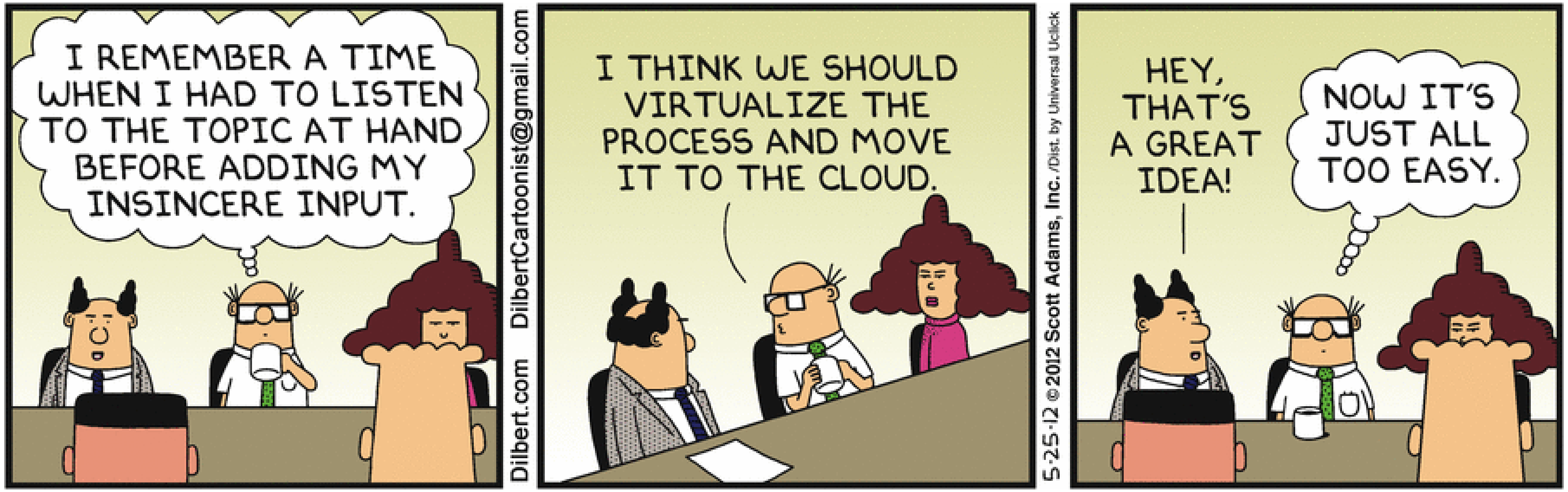
Aalto University

Securing Cloud-assisted Services

N. Asokan

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

Services are moving to “the cloud”



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

Services are moving to “the cloud”

Example: cloud-based malware scanning service

Example: cloud storage

…

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

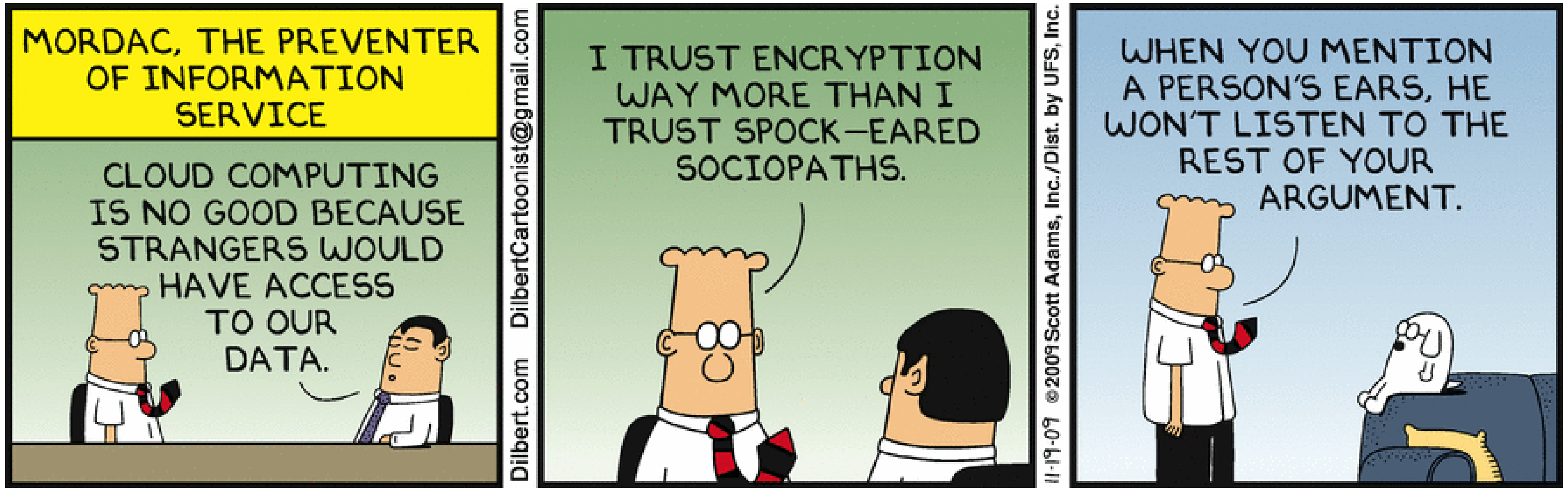
http://www.aaai.org/ocs/index.php/ICWSM/ICWSM16/paper/view/13047

Securing cloud storage

Client-side encryption of user data is desirable

But naïve client-side encryption conflicts with

- Storage provider's business requirement: deduplication ([LPA15] ACM CCS '15)
- End user's usability requirement: multi-device access ([P+17] IEEE IC '17, CeBIT '16)



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

New privacy and security concerns arise

Example: cloud-based malware scanning service

Example: cloud storage

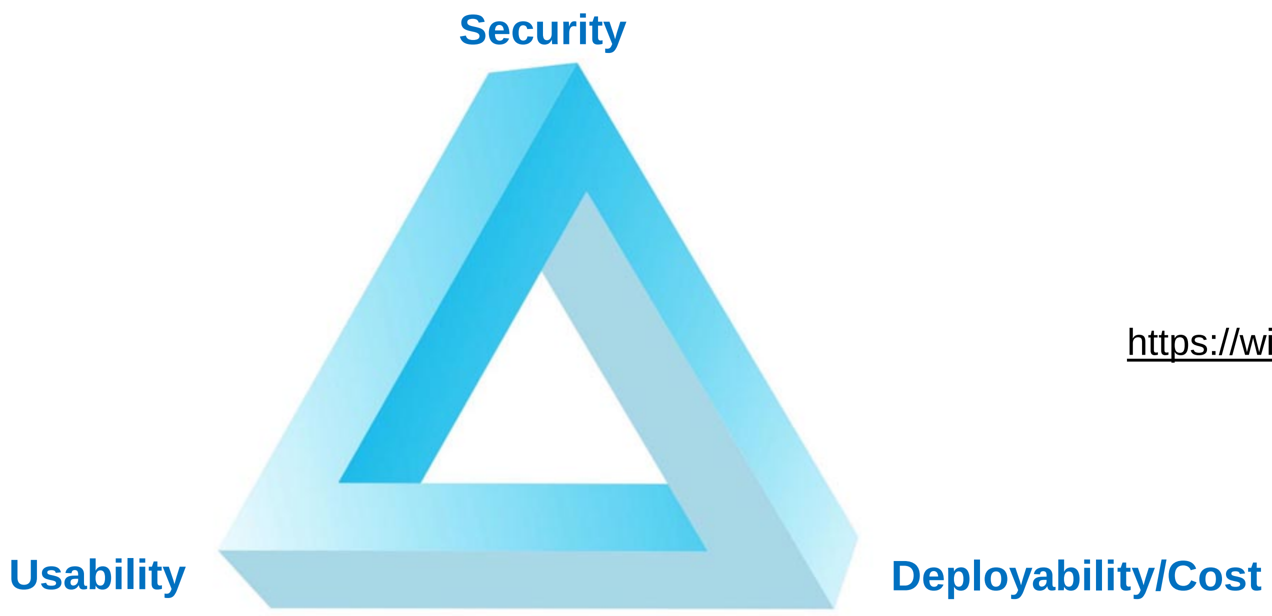
Naïve solutions conflict with other requirements

- privacy, usability, deployability

CloSer project: the big picture

Cloud Security Services

- 2014-2016, funded by Academy of Finland
- 2016-2018, funded by Tekes
- Academics collaborating with Industry



https://wiki.aalto.fi/display/CloSeProject/CloSer+Project+Public+Homepage

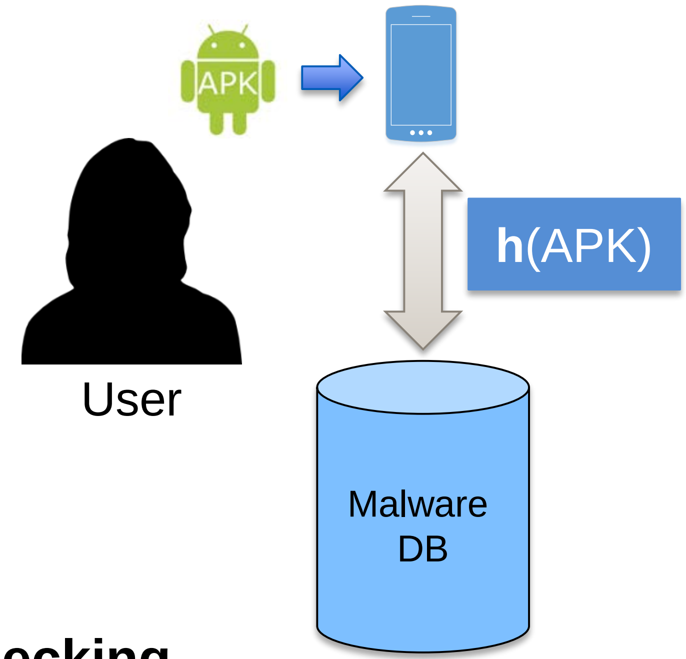
The Circle Game: Scalable Private Membership Test Using Trusted Hardware

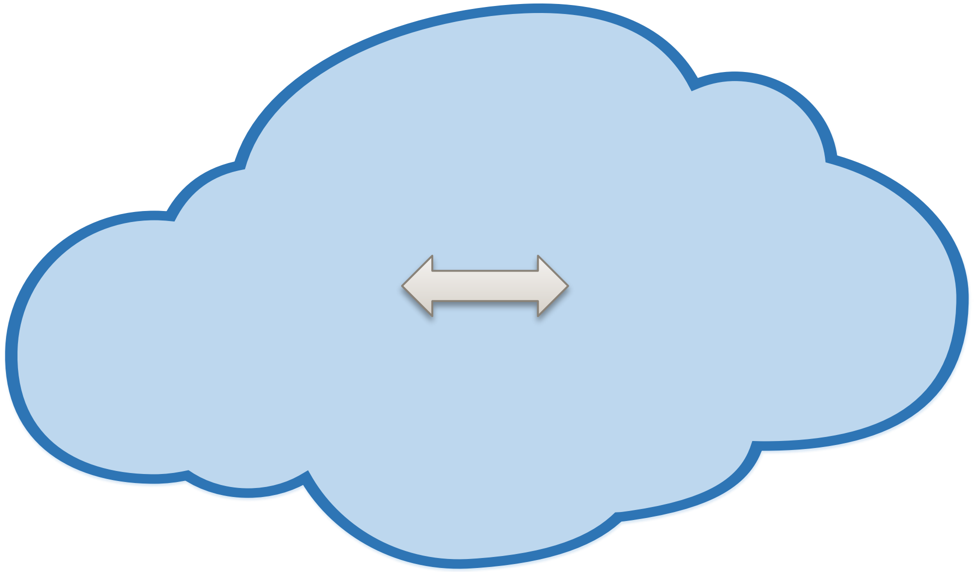
Sandeep Tamrakar [1] *Jian Liu* [1] *Andrew Paverd* [1]

Jan-Erik Ekberg [2] *Benny Pinkas* [3] *N. Asokan* [1]

1. *Aalto University, Finland* 2. *Huawei (work done while at Trustonic)* 3. *Bar-Ilan University, Israel*

Malware checking



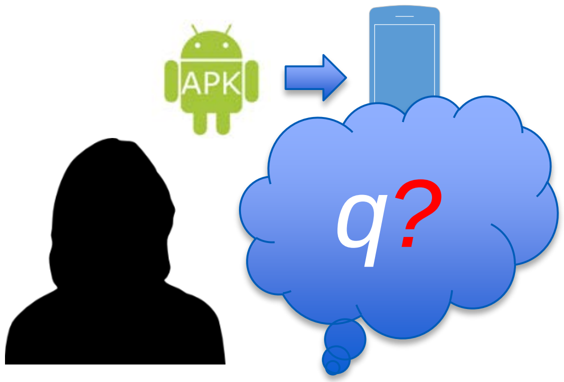


On-device checking
- High communication and computation costs
- Database changes frequently
- Database is revealed to everyone

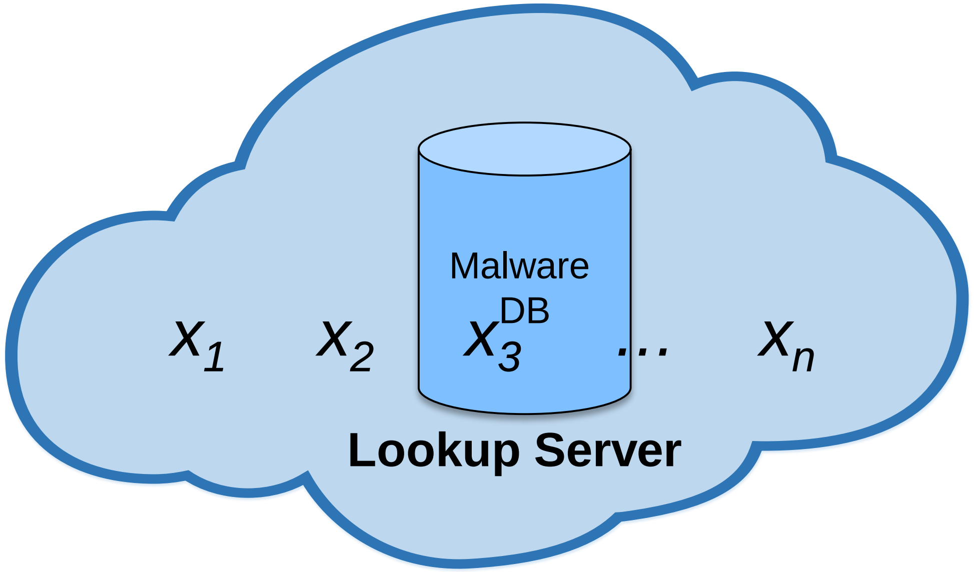
Cloud-based checking
- Minimal communication and computation costs
- Database can change frequently
- Database is not revealed to everyone
- User privacy at risk!

Private Membership Test (PMT)

The problem: How to preserve end user privacy when querying cloud-hosted databases?



User



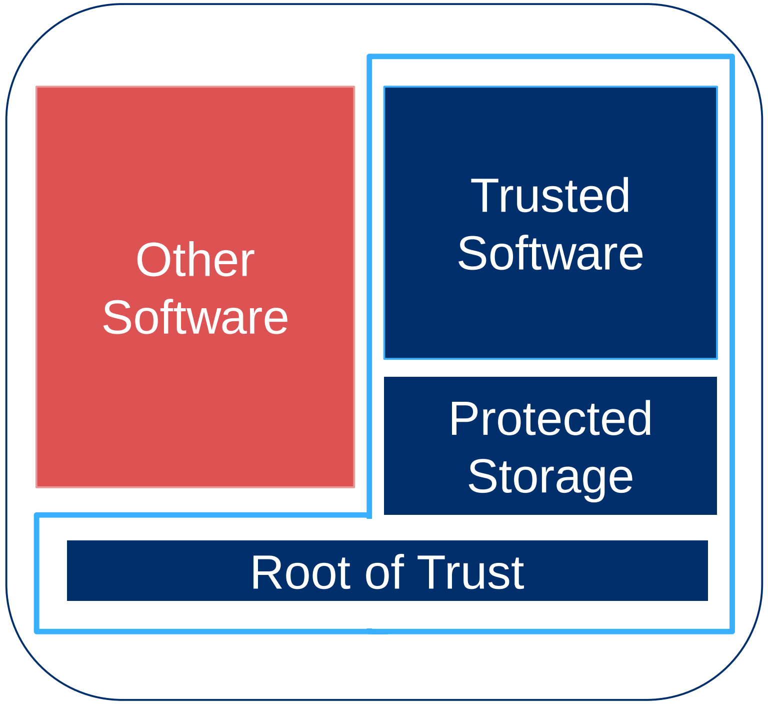
Server must not learn contents of client query (q).

Current solutions (e.g. private set intersection, private information retrieval):

- Single server: expensive in both computation and/or communication
- Multiple independent servers: unrealistic in commercial setting

Can *hardware-assisted trusted execution environments* provide a practical solution?

Trusted Execution Environments are pervasive



Hardware support for

- **Isolated execution: Trusted Execution Environment**
- **Protected storage: Sealing**
- **Ability to report status to a remote verifier: Remote Attestation**

Cryptocards

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

Trusted Platform Modules

https://www.infineon.com/tpm

ARM TrustZone

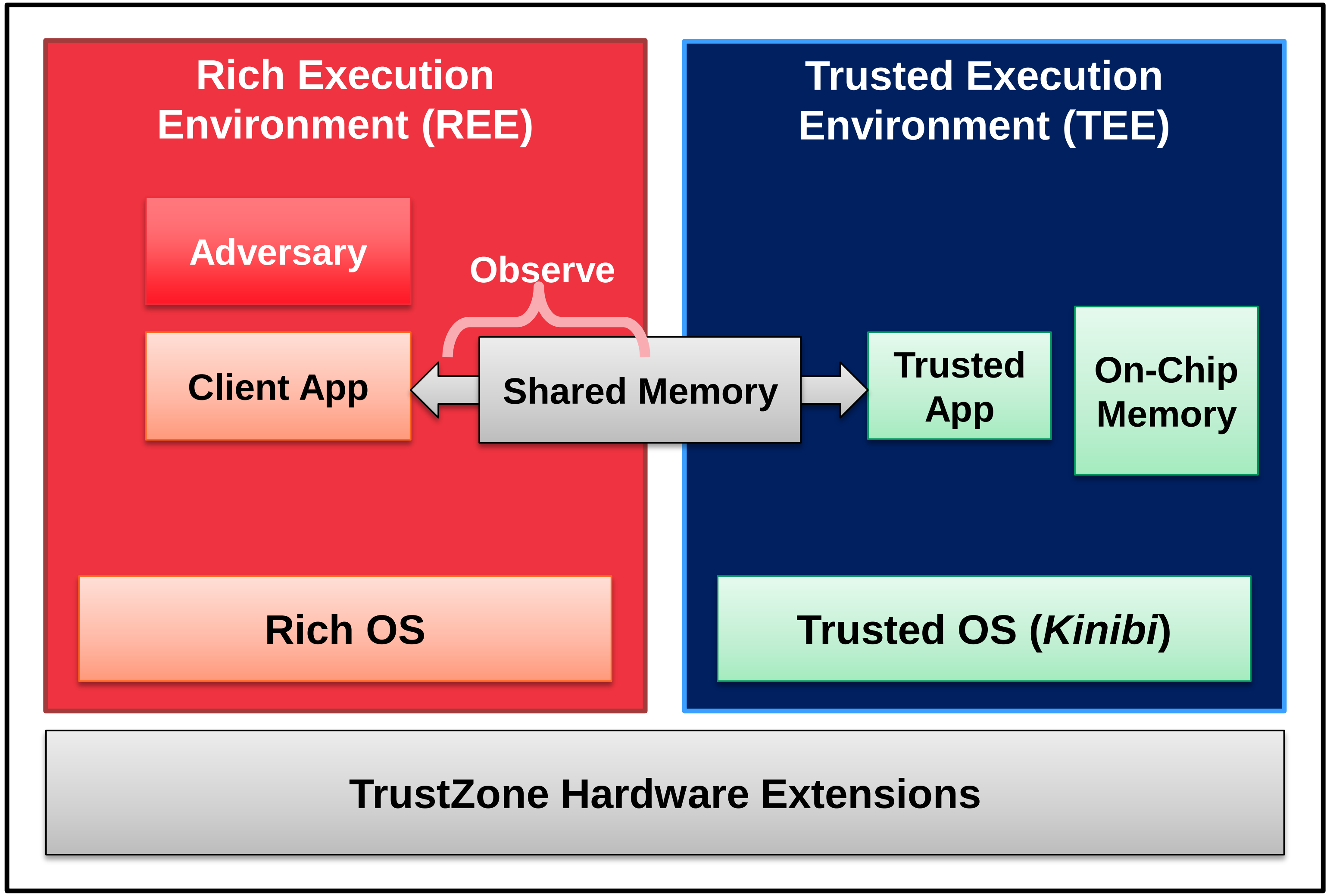
https://www.arm.com/products/security-on-arm/trustzone

Intel Software Guard Extensions

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

[EKA14] "*Untapped potential of trusted execution environments*", IEEE S&P Magazine, 12:04 (2014)

Background: Kinibi on ARM TrustZone



Kinibi

- Trusted OS from Trustonic

Remote attestation

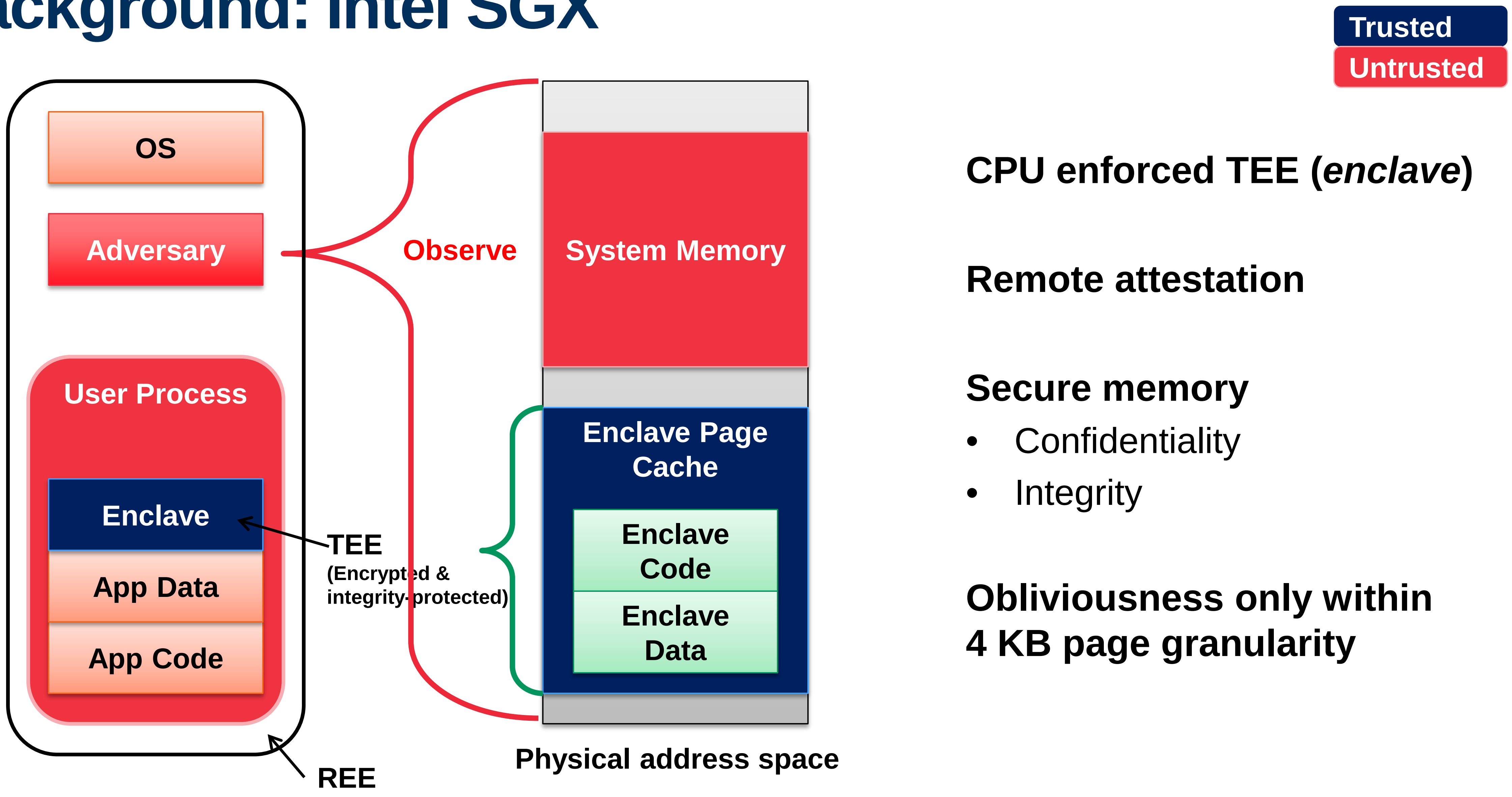
- Establish a trusted channel

Private memory

- Confidentiality
- Integrity
- *Obliviousness*

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

Background: Intel SGX



CPU enforced TEE (*enclave*)

Remote attestation

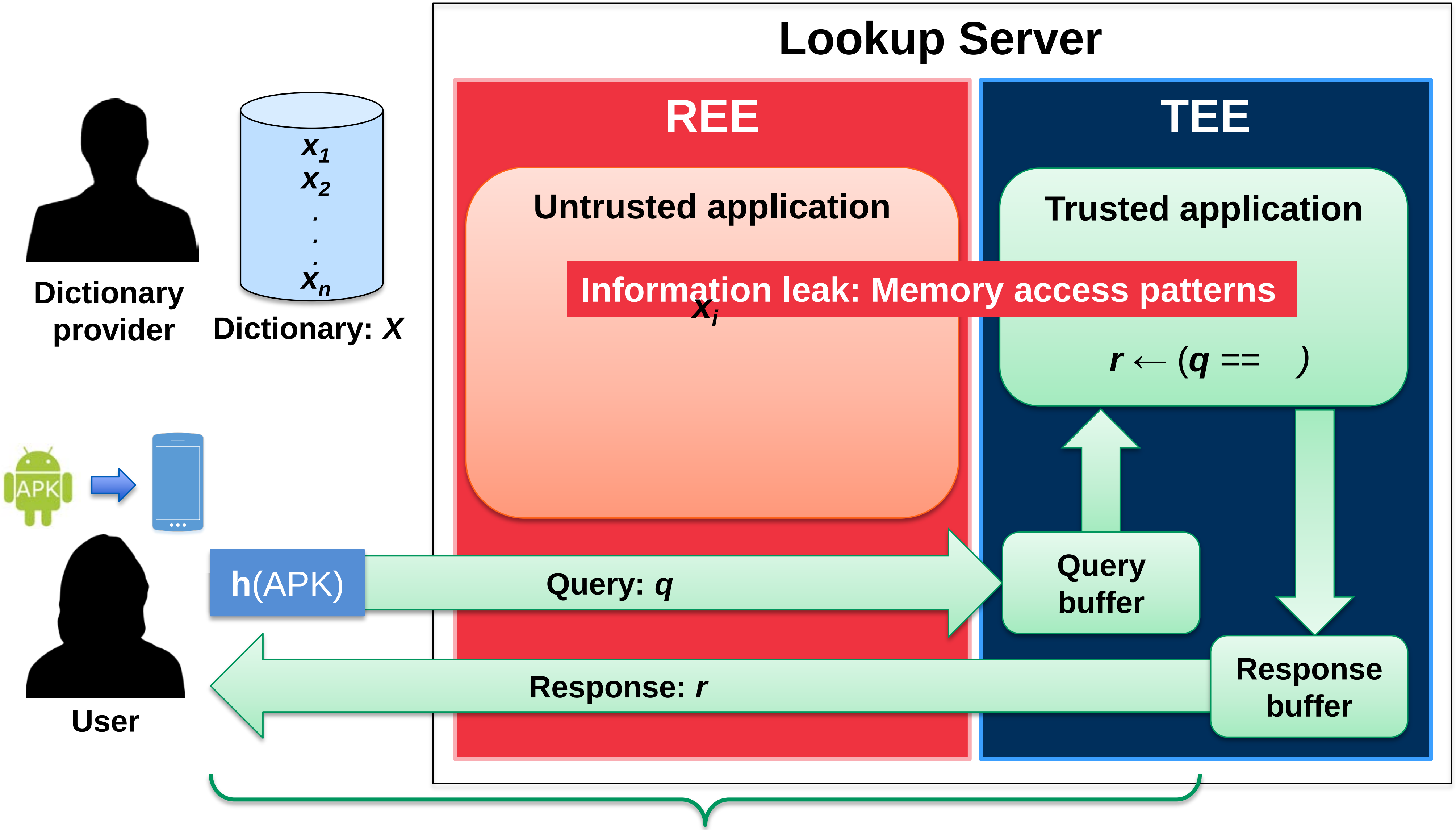
Secure memory

- Confidentiality
- Integrity

Obliviousness only within 4 KB page granularity

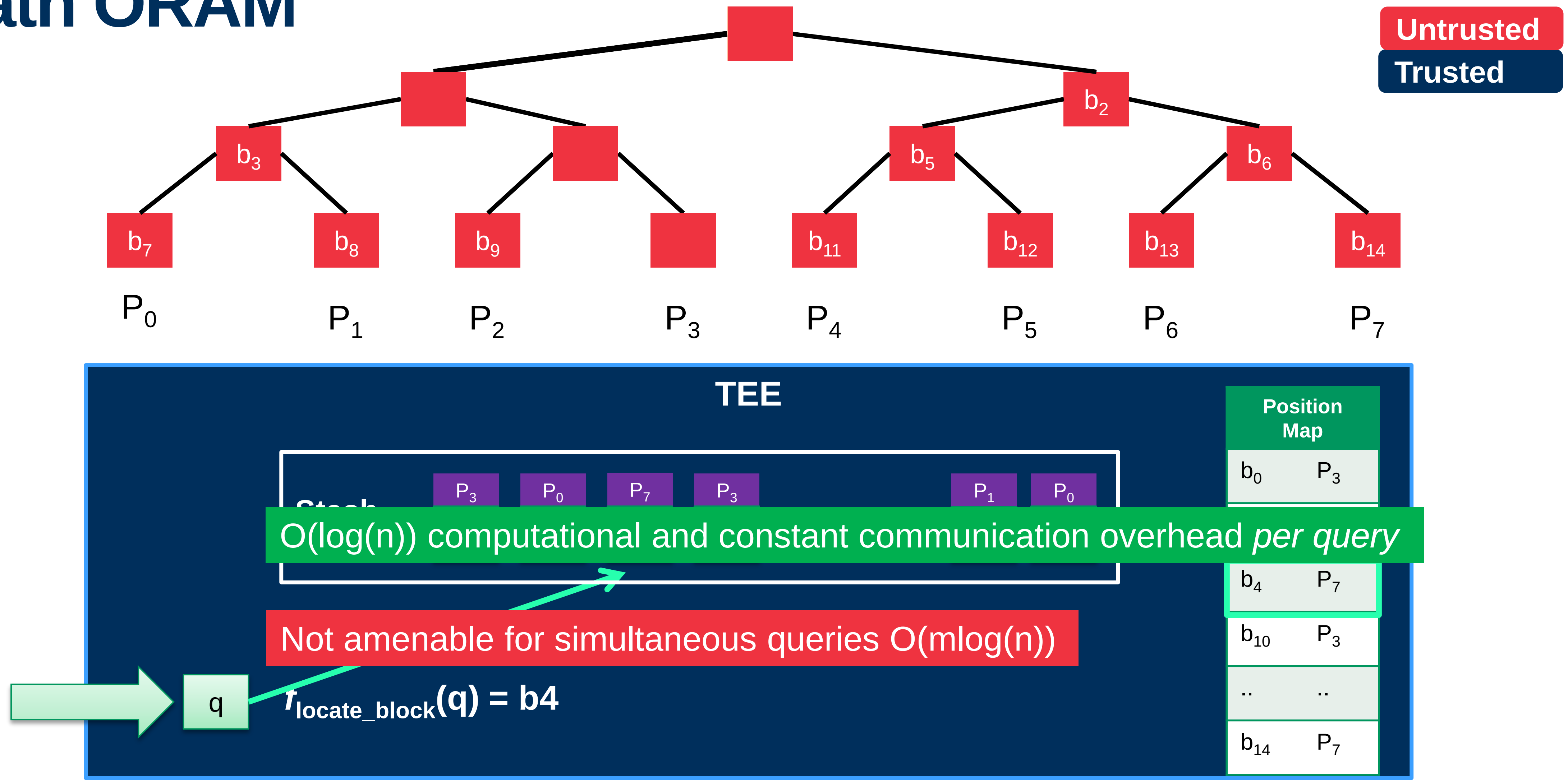
https://software.intel.com/sgx

System model



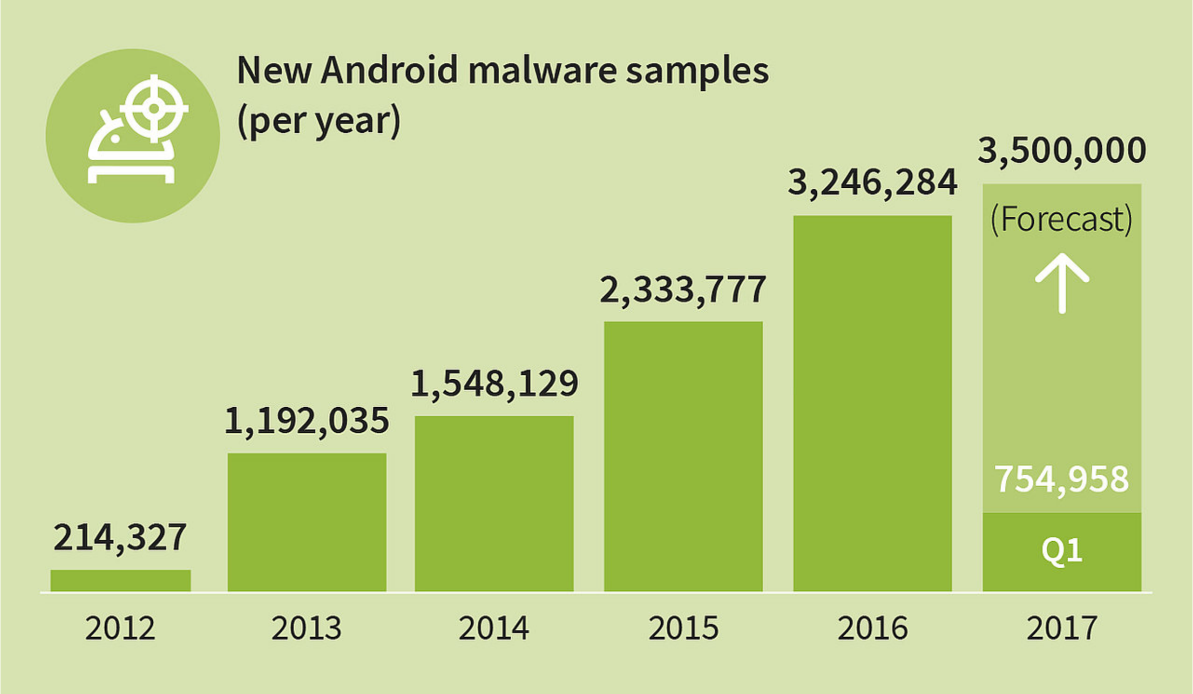
Secure channel with *remote attestation*

Path ORAM



Stefanov et al. ACM CCS 2013, https://dl.acm.org/citation.cfm?id=2516660

Android app landscape



Unique new Android malware samples
Source: G Data https://secure.gd/dl-en-mmwr201504
Source: G Data https://www.gdatasoftware.com/blog/2017/04/29712-8-400-new-android-malware-samples-every-day

Current dictionary size < 2^{24} entries

On average a user installs 95 apps (Yahoo Aviate)

Yahoo Aviate study
Source: https://yahooaviate.tumblr.com/image/95795838933

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

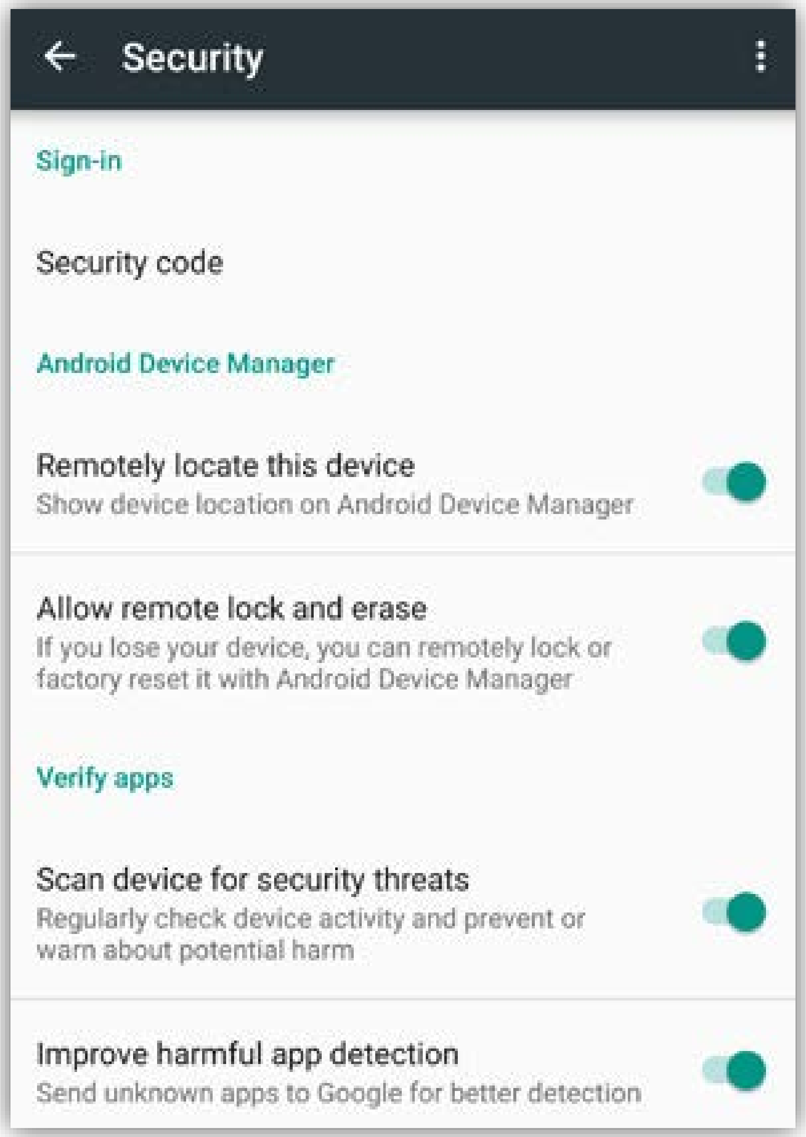
Cloud-scale PMT

***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"

<u>Android Security 2015 Year in Review</u>

(c.f. < 13 million malware samples)



Requirements

Query Privacy: Adversary cannot learn/infer query or response content

- User can always choose to reveal query content

Accuracy: No false negatives

- However, some false positives are tolerable (i.e. non-zero false positive rate)

Response Latency: Respond quickly to each query

Server Scalability: Maximize overall throughput (queries per second)

Requirements revisited

Query Privacy: Adversary cannot learn/infer query or response content

- User can always choose to reveal queries

Accuracy: No false negatives

- However, some false positives are tolerable (i.e. non-zero false positive rate)

FPR* = 2^{-10}

Response Latency: Respond quickly to each query

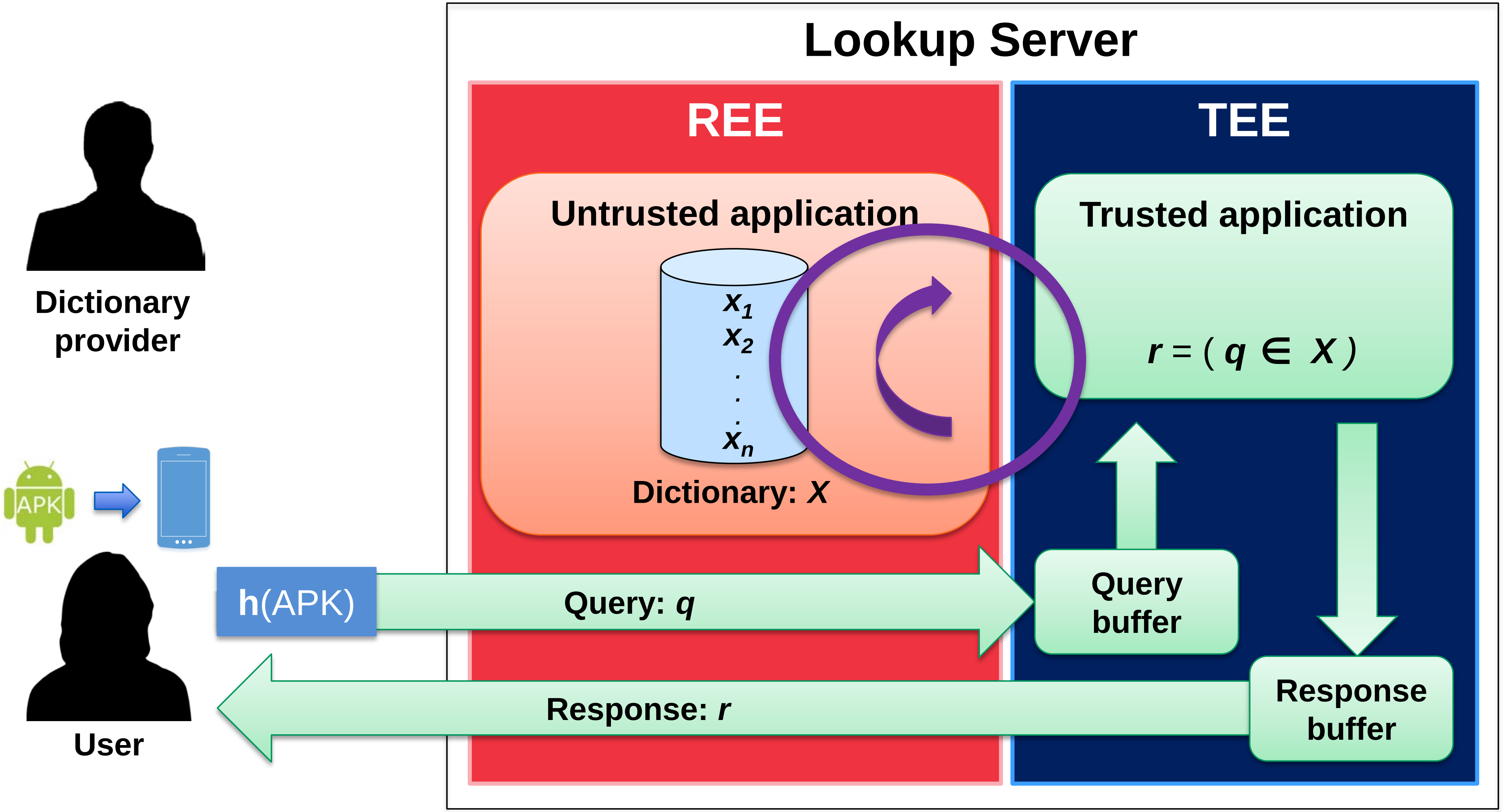
Latency* ~ 1s

Server Scalability: Maximize overall throughput (queries per second)

Dictionary size* = 2^{26} entries (~ 67 million entries)

**** parameters suggested by a major anti-malware vendor***

Carousel design pattern



Carousel caveats

1. **Adversary can measure dictionary processing time**
   - Spend equal time processing each dictionary entry

2. **Adversary can measure query-response time**
   - Only respond after one full carousel cycle

Both impact response latency (recall Requirements)

Therefore, aim to *minimize carousel cycle time*

How to minimize carousel cycle time?

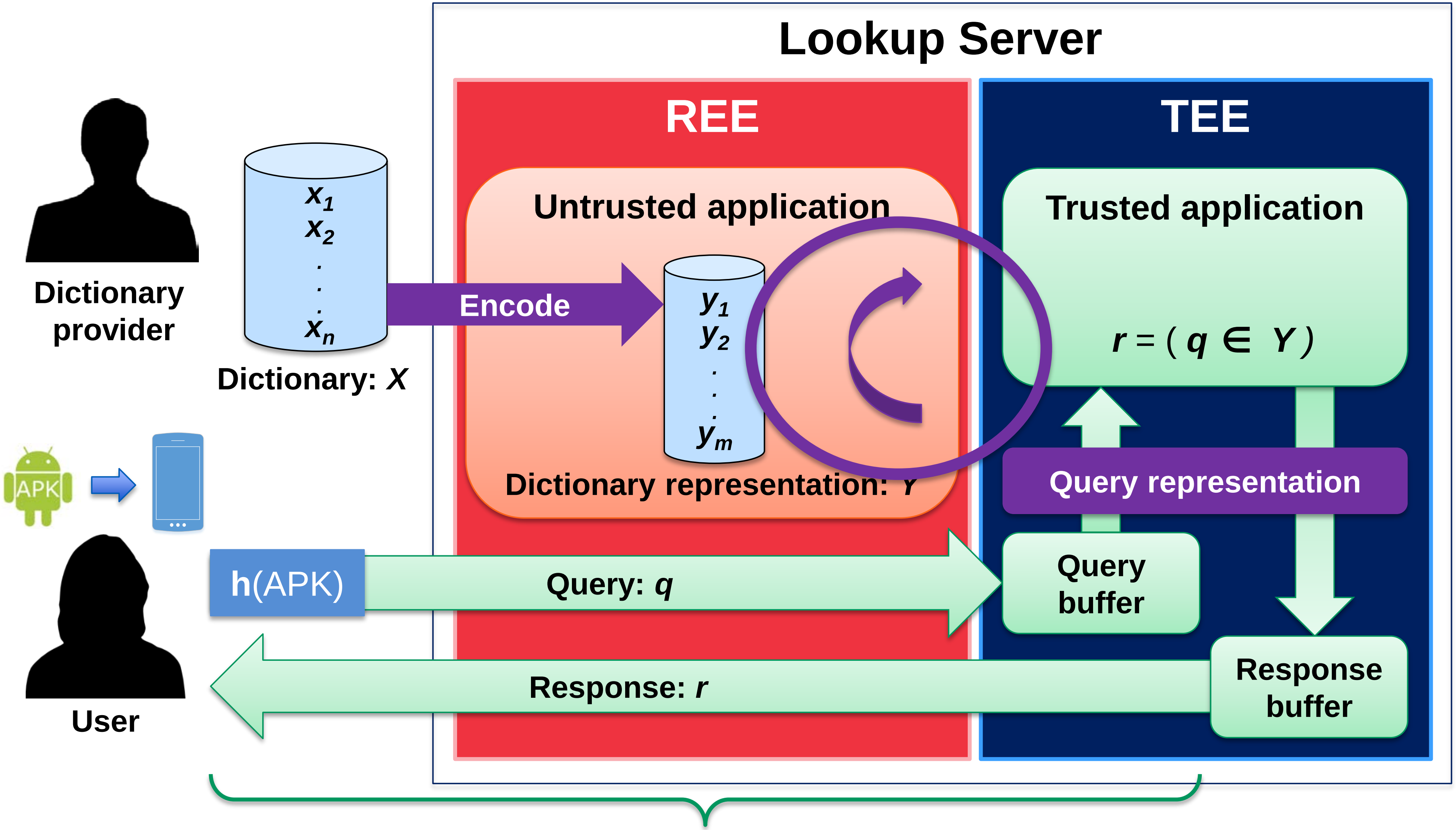
Represent dictionary using efficient data structure

Various existing data structures support membership test:

- Bloom Filter
- Cuckoo hash

Experimental evaluation required for carousel approach

Carousel design pattern



Secure channel with *remote attestation*

Experimental evaluation

Kinibi on ARM TrustZone

- Samsung Exynos 5250 (Arndale)
- 1.7 GHz dual-core ARM Cortex-A17
- Android 4.2.1
- ARM GCC compiler and Kinibi libraries
- Maximum TA private memory: 1 MB
- Maximum shared memory: 1 MB

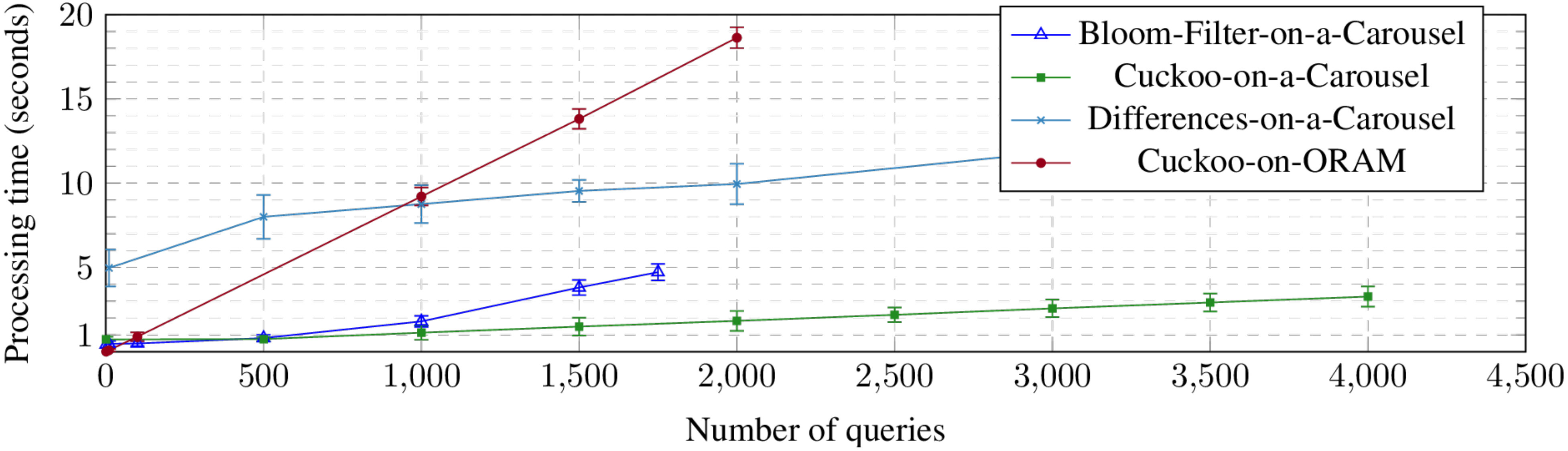
Intel SGX

- HP EliteDesk 800 G2 desktop
- 3.2 GHz Intel Core i5 6500 CPU
- 8 GB RAM
- Windows 7 (64 bit), 4 KB page size
- Microsoft C/C++ compiler
- Intel SGX SDK for Windows

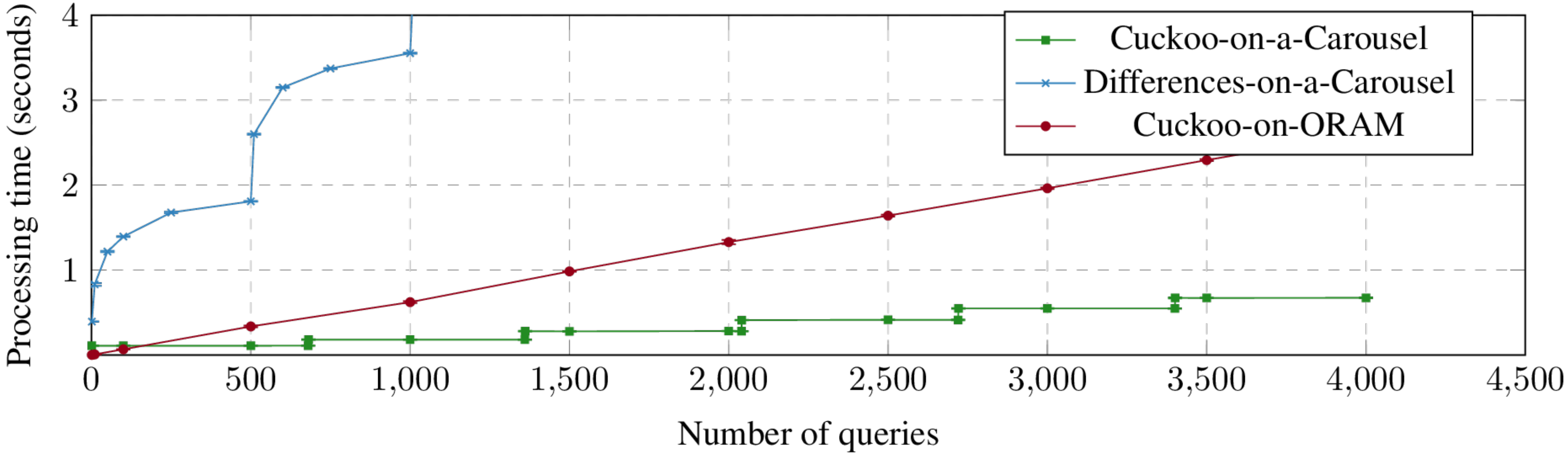
Note: Different CPU speeds and architectures

Performance: batch queries

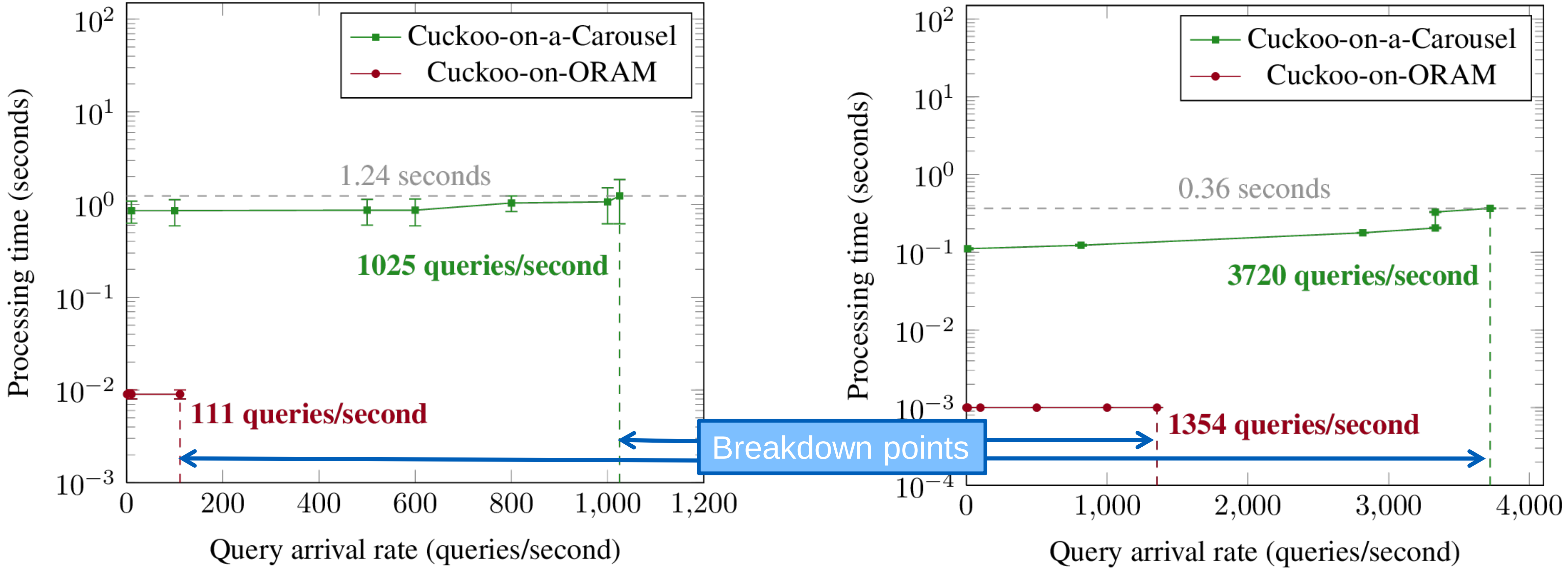
Kinibi on ARM TrustZone



Intel SGX



Performance: steady state



Kinibi on ARM TrustZone **Intel SGX**

Beyond ***breakdown point*** query response latency increases over time

Other applications of PMT

Private contact discovery in messaging apps

Discovery of leaked passwords

…

[KLSAP17] PETS 2017

Signal private contact discovery, Sep 2017

This is much faster. The above code still iterates across the entire set of registered users, but it only does so once for the entire collection of submitted client contacts. By keeping one big linear scan over the registered user data set, access to unencrypted RAM remains "oblivious," since the OS will simply see the enclave touch every item once for each contact discovery request.

The full linear scan is fairly high latency, but by batching many pending client requests together, it can be high throughput.

https://signal.org/blog/private-contact-discovery

The Circle Game: Scalable Private Membership Test Using Trusted Hardware

Sandeep Tamrakar [1] *Jian Liu* [1] *Andrew Paverd* [1]

Jan-Erik Ekberg [2] *Benny Pinkas* [3] *N. Asokan* [1]

1. Aalto University, Finland *2. Darkmatter (work done while at Trustonic)* *3. Bar-Ilan University, Israel*

Aalto University

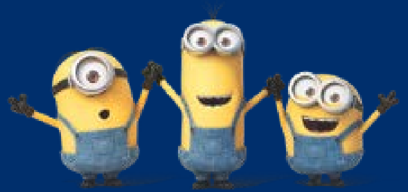
Oblivious Neural Network Predictions via MiniONN Transformations

N. Asokan

http://asokan.org/asokan/

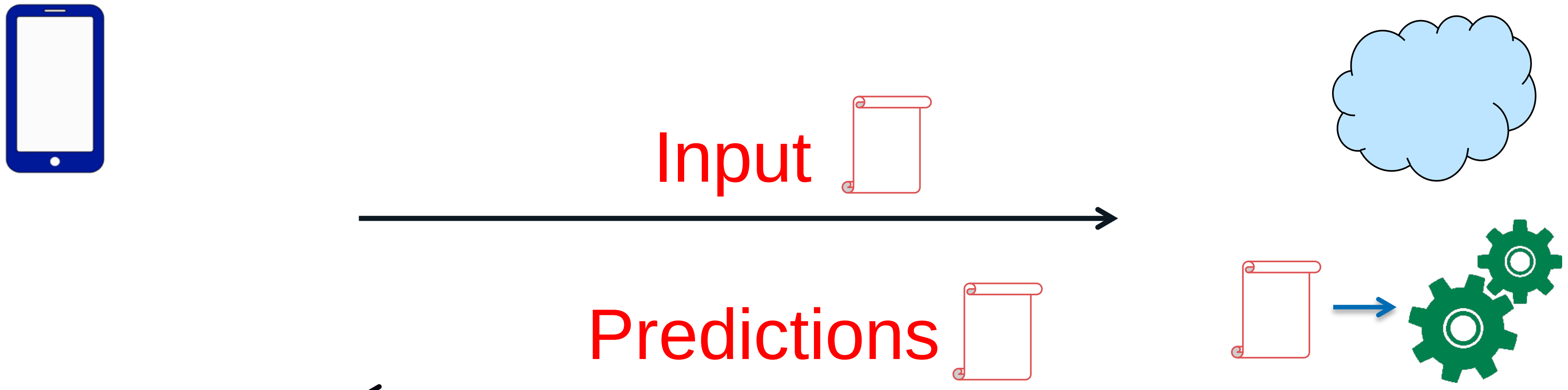
@nasokan

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



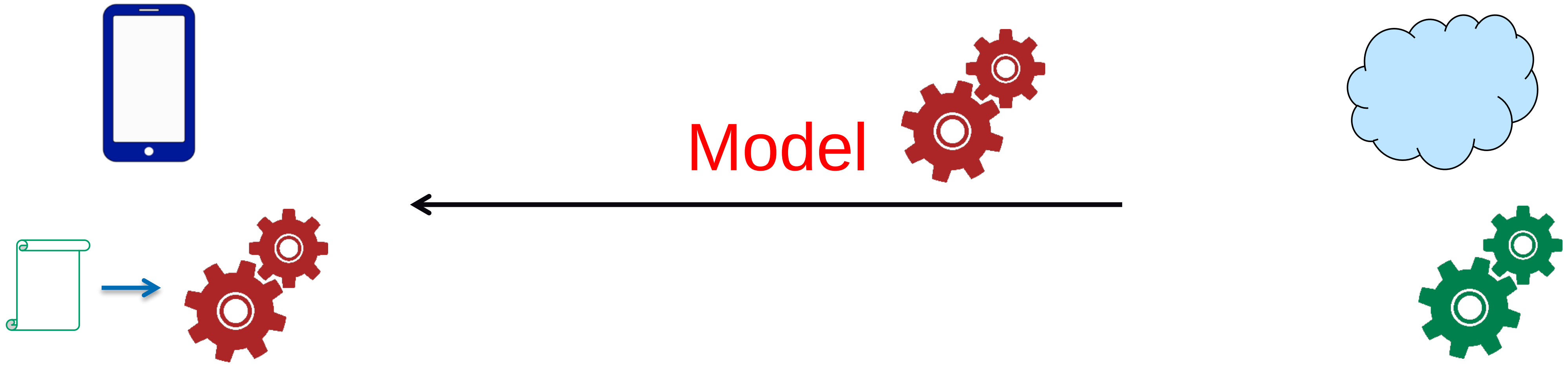
By Source, Fair use, https://en.wikipedia.org/w/index.php?curid=54119040

Machine learning as a service (MLaaS)



violation of clients’ privacy

Running predictions on client-side



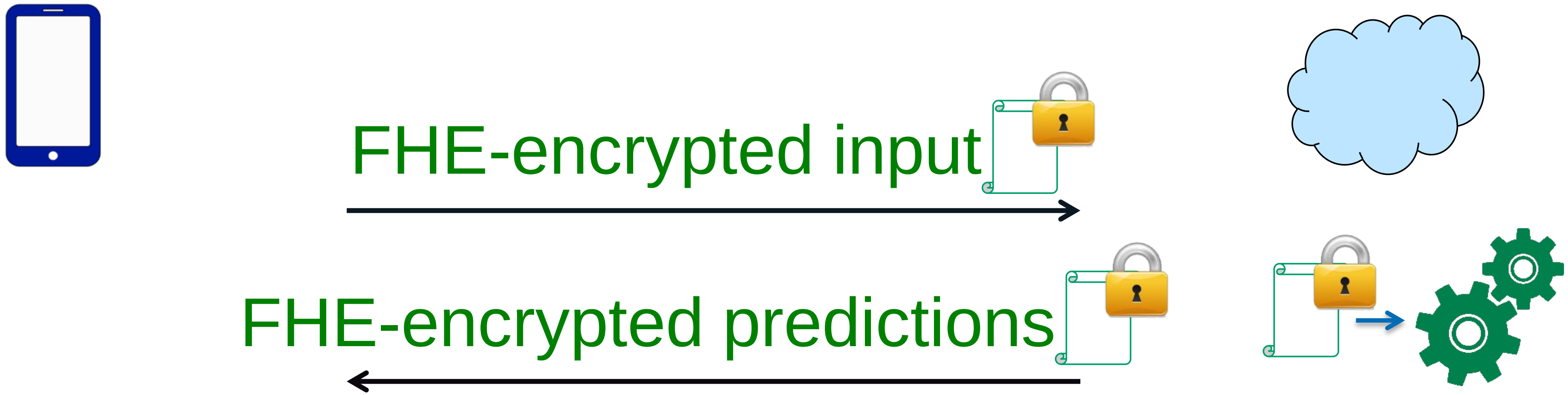
model theft
evasion
model inversion

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

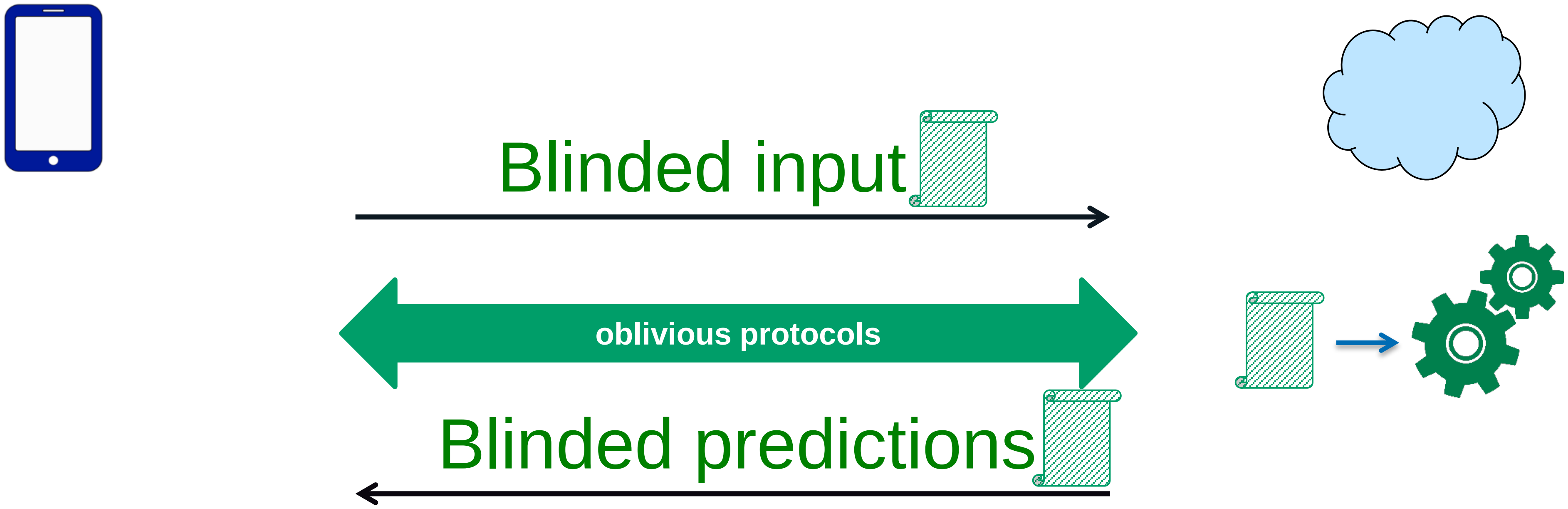


- High throughput for batch queries from same client
- High overhead for single queries: 297.5s and 372MB (MNIST dataset)
- Cannot support: high-degree polynomials, comparisons, …

[GDLLNW16] CryptoNets, ICML 2016

MiniONN: Overview

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

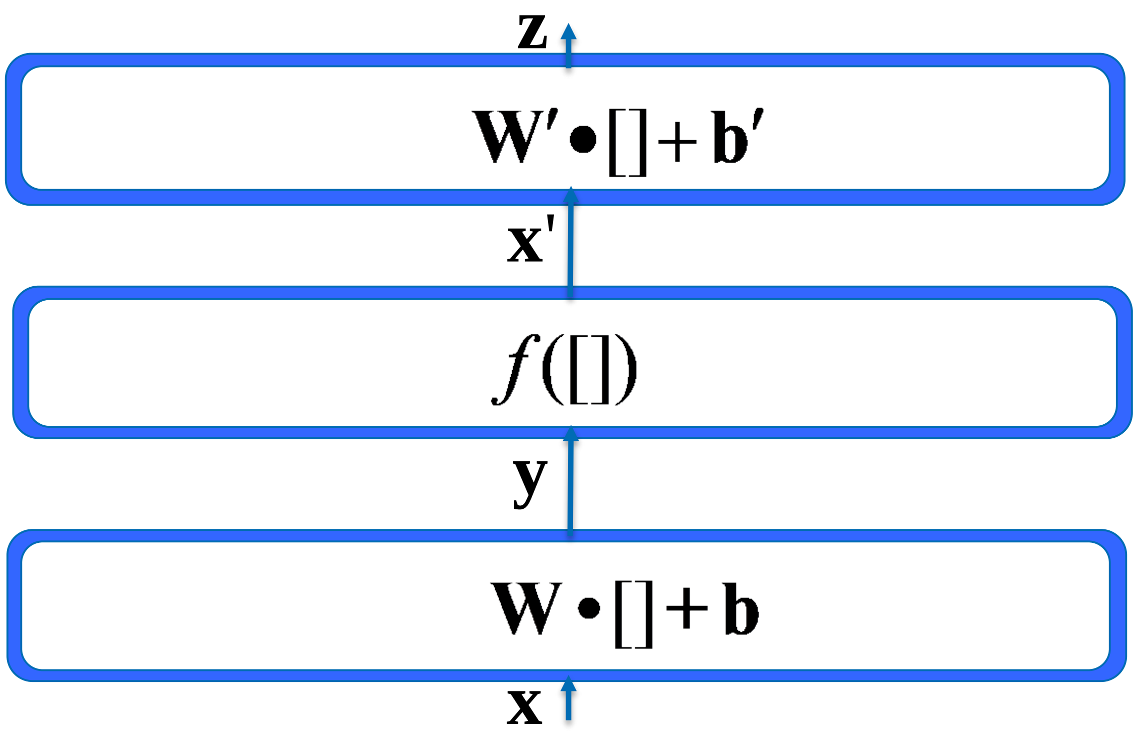


- Low overhead: ~1s
- Support all common neural networks

[Skip to performance](#)

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

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

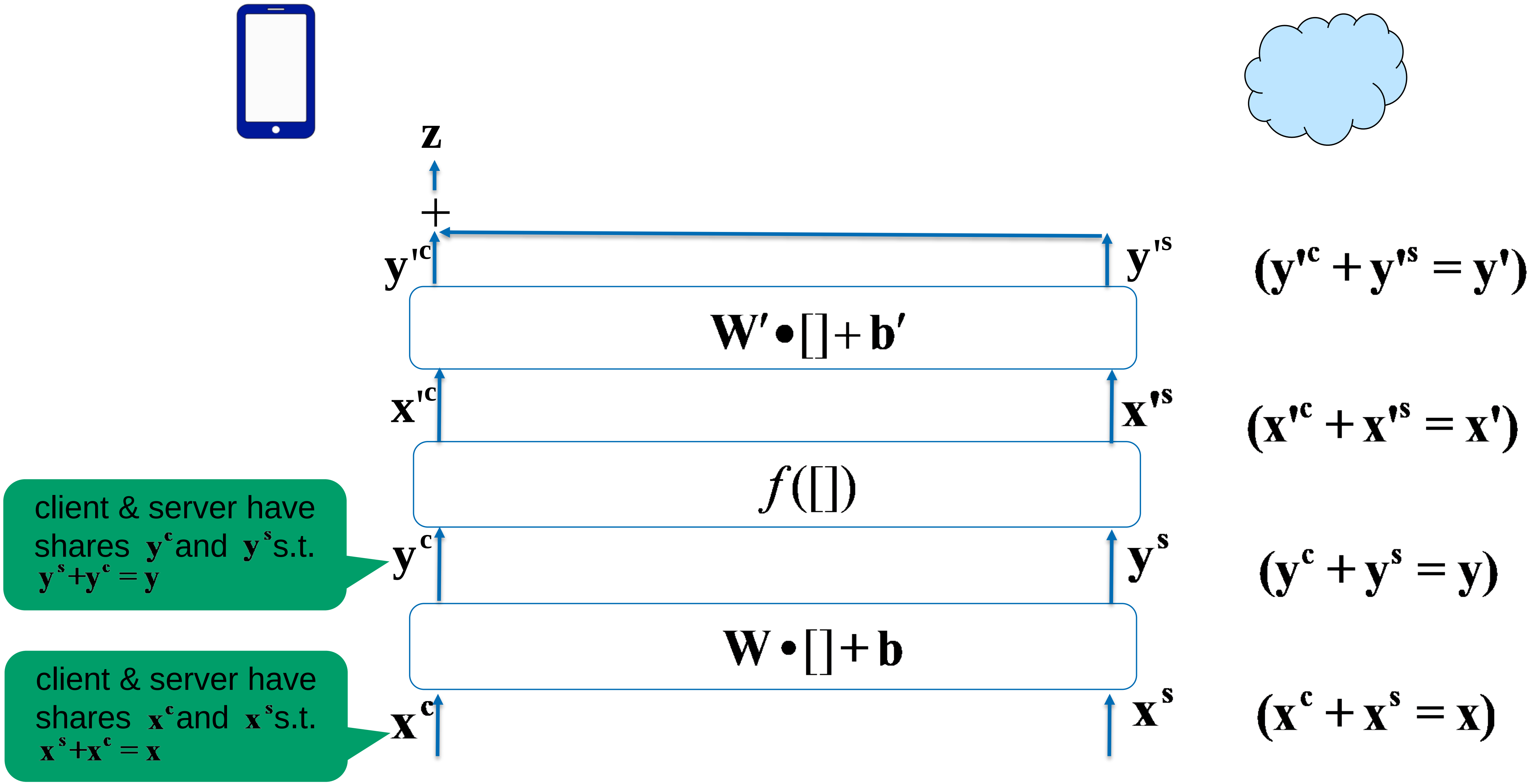


https://eprint.iacr.org/2017/452

All operations are in a finite field Z_N

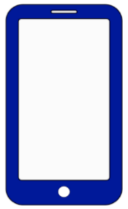
Skip to performance

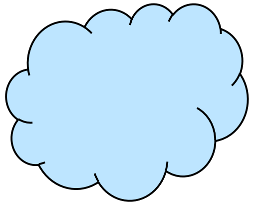
Core idea: use secret sharing for oblivious computation



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

Secret sharing initial input $\mathbf{x}$





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

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

Note that $\mathbf{x}^{\mathbf{c}}$ is independent of $\mathbf{x}$. Can be **pre-chosen**

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

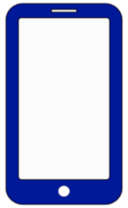
$$= \begin{bmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & w_{2,2} \end{bmatrix} \bullet \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} w_{1,1} & w_{1,2} \\ w_{2,1} & w_{2,2} \end{bmatrix} \bullet \begin{bmatrix} x_1^s + x_1^c \\ x_2^s + x_2^c \end{bmatrix}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

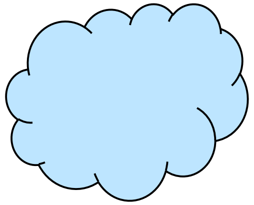
$$= \begin{bmatrix} w_{1,1}(x_1^s + x_1^c) + w_{1,2}(x_2^s + x_2^c) + b_1 \\ w_{2,1}(x_1^s + x_1^c) + w_{2,2}(x_2^s + x_2^c) + b_2 \end{bmatrix} = \begin{bmatrix} \boxed{w_{1,1}x_1^s + w_{1,2}x_2^s + b_1} + \boxed{w_{1,1}x_1^c + w_{1,2}x_2^c} \\ \boxed{w_{2,1}x_1^s + w_{2,2}x_2^s + b_2} + \boxed{w_{2,1}x_1^c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

Compute locally by the server

Dot-product

Oblivious linear transformation: dot-product





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

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

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

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

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

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

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

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

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

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**

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

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

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

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

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

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

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

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

Oblivious activation/pooling functions $f(y)$

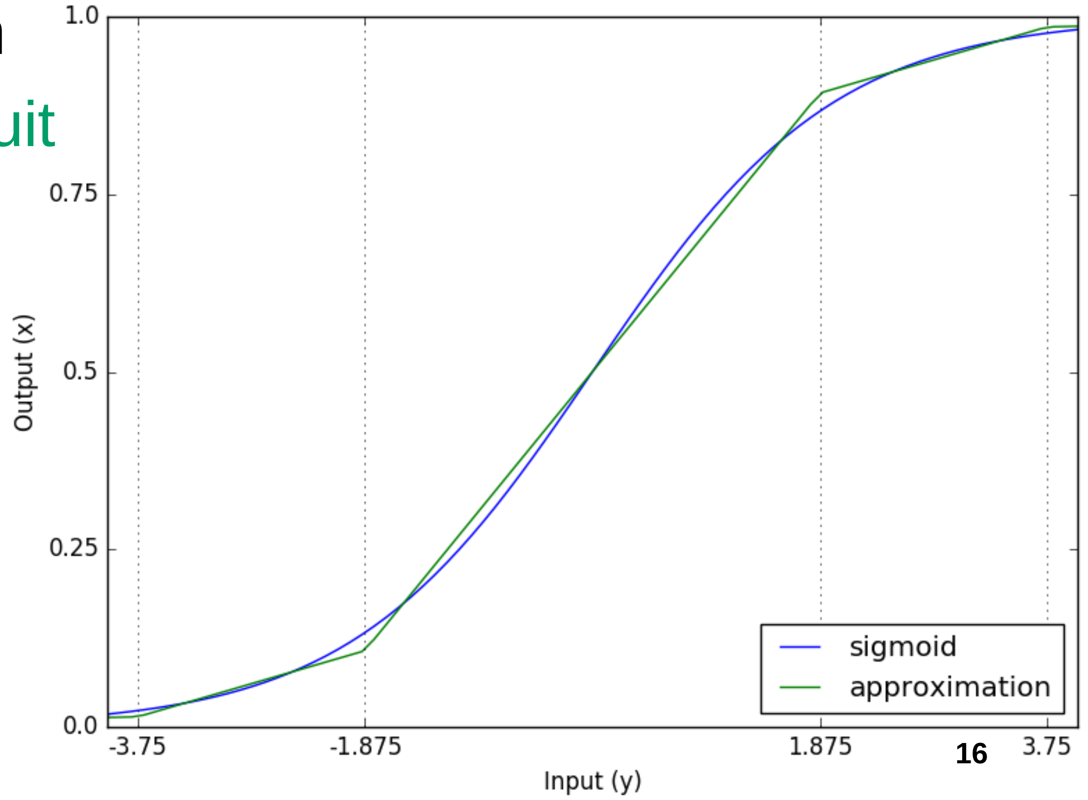
Piecewise linear functions e.g.,

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

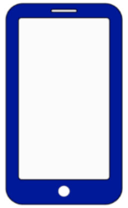
Oblivious activation/pooling functions $f(y)$

Smooth functions e.g.,

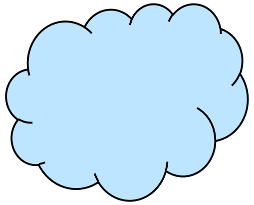
- Sigmoid: $x := 1/(1+e^{-y})$
- Oblivious sigmoid: $x^s + x^c := 1/(1+e^{-(y^s+y^c)})$
  - approximate by a piecewise linear function
  - then compute obliviously by a garbled circuit
  - empirically: ~14 segments sufficient



Combining the final result



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

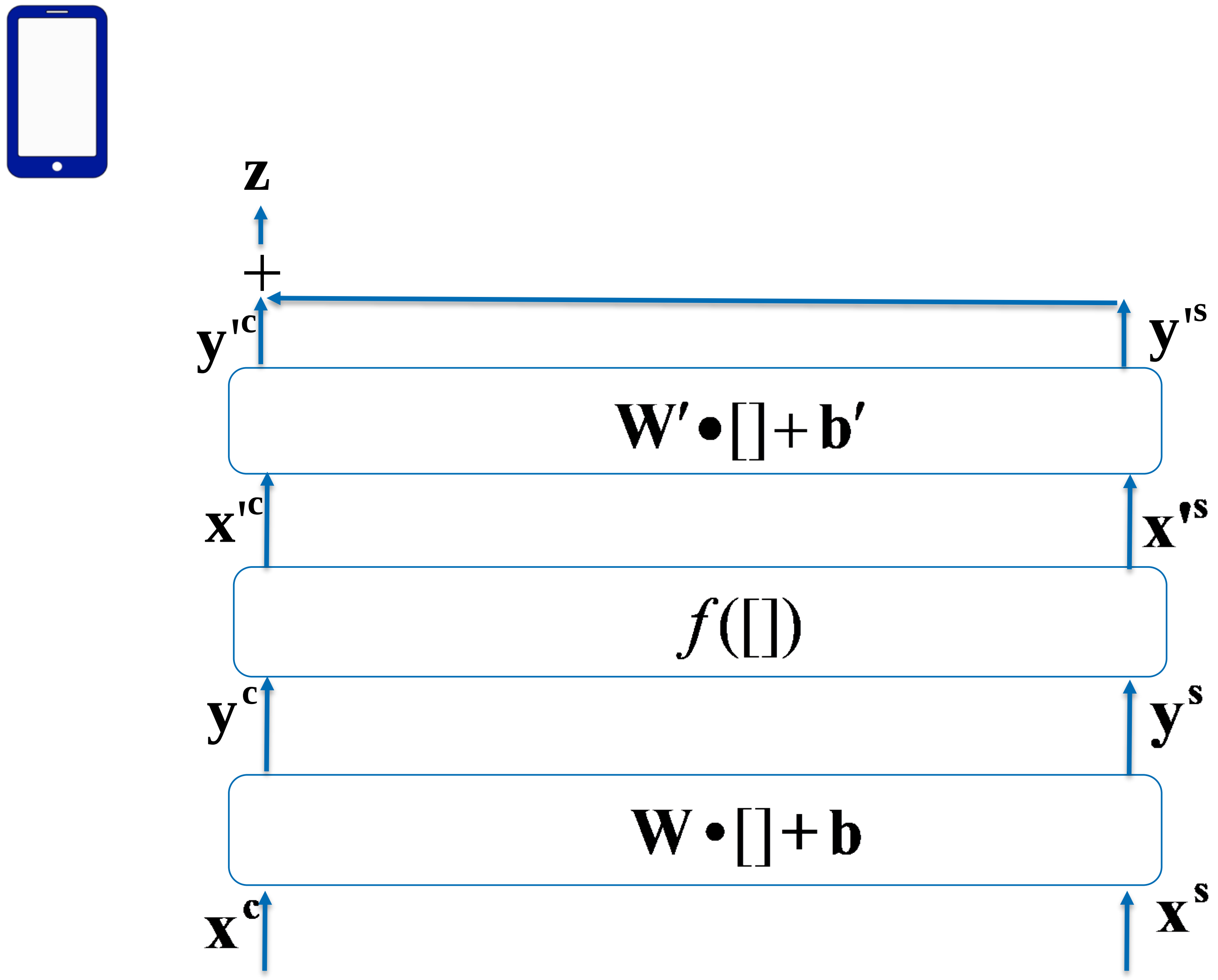


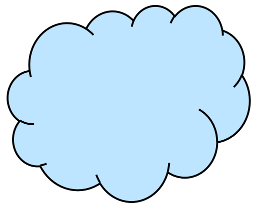
$$y_1 := y_1^s + y_1^c$$

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

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

Core idea: use secret sharing for oblivious computation





$(\mathbf{y}'^{c} + \mathbf{y}'^{s} = \mathbf{y}')$

$(\mathbf{x}'^{c} + \mathbf{x}'^{s} = \mathbf{x}')$

$(\mathbf{y}^{c} + \mathbf{y}^{s} = \mathbf{y})$

$(\mathbf{x}^{c} + \mathbf{x}^{s} = \mathbf{x})$

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

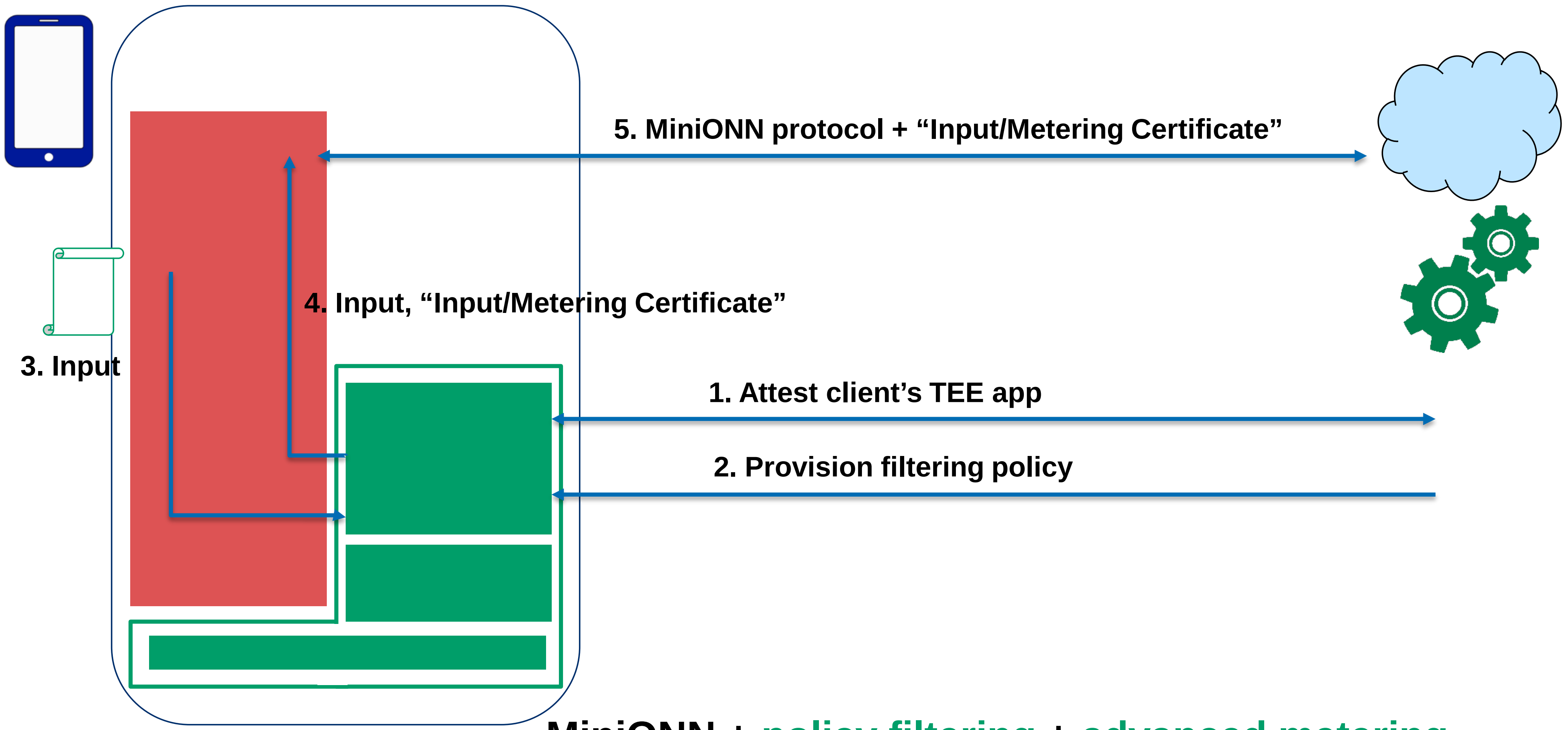
Still ~1000x slower than without privacy

Server can no longer filter requests or do sophisticated metering

Assumes online connectivity to server

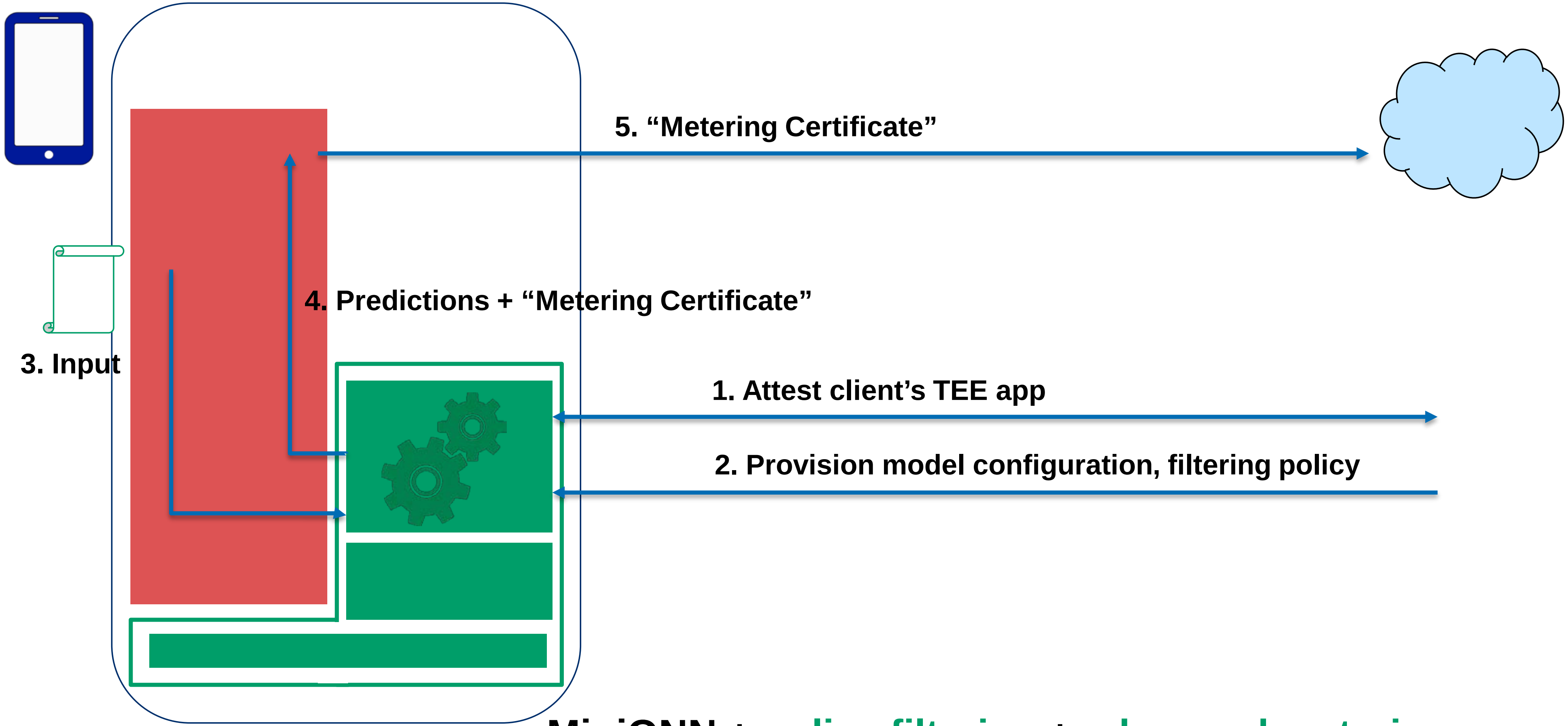
Reveals structure (but not params) of NN

Using a client-side TEE to vet input



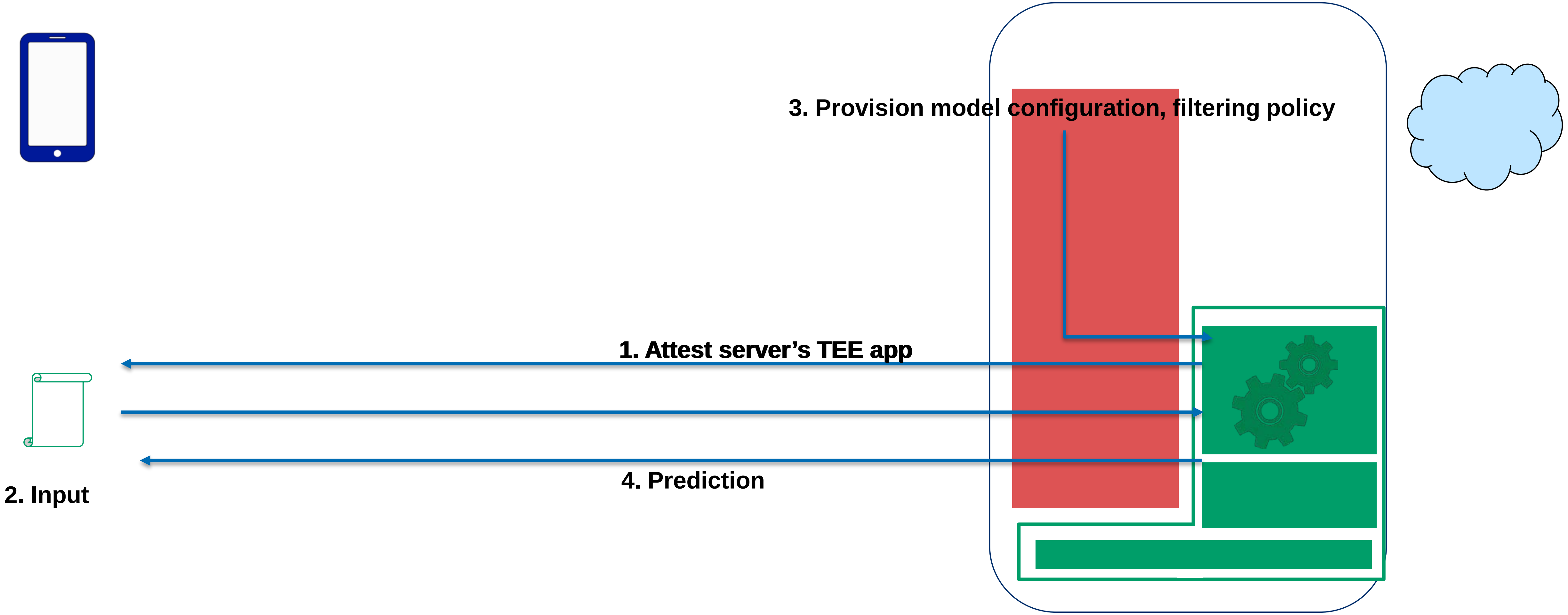
MiniONN + policy filtering + advanced metering

Using a client-side TEE to run the model



MiniONN + policy filtering + advanced metering
+ disconnected operation + performance + better privacy
- harder to reason about model secrecy

Using a server-side TEE to run the model



MiniONN + policy filtering + advanced metering
- disconnected operation + performance + better privacy

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

Trusted Computing can help realize improved security and privacy for ML

ML is very fragile in adversarial settings

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

Conclusions

Cloud-assisted services raise new security/privacy concerns

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

http://arxiv.org/abs/1606.01655

Cloud-assisted malware scanning

- Carousel approach is promising

Generalization to privacy-preserving ML predictions

[TLPEPA17] Circle Game, ASIACCS 2017
[LJLA17] MiniONN, ACM CCS 2017 https://eprint.iacr.org/2017/452