Aalto University

Securing Cloud-assisted Services

N. Asokan

https://asokan.org/asokan/
@nasokan

14 June 2019 @Uber

Helsinki, Finland

Home to leading universities

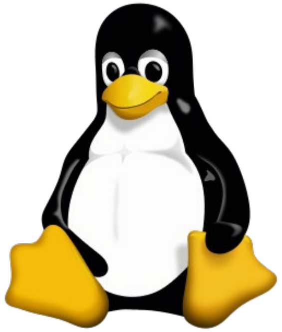
- **University of Helsinki** (top 100 overall)
- **Aalto[1] University** (top 100 in Computer Science)

Innovation hub

- Local giants: Nokia, Ericsson
- Security innovators: F-Secure, SSH, ...
- Recent arrivals: Intel, Samsung, Huawei, ...
- New tigers: Rovio, Supercell, ..., lots of startups

1. http://en.wikipedia.org/wiki/Alvar_Aalto

Software Made in Finland



Larry Ewing, http://isc.tamu.edu/~lewing/linux/

http://miss-nessa.deviantart.com/art/five-angry-birds-251226902

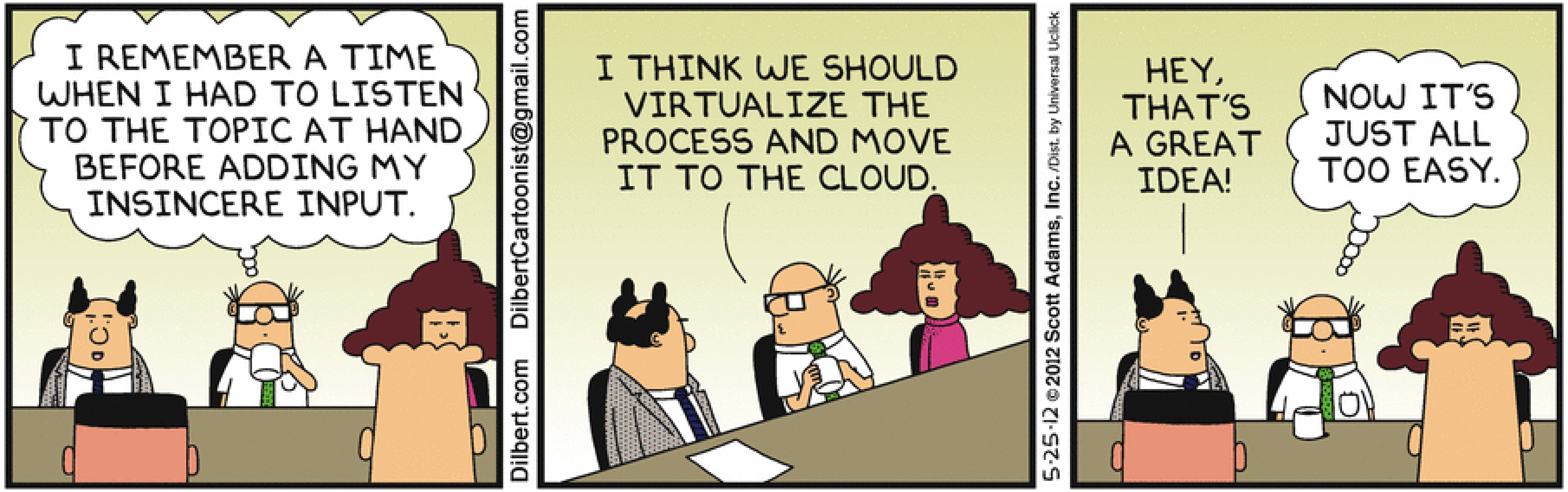
https://www.flickr.com/photos/85217387@N04/8638067405

https://www.coderew.com/tech/computers/ssh-add-user-remotely-script/

https://pixabay.com/en/protocol-irc-chat-icon-27279/

By Source, Fair use, https://en.wikipedia.org/w/index.php?curid=17119753

Services are moving to “the cloud”



http://dilbert.com/strip/2012-05-25

Services are moving to “the cloud”

Example: cloud storage

Example: cloud-based malware scanning service

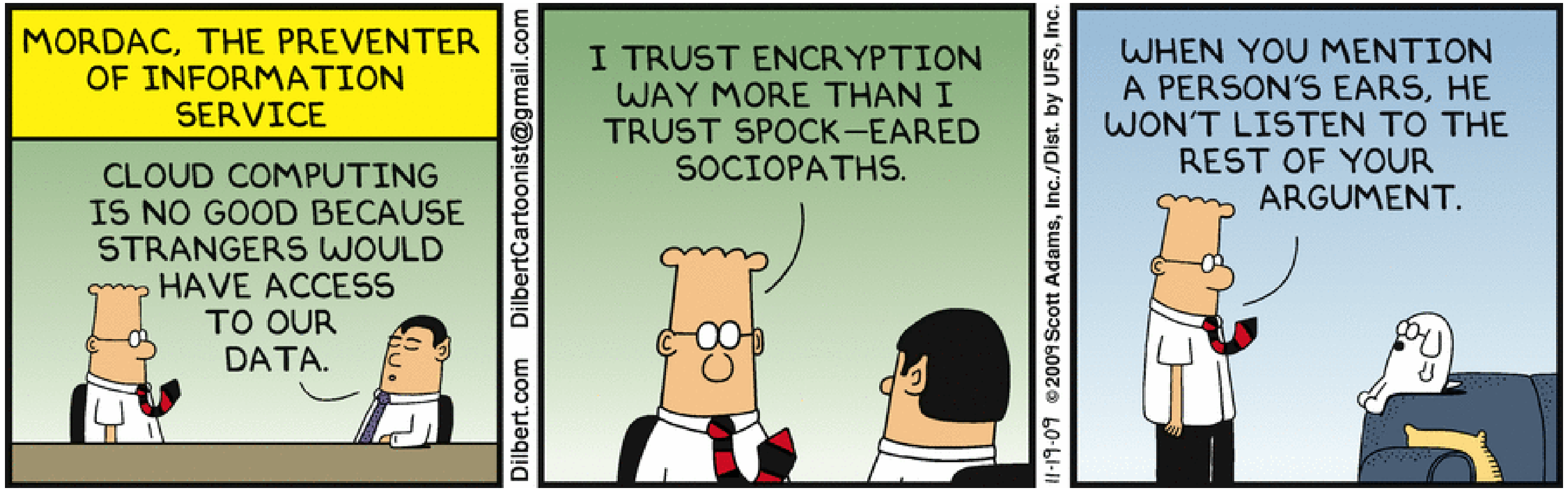
…

Securing cloud storage

Client-side encryption of user data is desirable

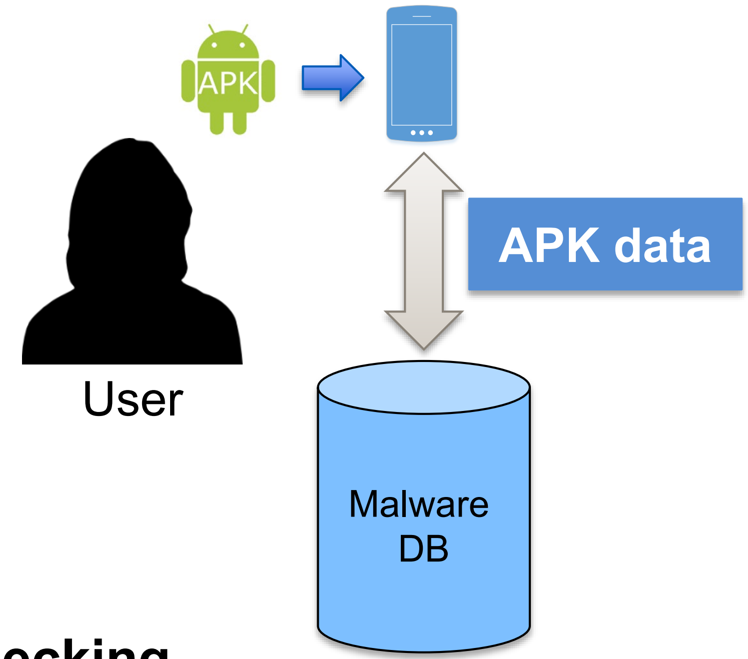
But naïve client-side encryption conflicts with

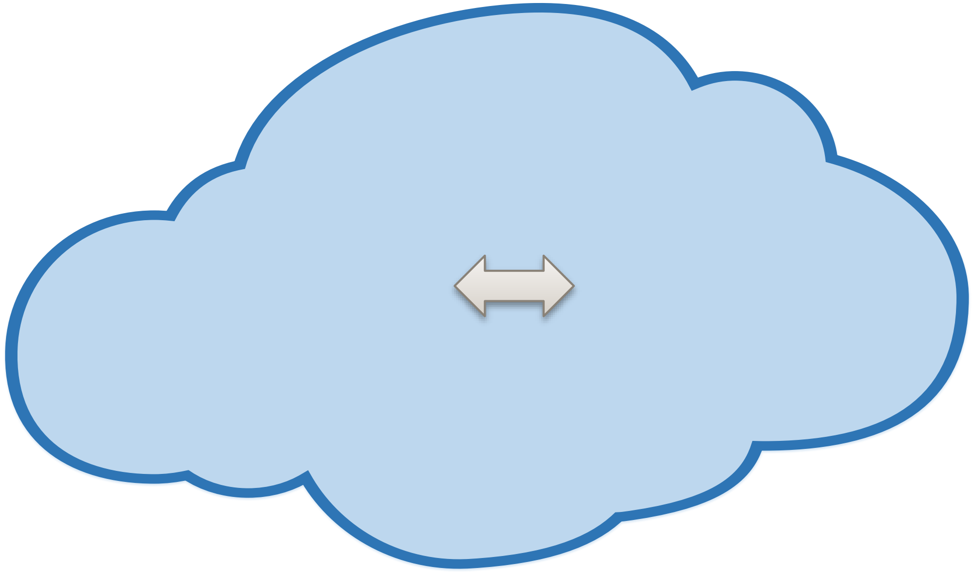
- Storage provider business requirement: deduplication ([LPA15] ACM CCS '15, [LDLA18] CT RSA '18)
- End user usability requirement: multi-device access ([P+AS18] IEEE IC '18, CeBIT '16)



http://dilbert.com/strip/2009-11-19

Malware checking





On-device checking

- High communication and computation costs
- Database changes frequently
- Database is revealed to everyone

Cloud-based checking

- Minimal communication and computation costs
- Database can change frequently
- Database is not revealed to everyone
- User privacy at risk!

Cloud-based malware scanning service

Needs to learn about apps installed on client devices

Can therefore infer personal characteristics of users

Predicting User Traits From a Snapshot of Apps Installed on a Smartphone

Suranga Seneviratne[a,b]
suranga.seneviratne@nicta.com.au
Aruna Seneviratne[a,b]
aruna.seneviratne@nicta.com.au
Prasant Mohapatra[c]
prasant@cs.ucdavis.edu
Anirban Mahanti[b]
anirban.mahanti@nicta.com.au

[a]School of EET, University of New South Wales, Australia
[b]NICTA, Australia
[c]Department of Computer Science, University of California, Davis

http://dx.doi.org/10.1145/2636242.2636244

Proceedings of the Tenth International AAAI Conference on Web and Social Media (ICWSM 2016)

You Are What Apps You Use: Demographic Prediction Based on User's Apps

Eric Malmi
Verto Analytics and Aalto University
Espoo, Finland
eric.malmi@aalto.fi

Ingmar Weber
Qatar Computing Research Institute
Doha, Qatar
iweber@qf.org.qa

http://www.aaai.org/ocs/index.php/ICWSM/ICWSM16/paper/view/13047

Aalto University

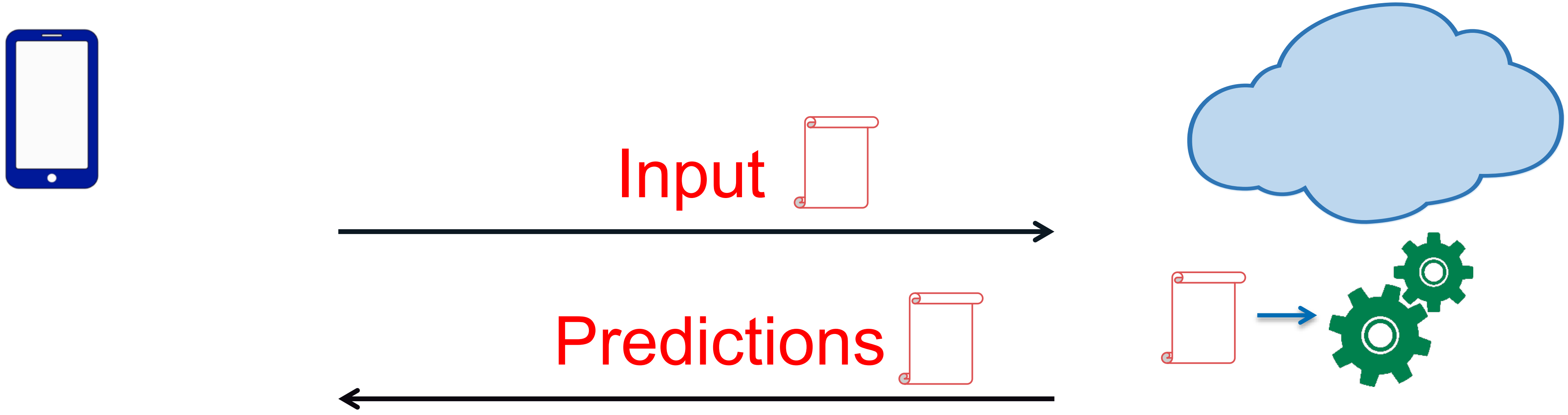
Oblivious Neural Network Predictions via MiniONN Transformations

N. Asokan
http://asokan.org/asokan/
@nasokan

(Joint work with Jian Liu, Mika Juuti, Yao Lu)

By Source, Fair use, https://en.wikipedia.org/w/index.php?curid=54119040

Machine learning as a service (MLaaS)



violation of clients' privacy

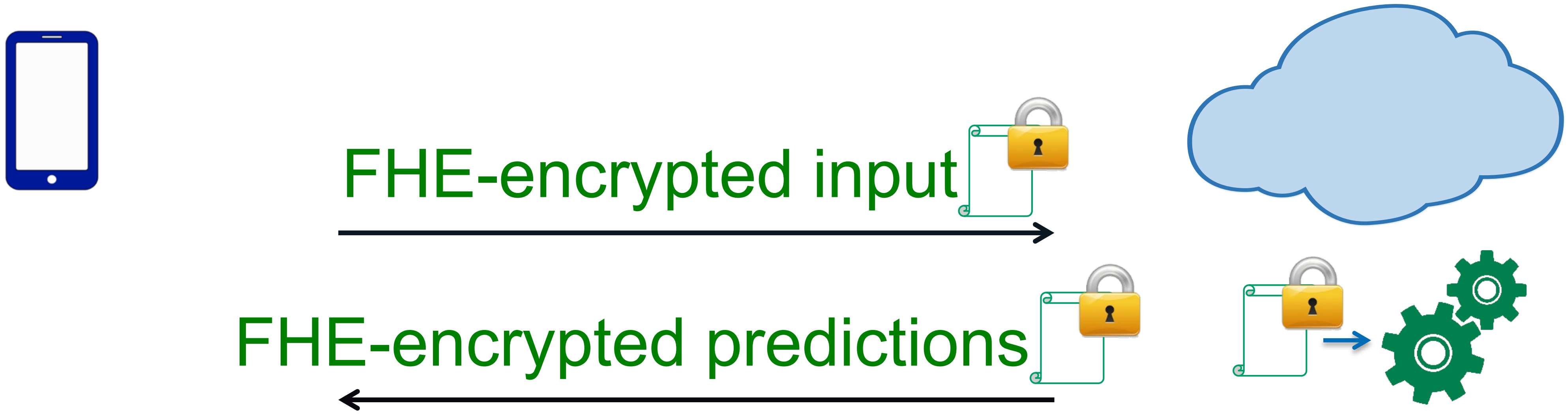
Oblivious Neural Networks (ONN)

Given a neural network, is it possible to make it oblivious?

- server learns nothing about clients' input

- clients learn nothing about the model

Example: CryptoNets



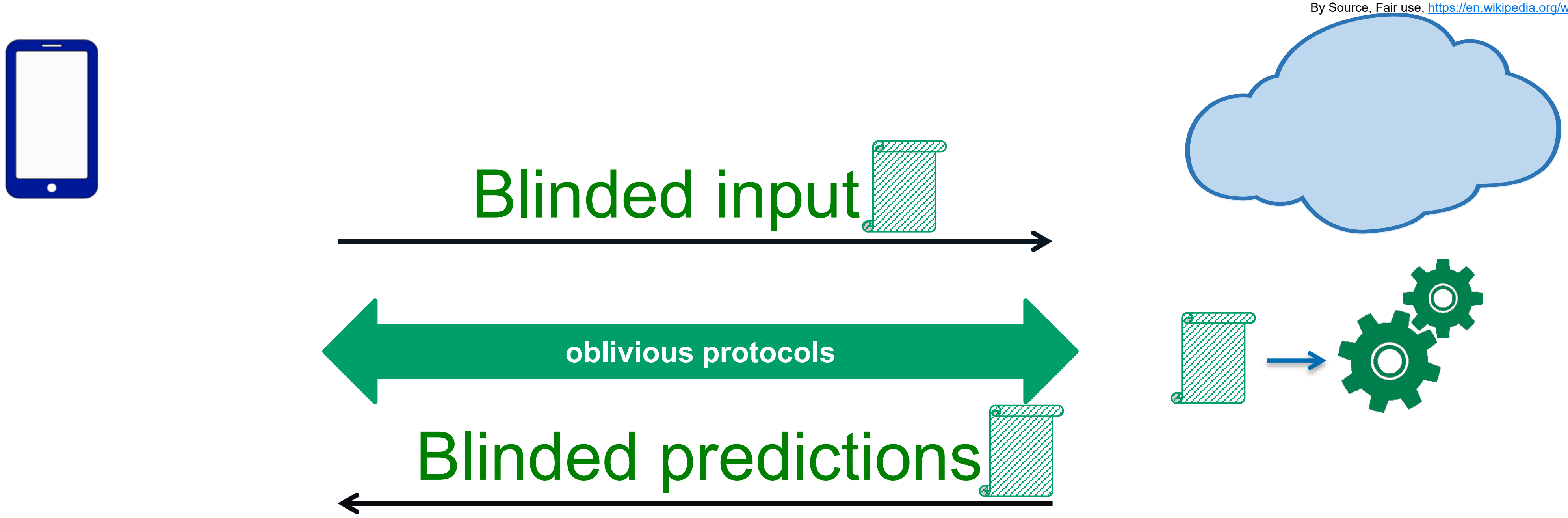
- High throughput for batch queries from same client
- High overhead for single queries: 297.5s and 372MB (MNIST dataset)
- Cannot support: high-degree polynomials, comparisons, …

[GDLLNW16] CryptoNets, ICML 2016

FHE: Fully homomorphic encryption (https://en.wikipedia.org/wiki/Homomorphic_encryption)

MiniONN: Overview

By Source, Fair use, https://en.wikipedia.org/w/index.php?curid=54119040

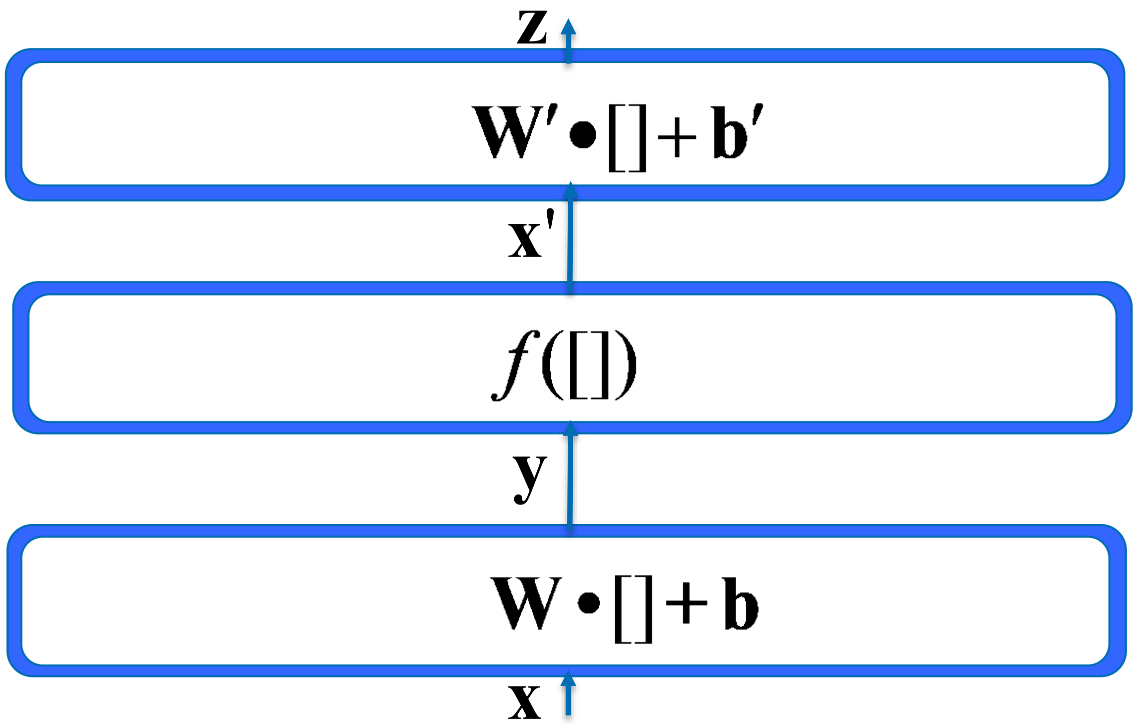


- Low overhead: ~1s
- Support all common neural networks

[Skip to performance]

Example $\mathbf{z} = \mathbf{W}' \bullet f(\mathbf{W} \bullet \mathbf{x} + \mathbf{b}) + \mathbf{b}'$

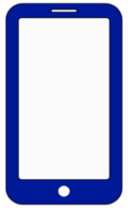
$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \mathbf{W} = \begin{bmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & w_{2,2} \end{bmatrix}, \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}, \mathbf{W}' = \begin{bmatrix} w'_{1,1} & w'_{1,2} \\ w'_{2,1} & w'_{2,2} \end{bmatrix}, \mathbf{b}' = \begin{bmatrix} b'_1 \\ b'_2 \end{bmatrix}$$

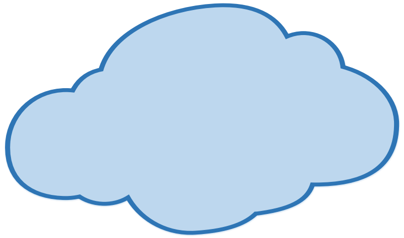


All operations are in a finite field Z_N

Skip to performance

Core idea: use secret sharing for oblivious computation

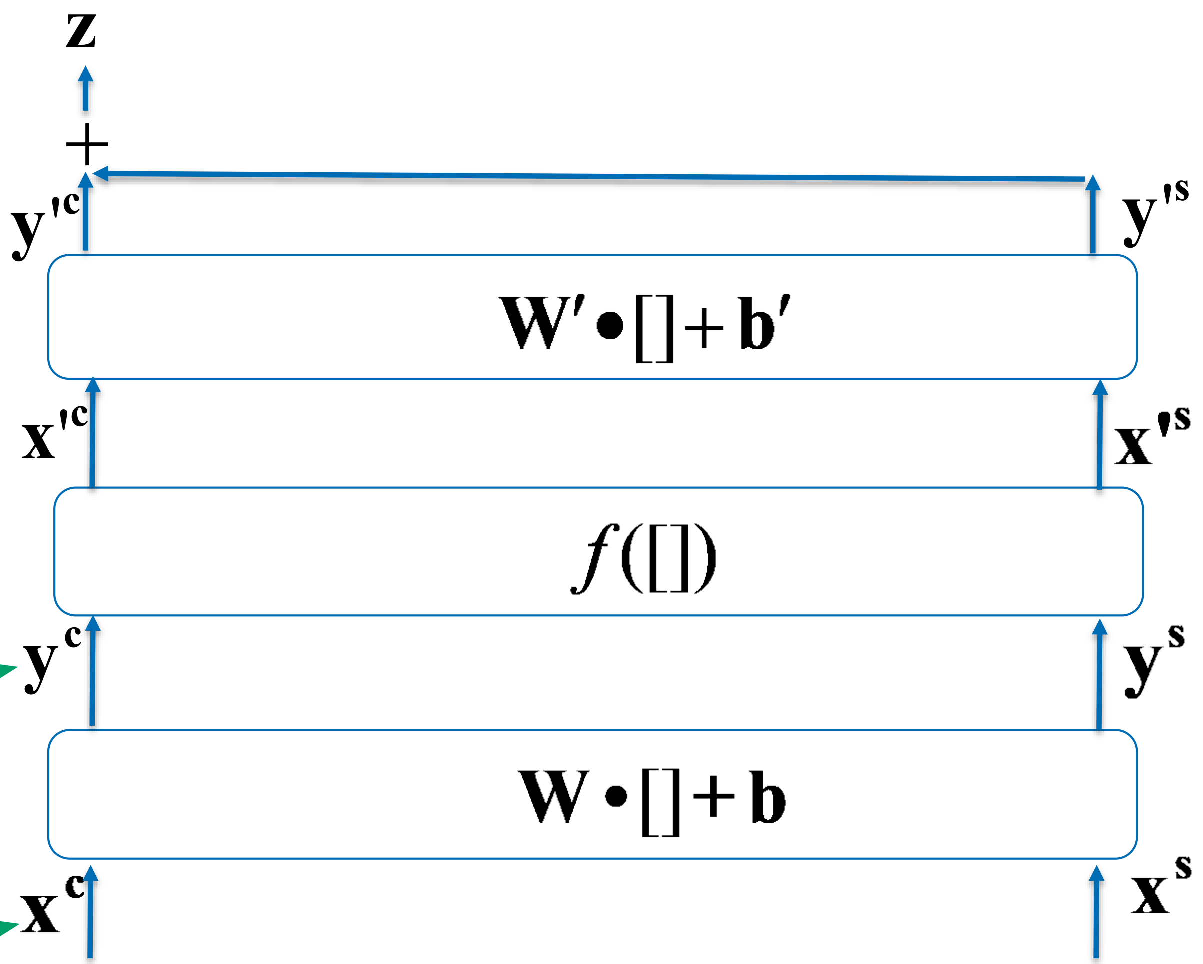




[LJLA17] MiniONN, ACM CCS 2017
https://eprint.iacr.org/2017/452

client & server have shares y^c and y^s s.t. $y^s + y^c = y$

client & server have shares x^c and x^s s.t. $x^s + x^c = x$



$(y'^c + y'^s = y')$

$(x'^c + x'^s = x')$

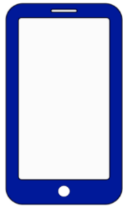
$(y^c + y^s = y)$

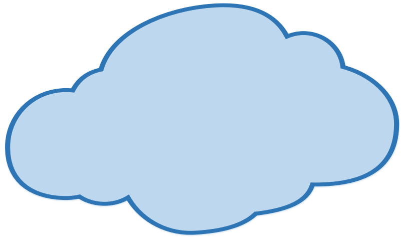
$(x^c + x^s = x)$

Use efficient cryptographic primitives (2PC, additively homomorphic encryption)

Skip to performance

Secret sharing initial input $\mathbf{x}$





$$x_1^c, x_2^c \xleftarrow{\$} Z_N$$

$$\xrightarrow{x_1^s := x_1 - x_1^c, \quad x_2^s := x_2 - x_2^c}$$

Note that **x^c** is independent of **x**. Can be **pre-chosen**

Skip to performance

Oblivious linear transformation $\mathbf{W} \bullet \mathbf{x} + \mathbf{b}$

$$= \begin{bmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & w_{2,2} \end{bmatrix} \bullet \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & w_{2,2} \end{bmatrix} \bullet \begin{bmatrix} x_1^s + x_1^c \\ x_2^s + x_2^c \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

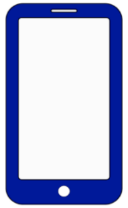
$$= \begin{bmatrix} w_{1,1}(x_1^s + x_1^c) + w_{1,2}(x_2^s + x_2^c) + b_1 \\ w_{2,1}(x_1^s + x_1^c) + w_{2,2}(x_2^s + x_2^c) + b_2 \end{bmatrix} = \begin{bmatrix} \boxed{\begin{matrix} w_{1,1}x_1^s + w_{1,2}x_2^s + b_1 \\ w_{2,1}x_1^s + w_{2,2}x_2^s + b_2 \end{matrix}} & \begin{matrix} + \\ + \end{matrix} & \boxed{\begin{matrix} w_{1,1}x_1^c + w_{1,2}x_2^c \\ w_{2,1}x_1^c + w_{2,2}x_2^c \end{matrix}} \end{bmatrix}$$

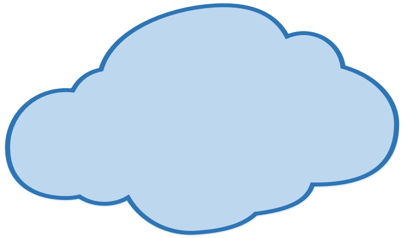
Compute locally by the server

Dot-product

Skip to performance

Oblivious linear transformation: dot-product





$$E(w_{1,1}), \overbrace{E(w_{1,2}), E(w_{2,1})}^{\text{Homomorphic Encryption with SIMD}}, E(w_{2,2})$$

$$r_{1,1}, r_{1,2}, r_{2,1}, r_{2,2} \xleftarrow{\$} Z_N \qquad \longleftarrow$$

$$c_{1,1} = E(w_{1,1}x_1^c - r_{1,1})$$

$$c_{1,2} = E(w_{1,2}x_2^c - r_{1,2}) \qquad \xrightarrow{c_{1,1}, c_{1,2}, c_{2,1}, c_{2,2}}$$

$$D(c_{1,1}), D(c_{1,2}), D(c_{2,1}), D(c_{2,2})$$

$$c_{2,1} = E(w_{2,1}x_1^c - r_{2,1})$$

$$c_{2,2} = E(w_{2,2}x_2^c - r_{2,2}) \qquad v_1 = r_{1,1} + r_{1,2} \qquad u_1 = w_{1,1}x_1^c + w_{1,2}x_2^c - (r_{1,2} + r_{1,1})$$

$$v_2 = r_{2,1} + r_{2,2} \qquad u_2 = w_{2,1}x_1^c + w_{2,2}x_2^c - (r_{2,1} + r_{2,2})$$

u + v = W•x^c; Note: **u**, **v**, and **W•x^c** are independent of **x**.
<**u**,**v**,**x^c**> generated/stored in a **precomputation phase**

Skip to performance

Oblivious linear transformation $\mathbf{W} \bullet \mathbf{x} + \mathbf{b}$

$$= \begin{bmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & w_{2,2} \end{bmatrix} \bullet \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & w_{2,2} \end{bmatrix} \bullet \begin{bmatrix} x_1^s + x_1^c \\ x_2^s + x_2^c \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

$$= \begin{bmatrix} w_{1,1}(x_1^s + x_1^c) + w_{1,2}(x_2^s + x_2^c) + b_1 \\ w_{2,1}(x_1^s + x_1^c) + w_{2,2}(x_2^s + x_2^c) + b_2 \end{bmatrix} = \begin{bmatrix} \boxed{\begin{matrix} w_{1,1}x_1^s + w_{1,2}x_2^s + b_1 \\ w_{2,1}x_1^s + w_{2,2}x_2^s + b_2 \end{matrix}} \begin{matrix} + \\ + \end{matrix} \boxed{\begin{matrix} w_{1,1}x_1^c + w_{1,2}x_2^c \\ w_{2,1}x_1^c + w_{2,2}x_2^c \end{matrix}} \end{bmatrix}$$

$$= \begin{bmatrix} \boxed{\begin{matrix} w_{1,1}x_1^s + w_{1,2}x_2^s + b_1 \\ w_{2,1}x_1^s + w_{2,2}x_2^s + b_2 \end{matrix}} \begin{matrix} + \\ + \end{matrix} \boxed{\begin{matrix} u_1 \\ u_2 \end{matrix}} \end{bmatrix} + \boxed{\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}}$$

Skip to performance

Oblivious linear transformation $\mathbf{W} \bullet \mathbf{x} + \mathbf{b}$

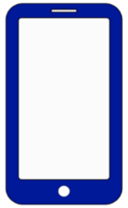
$$= \begin{bmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & w_{2,2} \end{bmatrix} \bullet \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & w_{2,2} \end{bmatrix} \bullet \begin{bmatrix} x_1^s + x_1^c \\ x_2^s + x_2^c \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

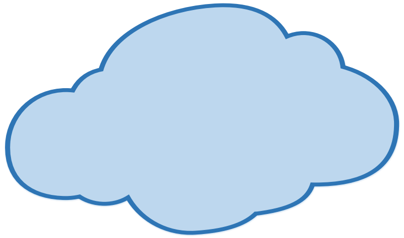
$$= \begin{bmatrix} w_{1,1}(x_1^s + x_1^c) + w_{1,2}(x_2^s + x_2^c) + b_1 \\ w_{2,1}(x_1^s + x_1^c) + w_{2,2}(x_2^s + x_2^c) + b_2 \end{bmatrix} = \begin{bmatrix} w_{1,1}x_1^s + w_{1,2}x_2^s + b_1 + w_{1,1}x_1^c + w_{1,2}x_2^c \\ w_{2,1}x_1^s + w_{2,2}x_2^s + b_2 + w_{2,1}x_1^c + w_{2,2}x_2^c \end{bmatrix}$$

$$= \begin{bmatrix} \boxed{\begin{matrix} w_{1,1}x_1^s + w_{1,2}x_2^s + b_1 + u_1 \\ w_{2,1}x_1^s + w_{2,2}x_2^s + b_2 + u_2 \end{matrix}} \end{bmatrix} + \begin{bmatrix} \boxed{\begin{matrix} v_1 \\ v_2 \end{matrix}} \end{bmatrix} := \begin{bmatrix} \boxed{\begin{matrix} y_1^s \\ y_2^s \end{matrix}} \end{bmatrix} + \begin{bmatrix} \boxed{\begin{matrix} y_1^c \\ y_2^c \end{matrix}} \end{bmatrix}$$

[Skip to performance](#)

Recall: use secret sharing for oblivious computation





https://eprint.iacr.org/2017/452

client & server have shares y^c and y^s s.t. $y^s + y^c = y$

client & server have shares x^c and x^s s.t. $x^s + x^c = x$

y^c $\qquad$ y^s $\qquad (y^c + y^s = y)$

$$W \bullet [\,] + b$$

x^c $\qquad$ x^s $\qquad (x^c + x^s = x)$

Skip to performance

Oblivious activation/pooling functions $f(y)$

Piecewise linear functions e.g.,

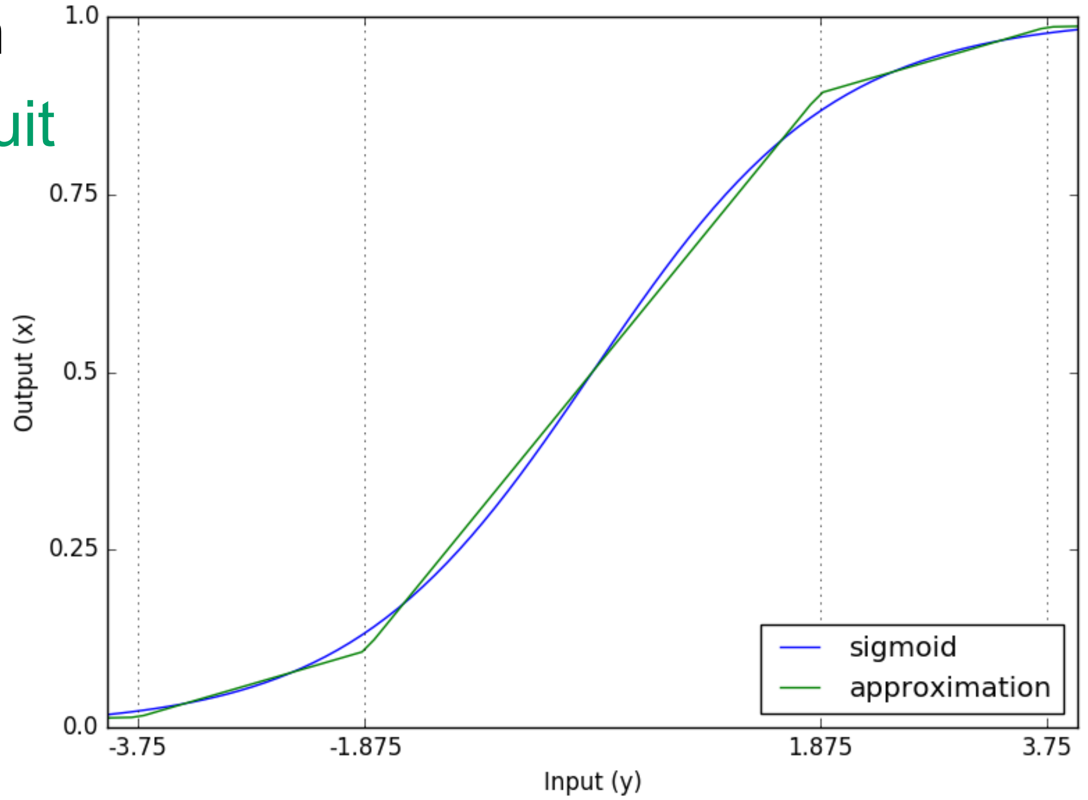
- ReLU: $x := \max(y, 0)$
- Oblivious ReLU: $x^s + x^c := \max(y^s + y^c, 0)$
  - easily computed obliviously by a garbled circuit

Skip to performance

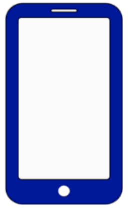
Oblivious activation/pooling functions $f(y)$

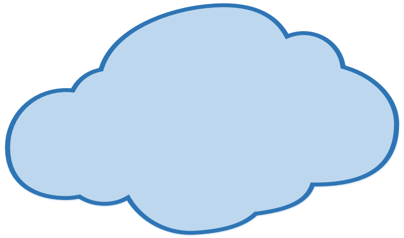
Smooth functions e.g.,

- Sigmoid: $x := 1/(1+e^{-y})$
- Oblivious sigmoid: $x^s + x^c := 1/(1+e^{-(y^s+y^c)})$
  - approximate by a piecewise linear function
  - then compute obliviously by a garbled circuit
  - empirically: ~14 segments sufficient



Combining the final result





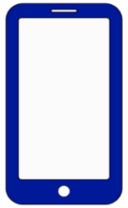
$$\xleftarrow{\quad y_1^s,\, y_2^s \quad}$$

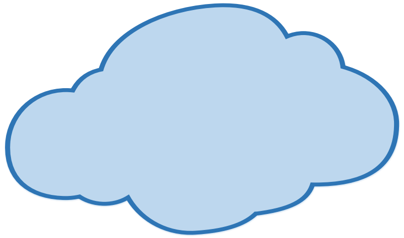
$$y_1 := y_1^s + y_1^c$$

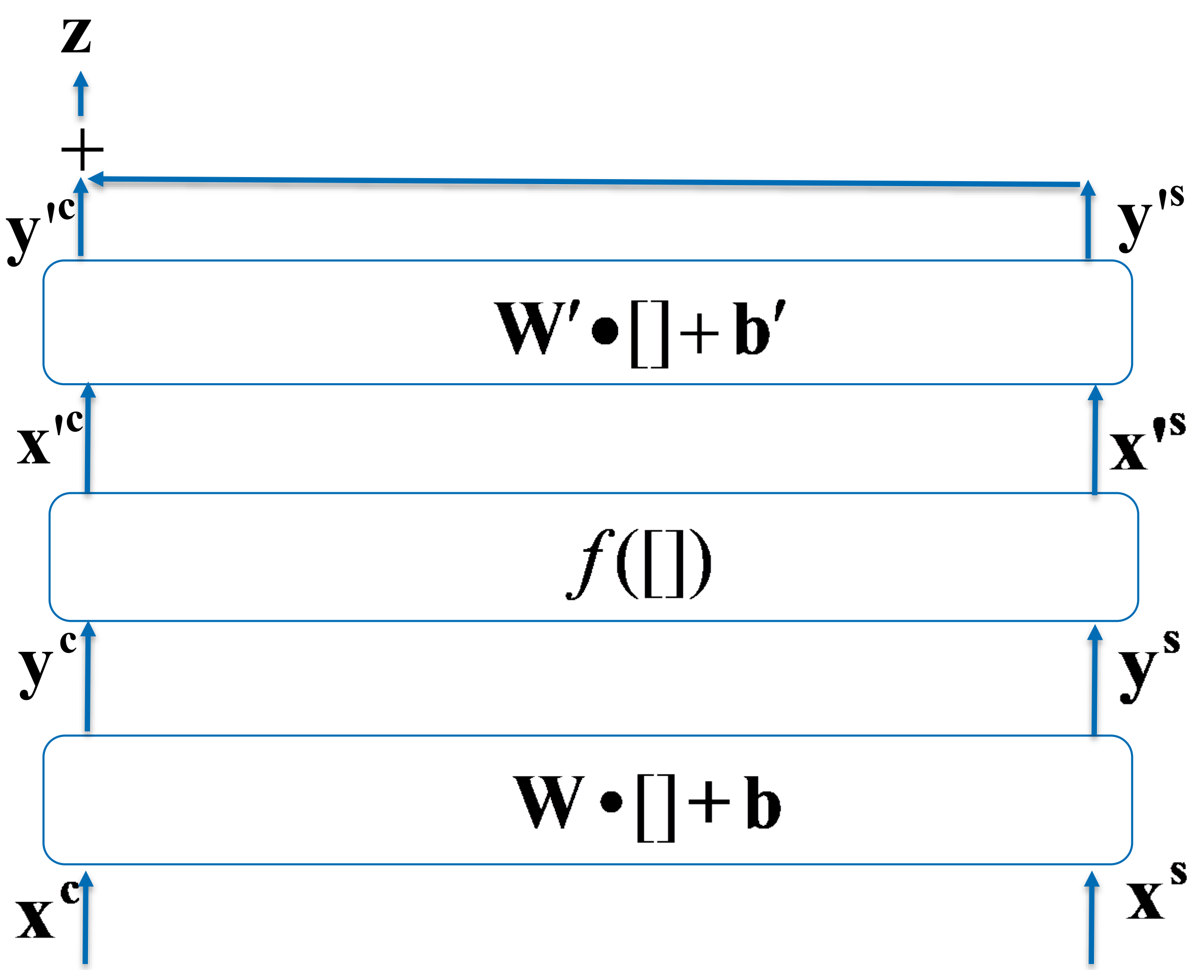
$$y_2 := y_2^s + y_2^c$$

They can jointly calculate max(y_1,y_2)
(for minimizing information leakage)

Recall: use secret sharing for oblivious computation







$(y'^c + y'^s = y')$

$(x'^c + x'^s = x')$

$(y^c + y^s = y)$

$(x^c + x^s = x)$

Skip to End

Performance (for single queries)

Model	Latency (s)	Msg sizes (MB)	Loss of accuracy
MNIST/Square	0.4 (+ 0.88)	44 (+ 3.6)	none
CIFAR-10/ReLU	472 (+ 72)	6226 (+ 3046)	none
PTB/Sigmoid	4.39 (+ 13.9)	474 (+ 86.7)	Less than 0.5% (cross-entropy loss)

Pre-computation phase timings in parentheses

PTB = Penn Treebank

Skip to End

MiniONN pros and cons

300-700x faster than CryptoNets

Can transform any given neural network to its oblivious variant

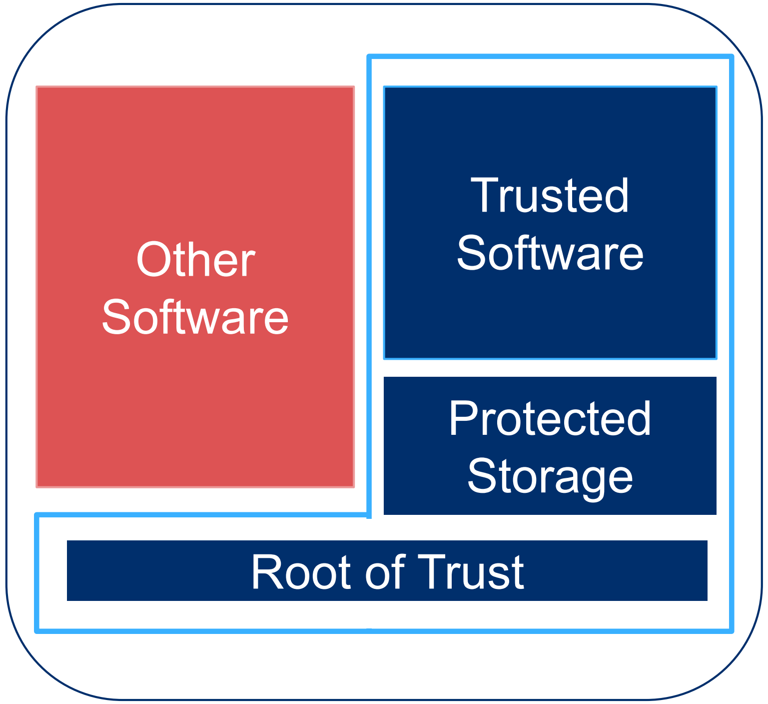
Still ~1000x slower than without privacy

Server can no longer filter requests or do sophisticated metering

Reveals structure (but not params) of NN

Skip to End

Hardware-security mechanisms are pervasive



Hardware support for

- **Isolated execution: Isolated Execution Environment**
- **Protected storage: Sealing**
- **Ability to convince remote verifiers: Remote Attestation**

Trusted Execution Environments (TEEs)

Operating in parallel with "rich execution environments" (REEs)

Cryptocards

https://www.ibm.com/security/cryptocards/

Trusted Platform Modules

https://www.infineon.com/tpm

ARM TrustZone

arm

https://www.arm.com/products/security-on-arm/trustzone

Intel Software Guard Extensions

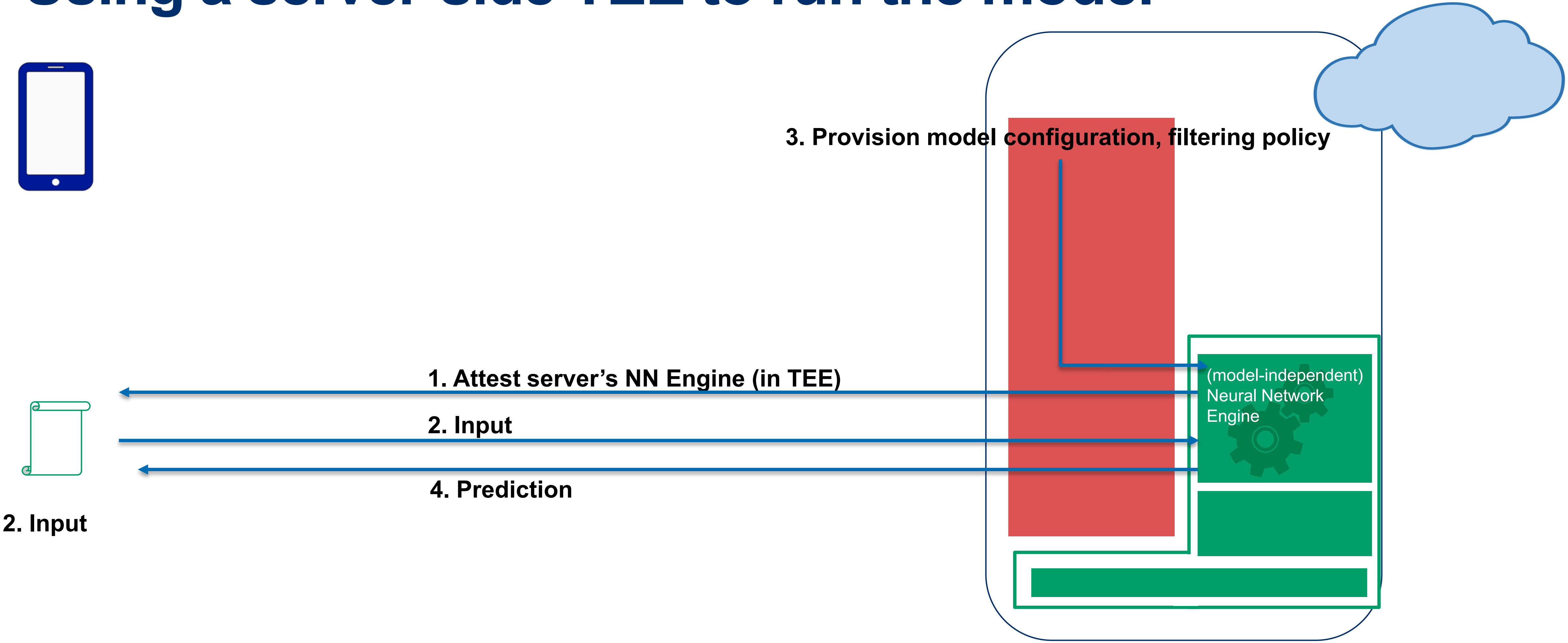
intel

https://software.intel.com/en-us/sgx

[A+14] "Mobile Trusted Computing", Proceedings of the IEEE, 102(8) (2014)
[EKA14] "Untapped potential of trusted execution environments", IEEE S&P Magazine, 12:04 (2014)

Skip to End

Using a server-side TEE to run the model



MiniONN + policy filtering + advanced metering + performance + better model secrecy

- harder to reason about client input privacy

MiniONN: Efficiently transform any given neural network into oblivious form with no/negligible accuracy loss

Try at: https://github.com/SSGAalto/minionn

[LJLA17] MiniONN, ACM CCS 2017,
[KNLAS19] Private Decision Trees, PETS 2019

Trusted Computing can help realize improved security and privacy for ML

ML is very fragile in adversarial settings

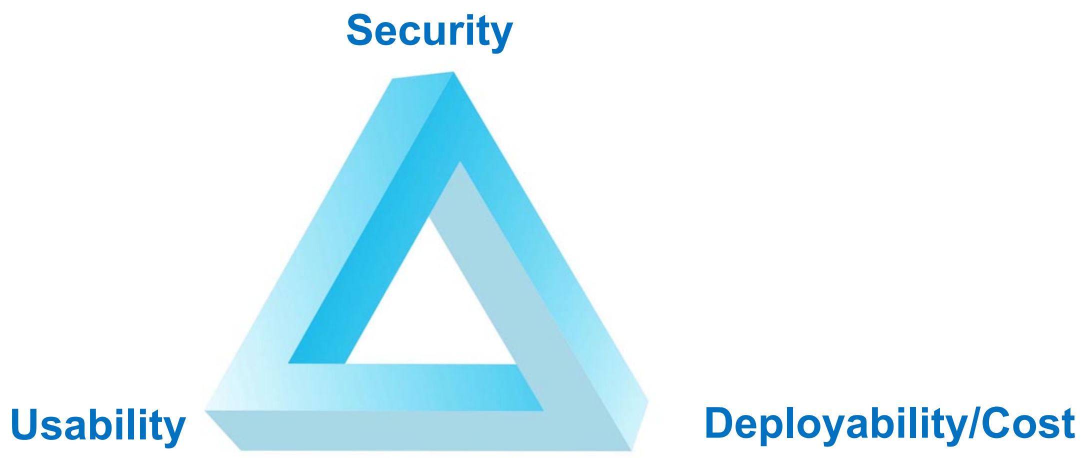
https://asokan.org/asokan/
@nasokan

Securing cloud-assisted services

Cloud-assisted services raise new security/privacy concerns

- But naïve solutions may conflict with privacy, usability, deployability, …

Solutions using Trusted hardware + cryptography

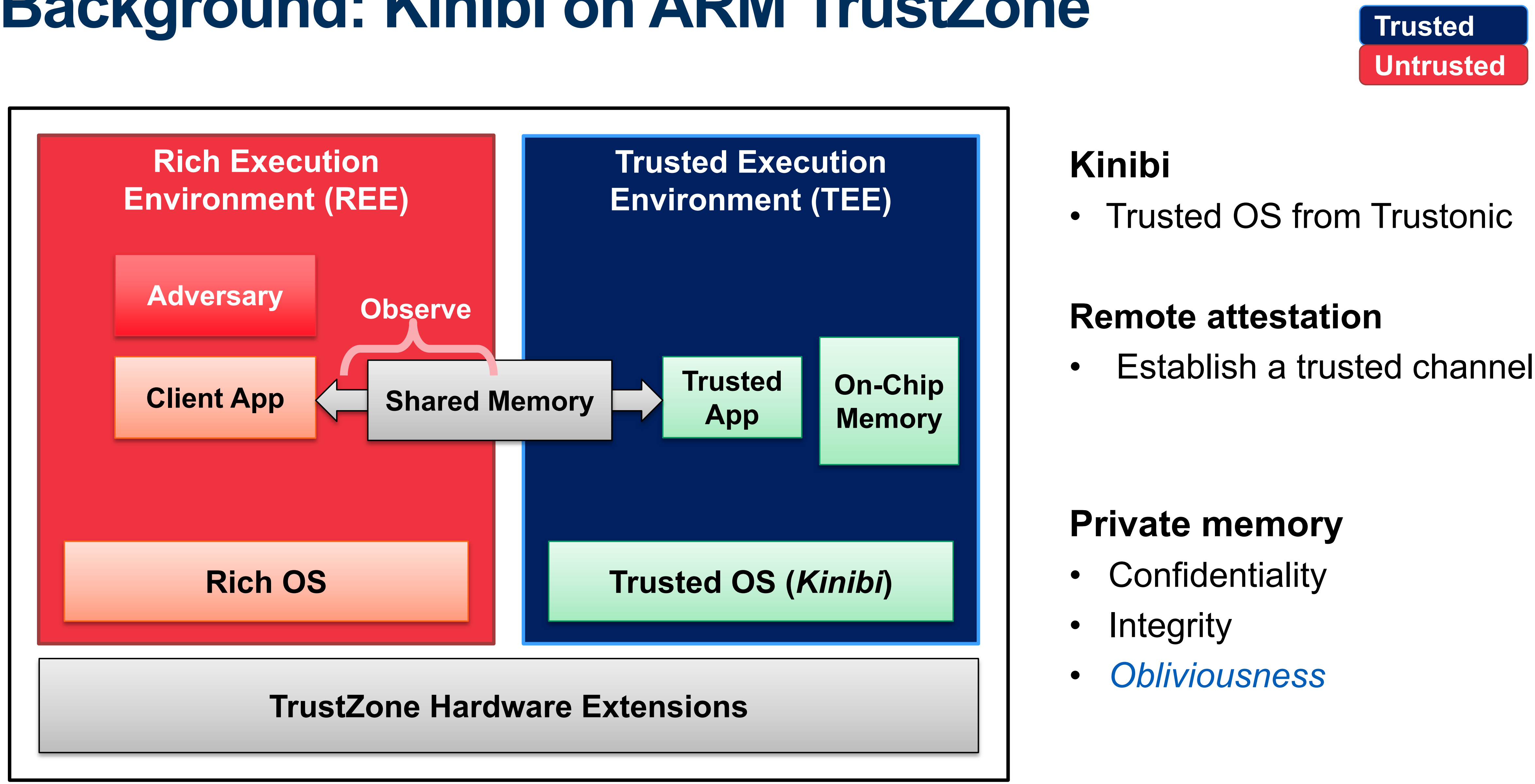


https://asokan.org/asokan/
@nasokan

[TLPEPA17] Circle Game, ACM ASIACCS 2017
[LJLA17] MiniONN, ACM CCS 2017, [KNLAS19] Private Decision Trees, PETS 2019

Supplementary material

Background: Kinibi on ARM TrustZone



Kinibi

- Trusted OS from Trustonic

Remote attestation

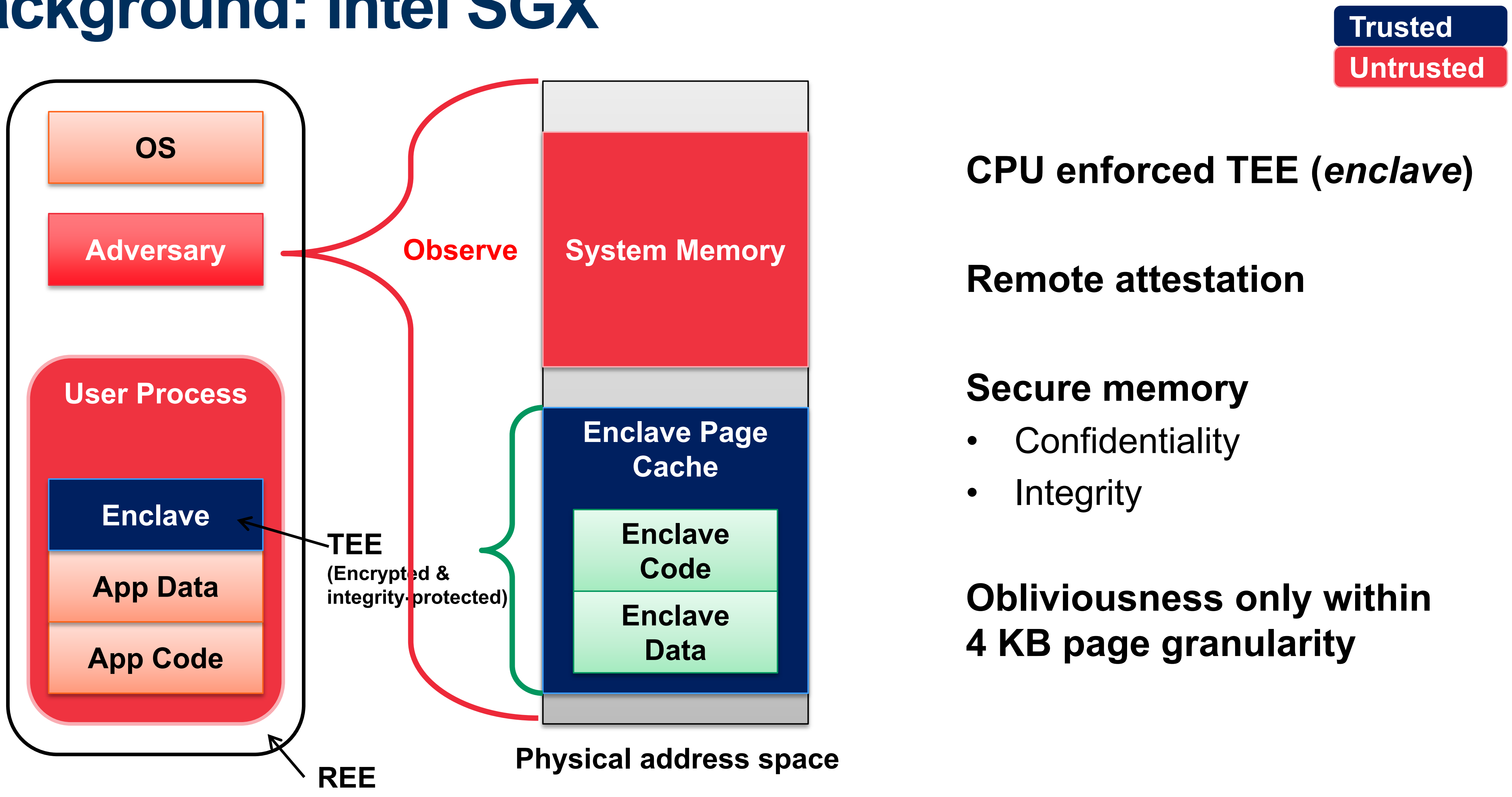
- Establish a trusted channel

Private memory

- Confidentiality
- Integrity
- *Obliviousness*

https://www.trustonic.com/solutions/trustonic-secured-platforms-tsp/

Background: Intel SGX



CPU enforced TEE (*enclave*)

Remote attestation

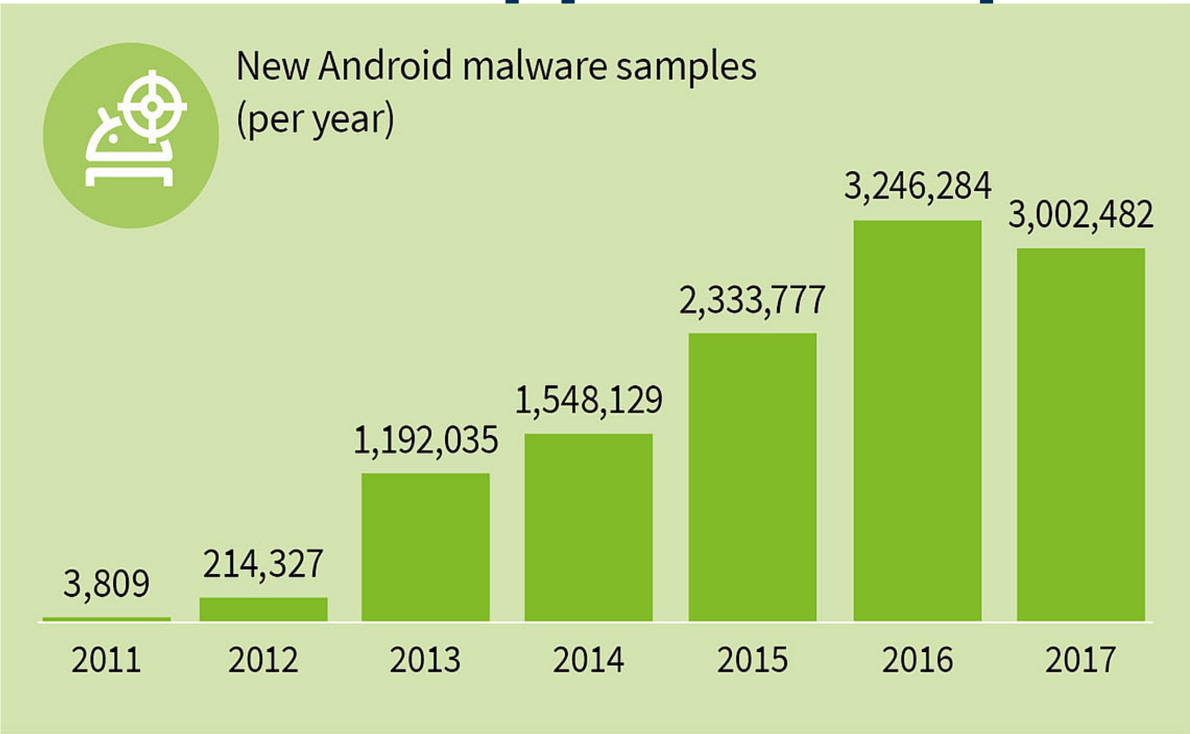
Secure memory

- Confidentiality
- Integrity

Obliviousness only within 4 KB page granularity

https://software.intel.com/sgx

Android app landscape



Unique new Android malware samples
Source: G Data
2015: https://secure.gd/dl-en-mmwr201504
2018: https://www.gdatasoftware.com/blog/2018/02/30491-some-343-new-android-malware-samples-every-hour-in-2017

On average a user installs 95 apps (Yahoo Aviate)
Yahoo Aviate study
Source:
https://yahooaviate.tumblr.com/image/95795838933

Current dictionary size < 2^{24} entries

Even comparatively “high” FPR (e.g., ~2^{-10}) may have negligible impact on privacy

Cloud-scale PMT

Verify Apps: cloud-based service to check for harmful Android apps prior to installation

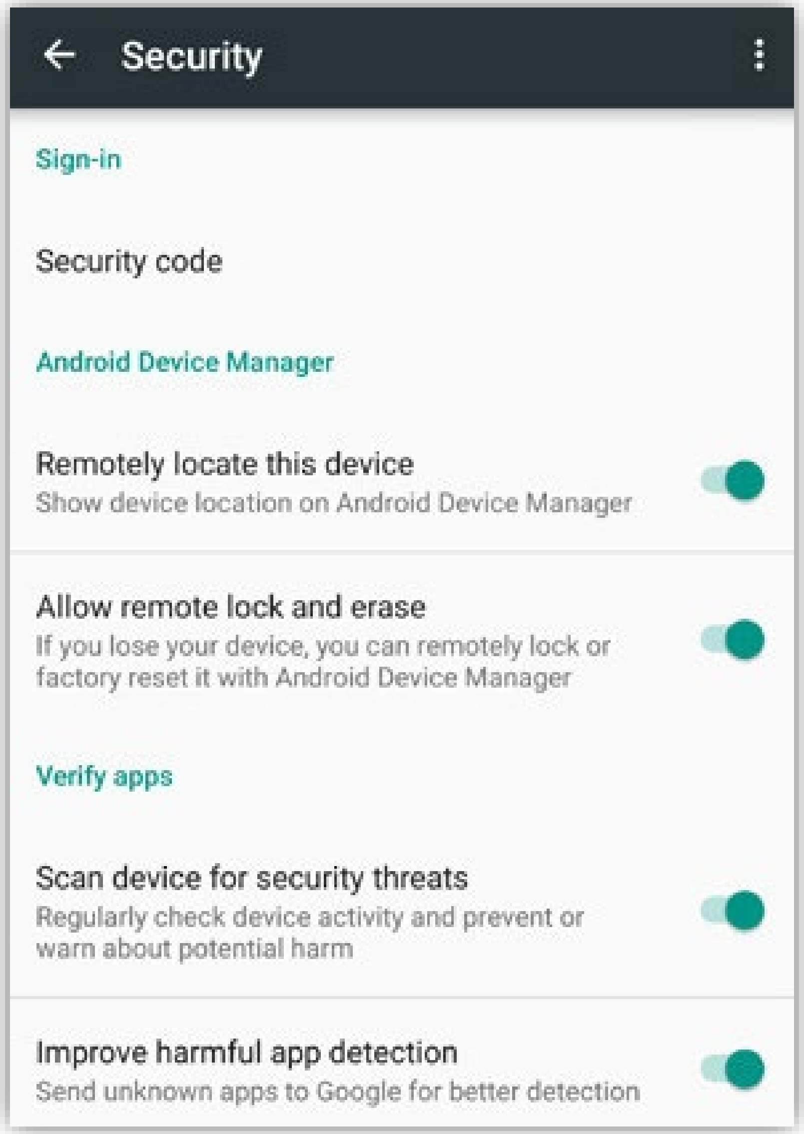
"… over 1 billion devices protected by Google's security services, and over 400 million device security scans were conducted per day"

Android Security 2015 Year in Review

"2 billion+ Android devices checked per day"

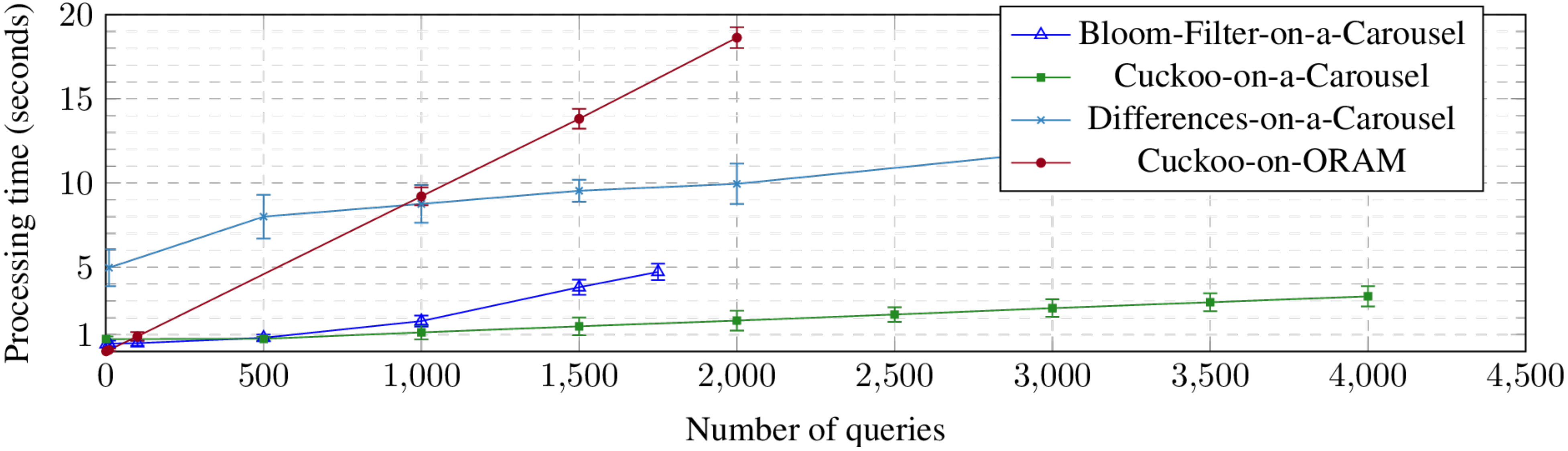
https://www.android.com/security-center/

(c.f. < 17 million malware samples)



Performance: batch queries

Kinibi on ARM TrustZone



Intel SGX