Practical Applications of Client-Side Trusted Computing

David Goltzsche, 2018-04-23
3rd year PhD student at distributed systems group, TU Braunschweig, Germany
Research area: trusted execution, distributed systems
Advisor: Rüdiger Kapitza
Overview

Problem: Loading computations to untrusted clients is limited.

Current best practice: Avoidance of loading or expensive recomputations.

Goal: Enable secure loading using client-side trusted computing.

Consequence: New paradigm for system design, because changed assumptions.

How can existing systems be redesigned? Which entirely new use cases are possible?
Overview

Problem:

- Loading computations to untrusted clients is limited.

Current best practice:

- Avoidance of loading or expensive recomputations.

Goal:

- Enable secure loading using client-side trusted computing.

Consequence:

- New paradigm for system design, because changed assumptions.

How can existing systems be redesigned?

Which entirely new use cases are possible?
- **Problem:** Offloading computations to untrusted clients is **limited**
Overview

- **Problem:** Offloading computations to untrusted clients is **limited**
- **Current best practice:** Avoidance of offloading or expensive recomputations

![Diagram showing server and client connections with untrusted clients marked with crosses.](image-url)
Overview

- **Problem:** Offloading computations to untrusted clients is **limited**

- **Current best practice:** Avoidance of offloading or expensive recomputations
Overview

- **Problem:** offloading computations to untrusted clients is **limited**
- **Current best practice:** avoidance of offloading or expensive recomputations
- **Goal:** enable secure offloading using client-side trusted computing
Overview

- **Problem:** offloading computations to untrusted clients is limited

- **Current best practice:** avoidance of offloading or expensive recomputations

- **Goal:** enable secure offloading using client-side trusted computing

- **Consequence:** New paradigm for system design, because changed assumptions
  - How can existing systems be **redesigned**?
  - Which entirely **new use cases** are possible?
Trusted Execution Environments

- How to make clients trusted?

  - Trusted Execution Environments (TEEs)
    - Data and execution protection
    - Memory encryption
    - Remote attestation

  - Implementations
    - Intel SGX: available on commodity hardware
    - Other vendors expected to follow
    - Research: Komodo [Ferraiuolo et al., SOSP'17]
Use Cases of Client-Side TEEs

- JavaScript in Web Browsers @EuroSec'17
  - **Problem:** Recomputation in back-end of web application
  - **Solution** **TRUSTJS:** trusted client-side execution of JS
Use Cases of Client-Side TEEs

- **JavaScript in Web Browsers @EuroSec’17**
  - **Problem:** Recomputation in back-end of web application
  - **Solution** **TRUSTJS:** trusted client-side execution of JS

- **Network Middleboxes @DSN’18**
  - **Problem:** Client-side offloading **not considered** so far
  - **Solution** **ENDBOX:** client-side middlebox functions
Use Cases of Client-Side TEEs

- **JavaScript in Web Browsers** \(\text{@EuroSec'17}\)
  - **Problem:** Recomputation in back-end of web application
  - **Solution** **TRUSTJS:** trusted client-side execution of JS

- **Network Middleboxes** \(\text{@DSN'18}\)
  - **Problem:** Client-side offloading **not considered** so far
  - **Solution** **ENDBOX:** client-side middlebox functions

- **Volunteer Computing Systems**
  - **Problem:** Jobs replicated to other clients to stop cheaters
  - **Solution** **TRUVC:** trusted volunteer computing