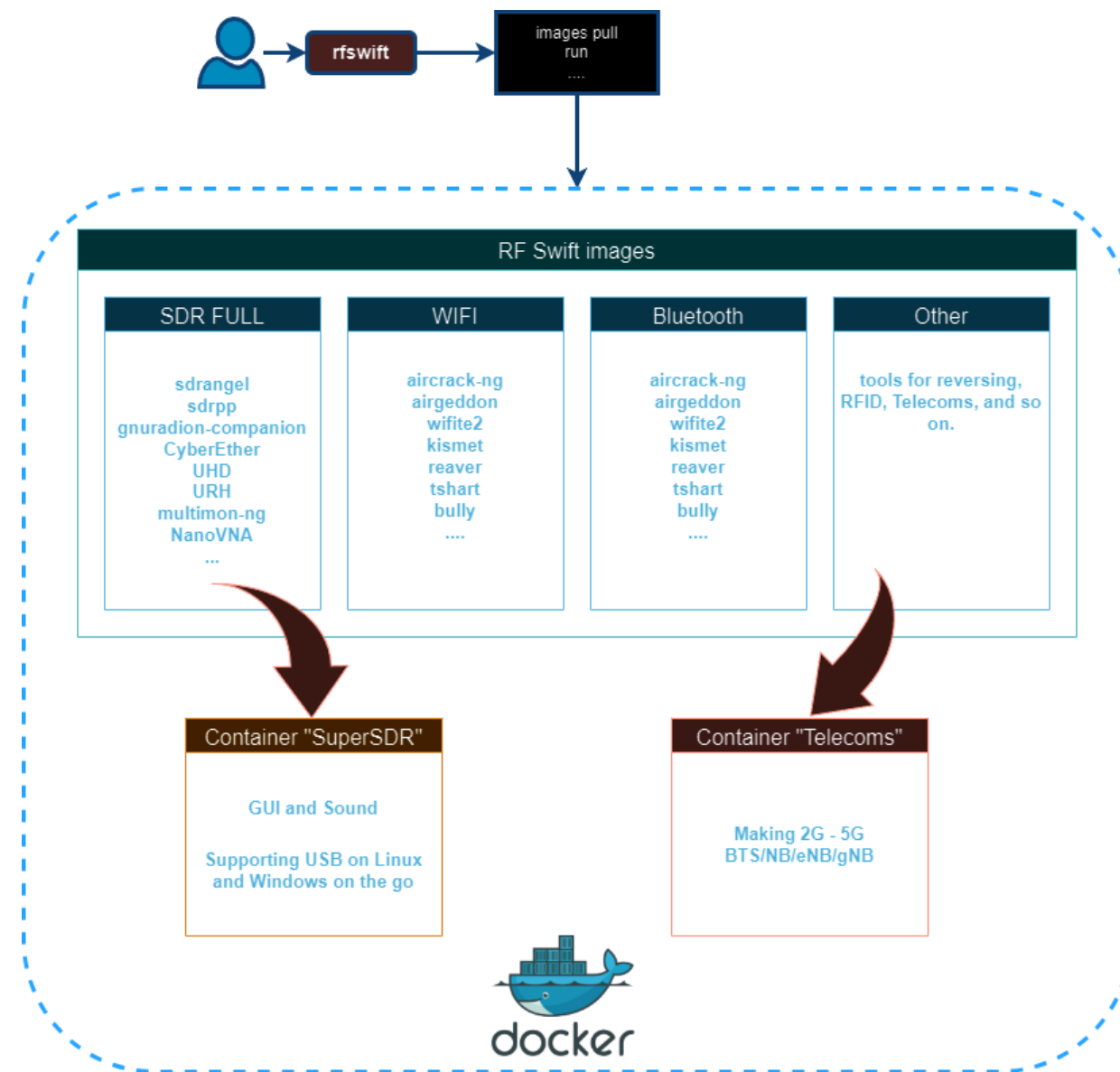
Images' hierarchy

- Following Docker images layers concept: reuse of layers -> speed and space saving



Architecture

- Each created container has tools included in dedicated images
- Each container represent a "mission"
 - Perfect for assessments separation: client1 and client2 are not in the same space
 - Messing with one container -> throw it and run a new container!



The background features a color gradient from deep purple on the left to bright blue on the right. Overlaid on this are two large, semi-transparent circular patterns of concentric lines. The left pattern is purple and the right pattern is blue, both centered on their respective sides of the frame.

Demo time!

Conclusion

To conclude

- You can travel and assess devices safely with RF Swift
- Keep you setup light based on your own "recipes"
- RF Swift is 3 months old --> will grow with more tools
- Need also contributors:
 - Documentation: <https://rfswift.io/>
 - Go binary for instrumentation and user experience
- Our discord: <https://discord.com/invite/NS3HayKrpA>



Thank You

Please contact us:

✉ contact@penthertz.com

☎ +33 1 73 13 82 77

🌐 penthertz.com

Watch us on

