Pierre-Louis Braun

Born on July 13th, 1998

Mulhouse – France

Currently living in Another City

plbraundev@gmail.com

Personal Interests

- Computers
- Science
- Physics
- AI
- Hacking / Cybersecurity

Education

2017 - 2020. **UHA 4.0**

Computer Science University

Mulhouse – France

Lycée Blaise Pascal Colmar

Highschool

Colmar – France

Collège Jaques Prévert

Middle school

France

Current situation:

Sept. 2021 – Now. **Self employed**

Working at my own company on various projects.

Bio:

I'm a computer hobbyist since the age of 9, i've been learning computer science, programming languages and various technologies since.
i don't have many diplomas but i can show my skills.

Skills:

Languages:

C | C++20 | ASM x86 & ARM | Python | Rust
Bash | ZSH | Vim, Vimscript | Make | Cmake | Awk
HTML5, css, Javascript | NodeJs | PHP | Go
Ruby | Java | Kotlin | Dart
Haskell | Lisp | Lua | Ada, Ada Sparks | Zig | Wasm
Julia | R | Cobol | Fortran
Markdown | LaTeX| XML | YAML | JSON

Tools / Software:

GNU coreutils, gdb, radare2, ghidra, ida pro, frida, sed, nmap
metasploit framework, bettercap, netcat, netstat, ip, ss
{s,m,l}trace, objdump, xxd, ld, ldd, wireshark, tshark, tcpdump [..]

Sysadmin skills:

Linux, Unix | KVM | Qemu | Libvirt | Docker | LXC, LXD
Nix | Kubernetes | ZFS | Btrfs
Regex, globs & wildcards
Systemd | systemd-nspawn | machinectl

Technology:

Git | UML | Graphviz | Ajax | Arduino | Jira | Confluence
SQL: mysql, postgres | NoSQL: mongodb, redis, scylladb
NewSQL: cockroachDB | Nginx

Libraries:

SDL2 , SFML, OpenGL, Vulkan
Ncurses, QT, GTK, Tk, XCB
Opencl, Sycl, Hip, OpenACC/OpenMP, Cuda
socket, libssh, libsodium, STD & STL algorithm[...]

Misc:

Cybersecurity, Reverse engineering, Binary exploitation
Forensic analysis, Cryptanalysis, Data Oriented Programming
AI, Machine learning and expert systems.
Android JDK, Android NDK | Computer Architecture
Arch Linux, Gentoo, LFS(linux from scratch), NixOS, debian*

Achievements

Currently in the top 50 of the DGSE hacking 404CTF **(Now)**
One of the winners of the DGSE Richelieu hacking CTF **(2019)**

Languages

French
English

Hobbies

Cooking | Programming | Hacking | Painting
Listening to music | Hiking | Electronic
Learning | Reading | Sport | Skateboarding

Last updated: August, 2022.

This resume was made using LaTeX

Projects:

Virtablet app:

Virtablet was an app that allowed you to use any android tablet with the required hardware like a cintiq graphic tablet (including video transfer and multitouch) on windows and Linux computers (mac and ipad support would be added later);

To do so, a kotlin android app was made, it registered the events (position, pressure, tilt, orientation, multitouch) then sent them over the network using a custom protocol over tcp and udp. on the computer side i had to write a server to receive and inject the event, the server was written in C++ and the networking was done with tcp and udp sockets, i used #ifdefs for linux and windows compatibility as i made my own networking library. then once the packets were received and the events parsed from my custom format, i had to inject them; since windows doesn't have any syscalls for input injection, i had to develop a driver using **KMDF** so that i'd be able to create a "virtual graphic tablet" to which i'd send pen and multitouch events.

on Linux on the other hand, i didn't need to write a driver and could just use **libinput**, basically allowing you to create virtual devices with ioctls where you define what kind of device it is and its varius parameters, and then sending the events to that device.

Network discovery was made using udp broadcast.

Only the video transfer part was left to do, but unfortunately someone made the same software and released it only a few weeks before my planned release date, i contacted the developper behind it and like me he was an independant, however he started working on the project two years earlier.
as the app lost most commercial viability, i never finished the project and moved on to something else.

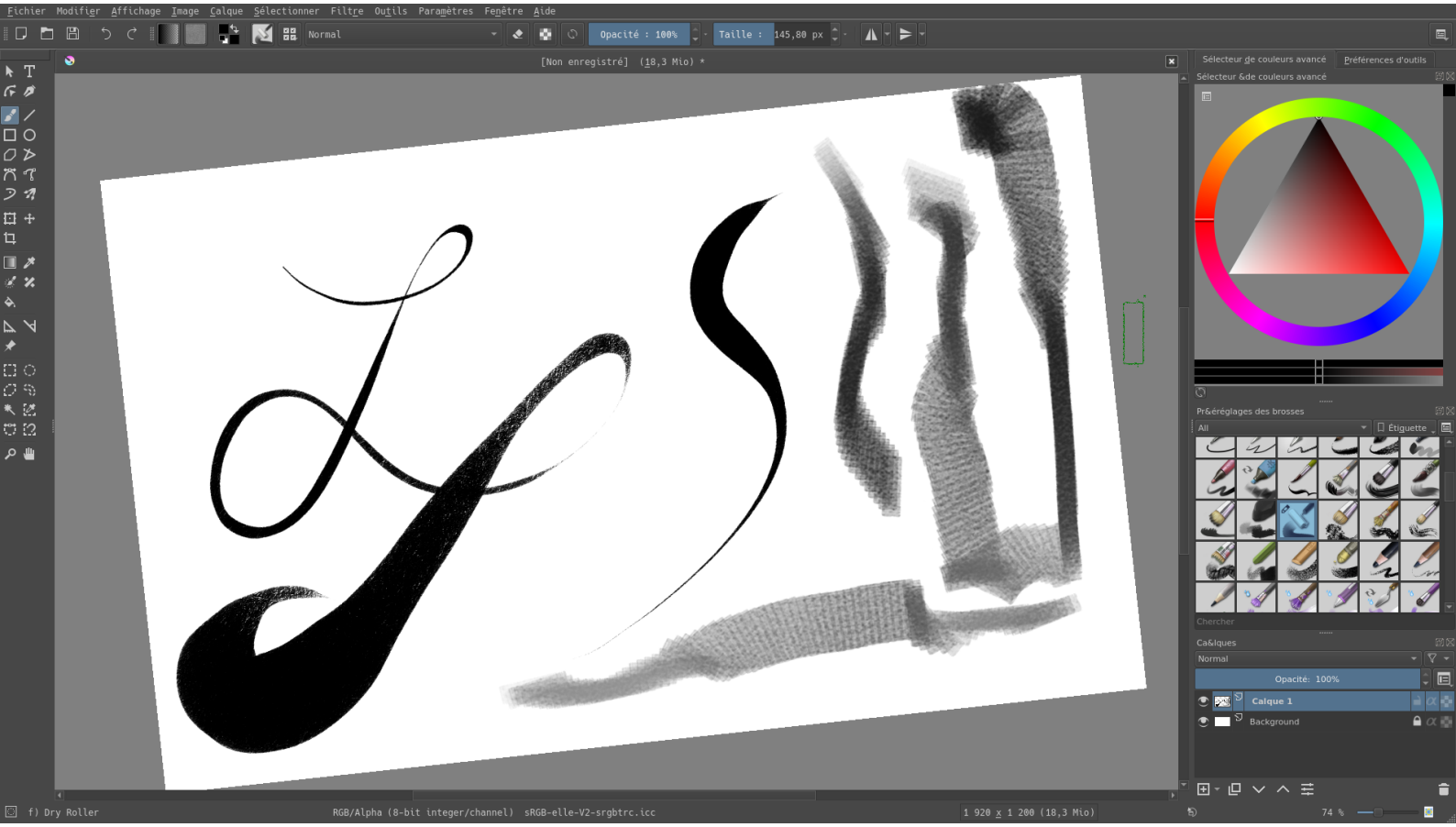

Figure 1: Strokes made in the krita software running on the computer using a galaxy tab s4 running my app and sending the events to the server running on the computer which in turn, injects said events
as you can see, pressure sensitivity and tilt is working. the canvas was also rotated using multitouch which was also sent from the tablet and injected by the computer server.

Dating app:

I made a dating app for a niche i won't specify, the backend was made in **Rust** and the database i used was scylladb and redis. the front was made in react, however i never finished that project as my personal financial situation no longer allowed me to finish it and my new priority was to find a reliable source of income.

which bring us to today.

i had many more side projects including writing a spiking neural network with dynamic topology (neurons can be created / removed whilst the network is running) library from scratch but i only put here what i think are the two most relevant.

PS: i hope you'll excuse my grammar as it is definitely not the thing i'm the best at.