Thomas **Hybel**

Software Developer · Security Researcher

(+45) 30 28 07 88 | tehybel@gmail.com | thomashybel.com | tehybel | tehybel

Education

Aarhus University — *Aarhus, Denmark*

M.S. in Computer Science — *Aug. 2016 - Jun. 2018*

- Focused on programming languages, program analysis, and cryptography
- Minored in mathematics

Skills

Fluent programming	C, Python
Capable programming	C++, Scheme, x86 & ARM assembly, SML, PHP, JavaScript, Solidity, Java, Bash, SQL
Software	GNU/Linux, Vim
Natural languages	English, Danish, Spanish (intermediate)

Experience

Aarhus University — *Aarhus, Denmark*

Student Programmer — *May 2014 - Sep. 2014*

- Implemented standard library for the relational programming language RASMUS
- Ensured that standard library functions scaled to large datasets by picking appropriate algorithms

TravelMarket — *Vejle, Denmark*

Web Developer — *May 2013 - Jan. 2014*

- Developed large parts of the front end for BedreBilist.dk, a website for taking theoretical driving lessons
- Performed security analysis of main company website, fixing multiple critical vulnerabilities

Extracurricular Activity

Team Tasteless (international CTF team) — *Various countries, Europe*

Core Member — *2011 - PRESENT*

- Solved challenging problems in hundreds of Capture-The-Flag (CTF) security competitions
- Team ranked #8 globally in 2016
- Qualified for the DEF CON finals in 2017
- Performed low-level binary exploitation and reverse engineering for the team

Projects

Compiler implementation — *Aarhus, Denmark*

Bachelor project — *Jan. 2016 - Jun. 2016*

- Implemented a compiler from Tiger to x86 in Standard ML
- Wrote each stage from scratch, including lexing, parsing, semantic analysis, IR conversion, and code generation

Operating System implementation — *Aarhus, Denmark*

Student project — *May 2017 - Jan. 2018*

- Implemented an x86 operating system from scratch in C
- Wrote code for booting, memory management, process management, concurrency, file system
- Implemented a graphics subsystem and adapted code to run on real hardware

Decompiler implementation — *Aarhus, Denmark*

Master's thesis project — *Dec. 2017 - Jun. 2018*

- Designed and implemented DSol, the first practical Ethereum smart contract decompiler
- DSol translates from EVM bytecode to Solidity, a high-level smart contract programming language
- Designed an intermediate representation suitable for translation of smart contracts
- Implemented program analyses, including expression propagation, dead-code elimination, and loop detection