# Master’s Thesis: Enclave Hardening for Private ML

### Project topic

One major problem in the insurance industry is **information asymmetry**. This means the insurer has significantly less information about the insured object (e.g., a car or a house) than the customers applying for the insurance. This makes risk assessment and insurance pricing challenging. The way that the insurers traditionally address this problem is through **questionnaires**, where the customers are expected to answer questions regarding the insured object (e.g., model of the car, age of the house).

**Cyber insurance** is a new form of insurance that has emerged recently. In a typical cyber insurance scenario, a potential insurance customer seeks protection against so-called “cyber risks” that may include data leaks, loss of data, important IT service downtime, and so on. The problem of information asymmetry is especially prevalent in cyber insurance because customers are hesitant to reveal the full details of their IT system configuration, management, and security practices. Customers are, rightly so, concerned that such information could be used to discriminate against them during the insurance pricing or potential future claim settlement.

The recent advances in **trusted execution environments (TEEs)** and **privacy-preserving machine learning** provide new opportunities to address the problem of information asymmetry in insurance in general and in cyber insurance in particular. One possible solution is to create a hardware-protected enclave (e.g., using technologies such as Intel’s SGX) and enabled insurance customers to send their questionnaire responses to the enclave which in turn trains a privacy-preserving machine learning model using responses collected from many customers. Techniques like **differential privacy (DP)** can be used to ensure that the trained model that is in then end exported by the enclave will not reveal individual customer’s private details. That is, the insurance company learns a useful aggregate model but at the same time, each individual customer’s privacy is protected.

Recent studies have shown that if such a privacy-preserving data collection and processing system are available, users can be **more willing to share private data with data collectors**, compared to a traditional situation where the data collector learns the users’ full data in plaintext. Such research results give us a reason to believe that realizing the above-outlined system could make insurance customers more willing to share their IT system configuration and security practices with the insurer, and thus address the pressing problem of information asymmetry in cyber insurance.

One of the remaining technical challenges to realizing the above vision is **information leakage** from enclaves. Although TEEs like SGX isolate enclave’s execution from other software on the same platform through hardware enforcements, recent research has demonstrated that various **side-channels**, for example, based on shared physical resources like caches on the computing platform, can leak information about the enclaves execution and reveal secrets that the enclave is processing to the adversary (in this case, the adversary is the insurance company). If such information leakage is not addressed, the above-outlined approach of private data sharing does not work.

The goal of this master’s thesis project is to address such information leakage problems through **enclave hardening.** One example technique is to transform the enclave’s code such that its execution trace and data access patterns are “oblivious” to the data that it is processing. In practice, this means that the enclave produces the same execution trace and the same data access pattern regardless of the input data which may be secret (in our case, questionnaire answers by insurance customer).

Additionally, this master’s thesis can investigate the manipulation of the enclave’s state through so-called **rollback attacks**, their implications on differential privacy guarantees, and potential defenses against such attacks. Besides side-channel leakage and rollbacks, in this master’s thesis, the enclave can also be hardened against classical **memory safety** problems. The exact scope and extent of the project will be defined during the course of the master’s thesis project.

### Preliminary tasks

* **Task 0: Background reading** on enclave side-channels, hardening techniques, differentially-private decision tree algorithms.
* **Task 1: Port an existing Python implementation** of ML algorithm into SGX enclave. The existing code implements a Differentially-Private Gradient Boosted Decision Tree algorithm. Evaluate baseline performance.
* **Task 2: Harden the enclave** using defensive techniques that are known in the research literature. Consider both control flow and data access patterns. Consider both algorithmic and microarchitectural protections. Evaluate hardening performance overhead.
* **Task 3: Investigate privacy attacks through enclave rollbacks**. Evaluate their impact on differential privacy guarantees and develop defenses against such attacks. Implement a chosen defense.
* **Task 4: Write the thesis.**

### Reading list

#### Privacy-Preserving Gradient Boosting Decision Trees

* Theo’s thesis
* The paper that was the basis for Theo’s thesis work: <https://ojs.aaai.org/index.php/AAAI/article/view/5422/5278>
* Tutorials
  + Decision Trees: <https://gdcoder.com/decision-tree-regressor-explained-in-depth/>
  + Gradient boosted decision trees: <https://towardsdatascience.com/machine-learning-part-18-boosting-algorithms-gradient-boosting-in-python-ef5ae6965be4>
  + Youtube video (first out of 4): <https://www.youtube.com/watch?v=3CC4N4z3GJc>

#### Attack examples

* **Paging attack** called “controlled-channel attacks”: <https://ieeexplore.ieee.org/document/7163052>
* **Cache attacks**: <https://www.usenix.org/system/files/conference/woot17/woot17-paper-brasser.pdf>  
  <https://dl.acm.org/doi/pdf/10.1145/3065913.3065915>
* **Branch prediction:** <https://www.usenix.org/system/files/conference/usenixsecurity17/sec17-lee-sangho.pdf>
* Single stepping: <https://dl.acm.org/doi/abs/10.1145/3152701.3152706>
* Fine-grained control-flow attack: <https://arxiv.org/pdf/2005.11516.pdf>

#### Defensive techniques

* Oblivious execution system **Raccoon**: <https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-rane.pdf>
* Another oblivious execution solution **ZeroTrace**: <https://www.ndss-symposium.org/wp-content/uploads/2018/02/ndss2018_02B-4_Sasy_paper.pdf>
* A recent paper on control and data flow linearization called **Constatine**: <https://arxiv.org/pdf/2104.10749.pdf>

#### Hardened applications

* **BITE:** side-channel hardening of Bitcoin light clients: <https://www.usenix.org/system/files/sec19-matetic.pdf>

#### Rollback attacks and defenses

* **ROTE:** distributed counter: <https://www.usenix.org/system/files/conference/usenixsecurity17/sec17-matetic.pdf>