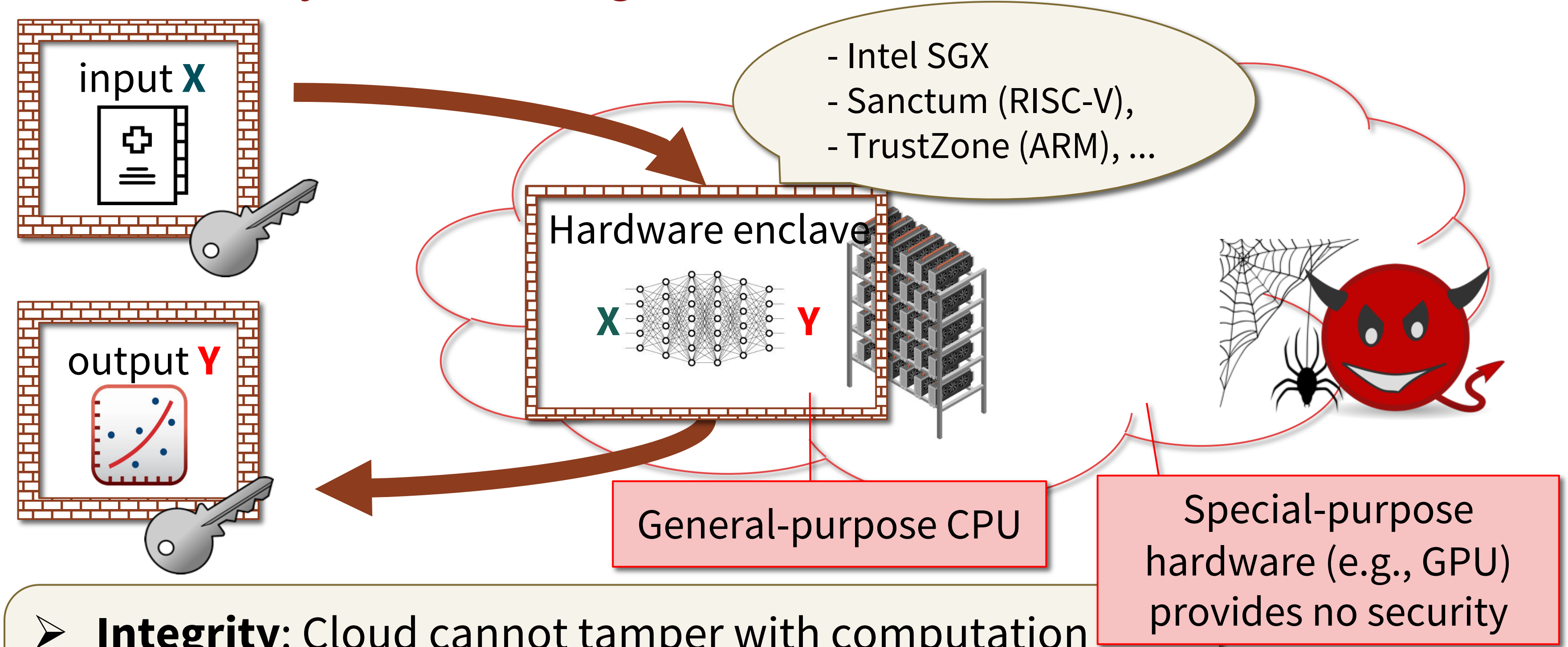
Slalom:
Fast, Verifiable and Private Execution of Neural Networks in Trusted Hardware

Florian Tramèr & Dan Boneh

ICLR, New Orleans

May 7th 2019

Securely outsourcing ML inference with hardware isolation



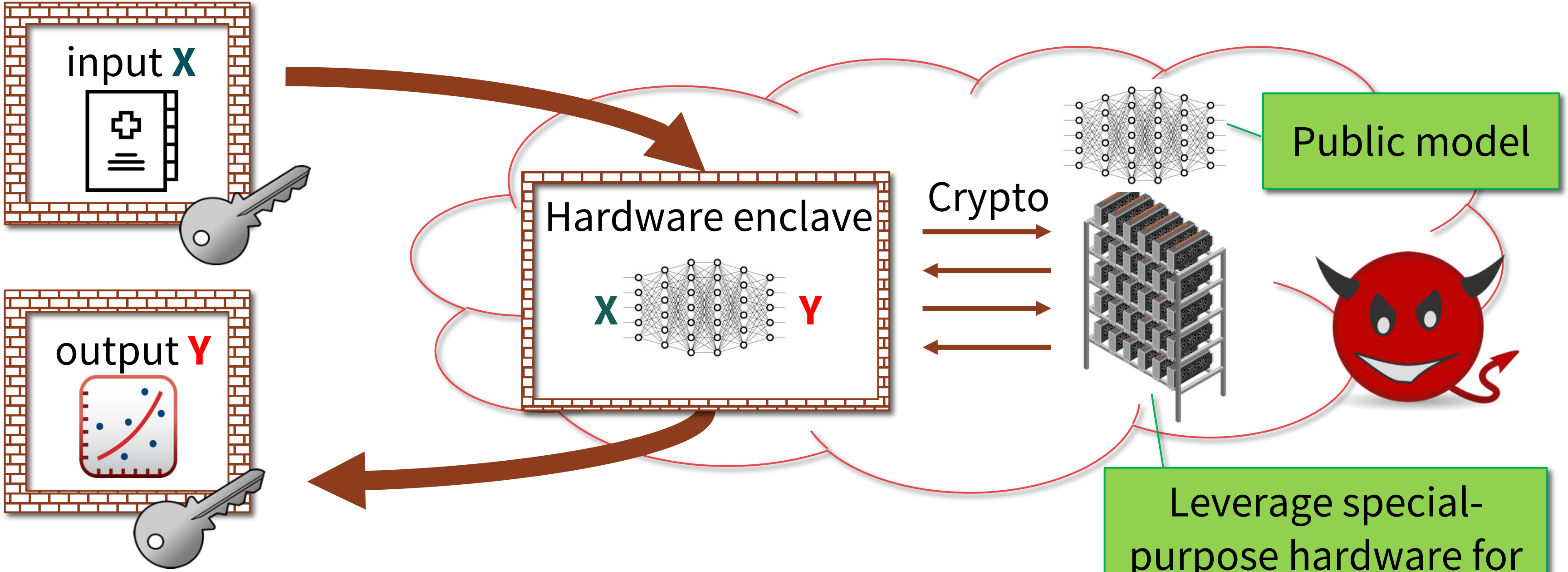
- Intel SGX
- Sanctum (RISC-V),
- TrustZone (ARM), ...

General-purpose CPU

Special-purpose hardware (e.g., GPU) provides no security

- ➢ **Integrity**: Cloud cannot tamper with computation
- ➢ **Privacy**: Integrity + Cloud does not learn inputs
- ➢ **Model Privacy**: Cloud does not learn model

Slalom: Outsource ML from CPU enclave to special-purpose hardware



Leverage special-purpose hardware for higher efficiency

- **Integrity**: Cloud cannot tamper with computation
- **Privacy**: Integrity + Cloud does not learn inputs
- ~~**Model Privacy**: Cloud does not learn model~~

Outsourcing ML inference using cryptography

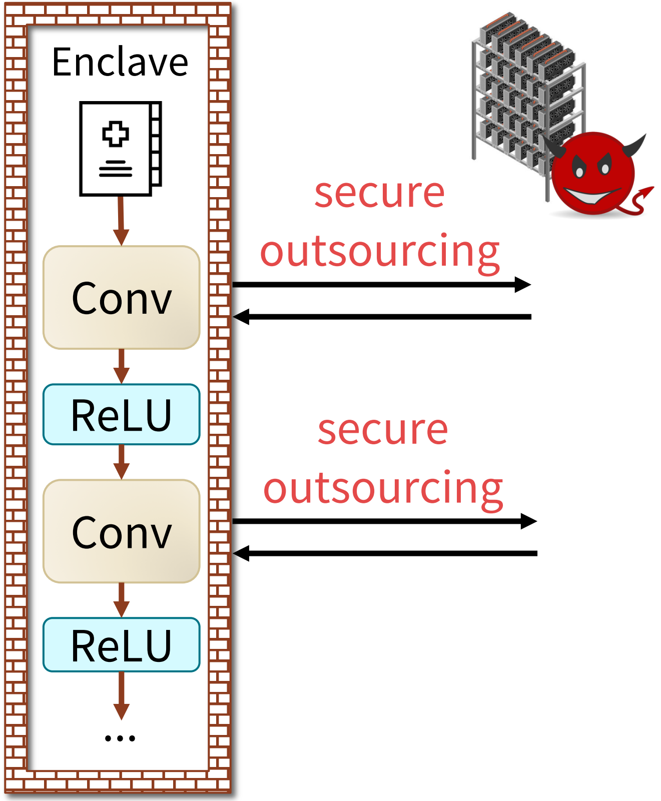
Slalom uses cryptographic protocols to securely outsource all linear layers from the enclave to a GPU.

⚠

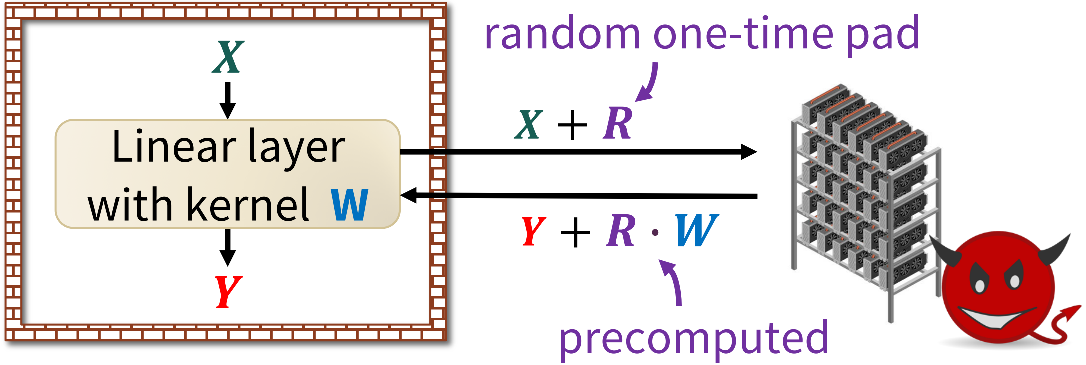
- Crypto protocols have high communication costs
  - 💡 Enclave processor and GPU are co-located
  - For VGG16, Slalom sends **50MB** of data from the enclave to the GPU per inference

⚠

- Crypto protocols are very efficient for securely outsourcing linear functions
  - 💡 Most of the computation in a DNN is linear (convolutions, dense, etc.)
  - E.g., **~99%** for VGG16 and MobileNet



How to securely outsource a matrix product



- **Integrity**:
  - › Verify that $Y = X \cdot W$
  - › Check $Y \cdot \vec{r} \overset{?}{=} X \cdot (W \cdot \vec{r})$ [Freivalds 1977]

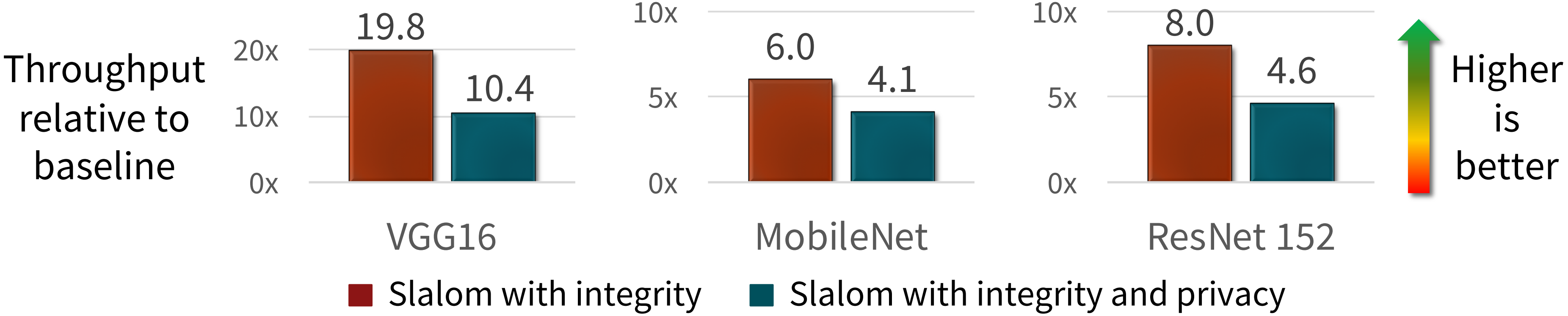
Verify a matrix product with a few inner products
(generalizes to arbitrary linear layer)

- **Privacy**:
  - › Evaluate model on random data R in offline pre-processing phase
  - › Store $(R, R \cdot W)$ in the enclave and use these to encrypt & decrypt the communication with the GPU

Evaluation

Evaluate DNN in TEE

- Intel SGX + Nvidia Titan XP
- Throughput for ImageNet inference
- **Goal: Slalom (TEE⟷GPU) ≫ TEE$_{\text{baseline}}$**



Slalom is **10-20x slower** than evaluating on GPU **(with no security guarantees)**

⇒ But, Slalom only utilizes the GPU ~10% of the time

⇒ Multiple CPU enclaves can outsource to the same GPU

Conclusions & Open Problems

- **Slalom allows efficient and secure outsourcing of sensitive DNN computations to the cloud**
  - Hardware isolation protects privacy & integrity, but is slow
  - Slalom uses cryptography to leverage fast special-purpose hardware without any isolation guarantees

- **What about training?**
  - Integrity: Freivalds' still works ☺
  - Privacy: Model itself should remain secret ☹

Poster @4:30 - Great Hall BC #44

https://arxiv.org/abs/1806.03287

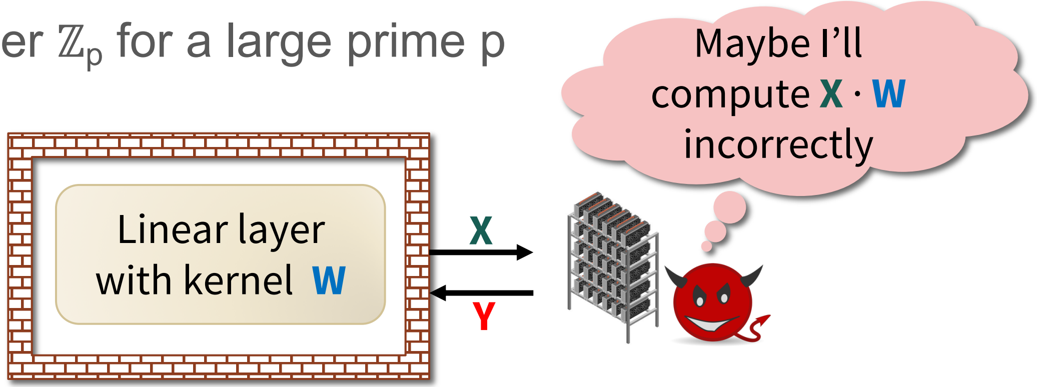
https://github.com/ftramer/slalom

https://floriantramer.com

How to securely outsource a linear layer

- **Quantization**: Evaluate a DNN over $\mathbb{Z}_p$ for a large prime p

- **Integrity**: Freivalds' 1977


Maybe I'll compute X · W incorrectly
Linear layer with kernel W
X
Y

$$Y \overset{?}{=} X \cdot W$$

$$\text{check } Y \cdot r \overset{?}{=} X \cdot (W \cdot r)$$

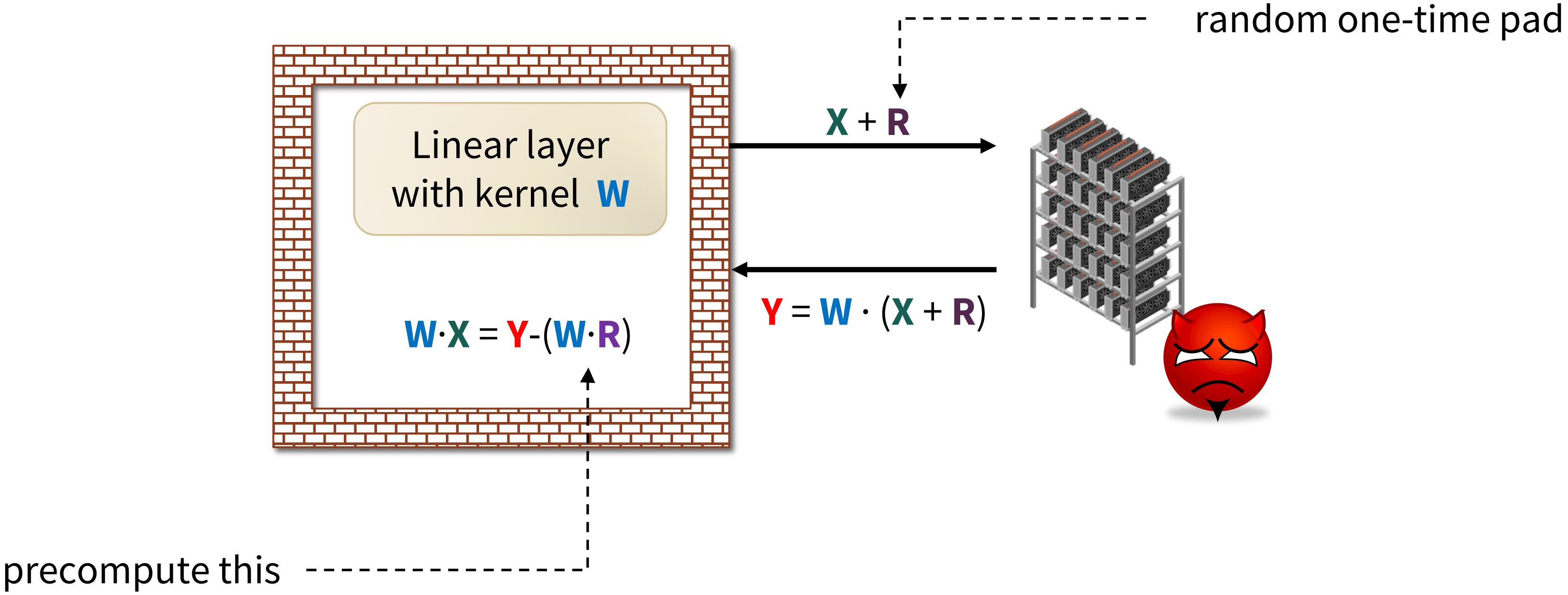
random vector

Verify any linear layer with a few inner products ≈ $O(n^2)$ instead of $O(n^3)$

- **Privacy**: precomputed "one-time pads"
  - See paper for details

Evaluate model on **random** data in offline preprocessing phase

Privacy with precomputed one-time pads



random one-time pad

$\mathbf{X} + \mathbf{R}$

Linear layer with kernel $\mathbf{W}$

$\mathbf{Y} = \mathbf{W} \cdot (\mathbf{X} + \mathbf{R})$

$\mathbf{W}\cdot\mathbf{X} = \mathbf{Y}-(\mathbf{W}\cdot\mathbf{R})$

precompute this