1. Problem

- Cloud services process sensitive image content
  
  Ex.: Facebook, Instagram, Google...
- Image processing requires access to raw image data on the cloud
  
  Ex.: Rescaling for thumbnail generation, image filters...
- Outstanding server vulnerabilities may lead to sensitive image data disclosure
  
  Hackers or malware may exploit application bugs or OS misconfiguration

2. Challenges

1. Basic end-to-end encryption is too inflexible
   - Image content becomes exposed to the OS when unencrypted
   - Hard to securely maintain encryption keys

2. OS-based sandboxing requires large TCB
   - Image transformations inside an OS service
   - Secure channel between client and OS
   - Solves exposure of content to the server app
   - But depends on large Trusted Computing Base

3. Darkroom

Our approach: develop a Trusted Execution Environment for the cloud based on ARM TrustZone

- ARM is an alternative architecture for the cloud (less energy consumption)
- A conceptually similar approach can be implemented using Intel SGX (secure enclaves)

4. Implementation

- Hardware:
  NXP (Freescale) i.MX53 QSB
- Darkroom Kernel:
  - Adapted Genode’s micro-kernel
  - Shared memory for data communication between normal and secure worlds
- Cryptographic engine: adapted RSA and AES implementation from the mbed_TLS library
- Image Processing Engine: small set of simple transformation functions

5. Preliminary Evaluation

Compare Darkroom performance running in NW and SW

- Constant penalty for using Darkroom across all transformations
- Negligible overhead from the context-switch between worlds

So... Small overhead by Darkroom for image processing

5. Conclusions

- Darkroom: an ARM TrustZone based system for secure image processing on the cloud
- Provides isolation between potentially compromised rich OS and the Darkroom image processing engine
- Image processing using Darkroom adds reduced overhead