DeepContract: Controllable Authorization of Deep Learning Models

Xirong Zhuang[1], Lan Zhang[1,2], Chen Tang[1], Huiqi Liu[1], Bin Wang[3], Yan Zheng[3], Bo Ren[3].

1 University of Science and Technology of China,

2 Institute of Dataspace, Hefei Comprehensive National Science Center,

3 Tencent YouTu Lab.

Introduction

- Well-trained DL models have become recognized as **valuable intellectual property (IP)** for significant upfront investment during the training process.

High-quality datasets

Experienced experts

Computing resources

Introduction

- To fully capitalize on the value, owners are often willing to **offer their models as services**, as long as they can safeguard their IP rights and receive the corresponding revenue.

High-quality datasets

Experienced experts

Computing resources

Introduction

Cloud

Machine learning as a service (MLaaS)

On-device

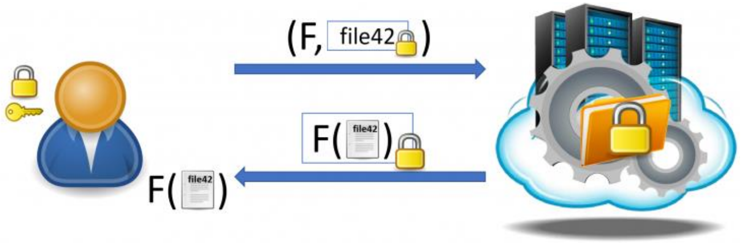
SGX

ARM TrustZone

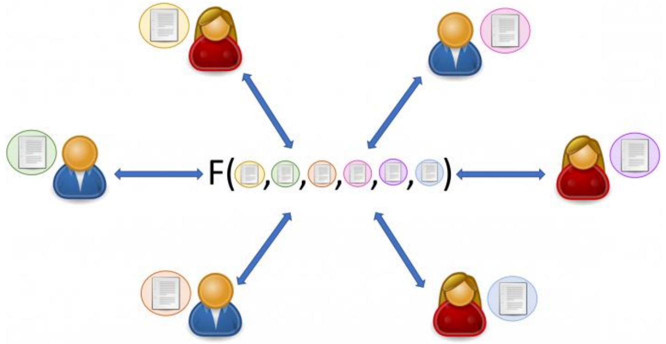
AMD SEV

Trusted Execution Environment (TEE)

Cryptographic



Homomorphic encryption (HE)



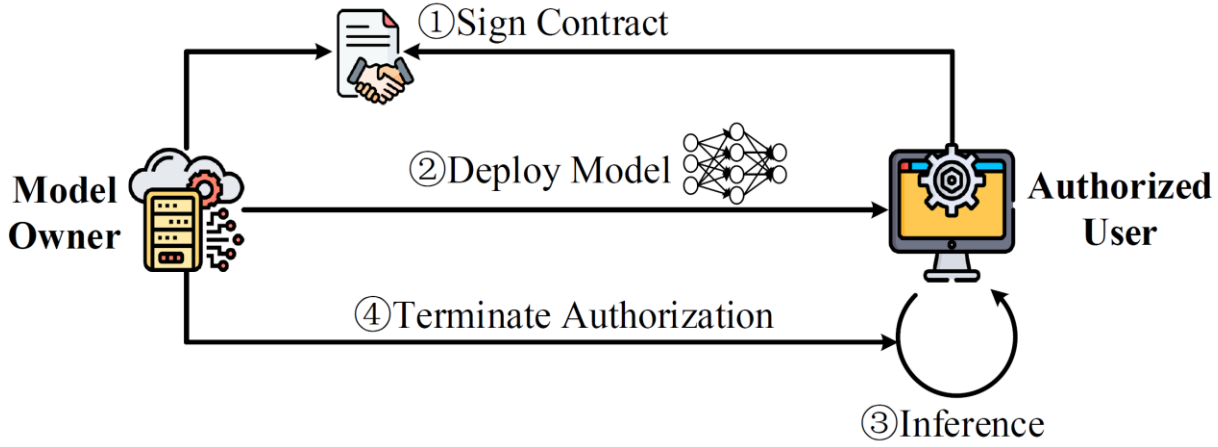
Secure multi-party computation (MPC)

Introduction

- **Active authorization**
  - Modifies models and granting correct usage to authorized users
  - Prevents models from being stolen by unauthorized users
  - Loses subsequent control over the authorized model

Introduction

- Active authorization
  - Modifies models and granting correct usage to authorized users
  - Prevents models from being stolen by unauthorized users
  - Loses subsequent control over the authorized model
- **Controllable authorization**
  - Model owners can **grant and revoke** the right to use their models



Design Goals

- **Model confidentiality**
  - Original models cannot be exposed to authorized users
  - Encrypted models cannot be restored effortlessly

Design Goals

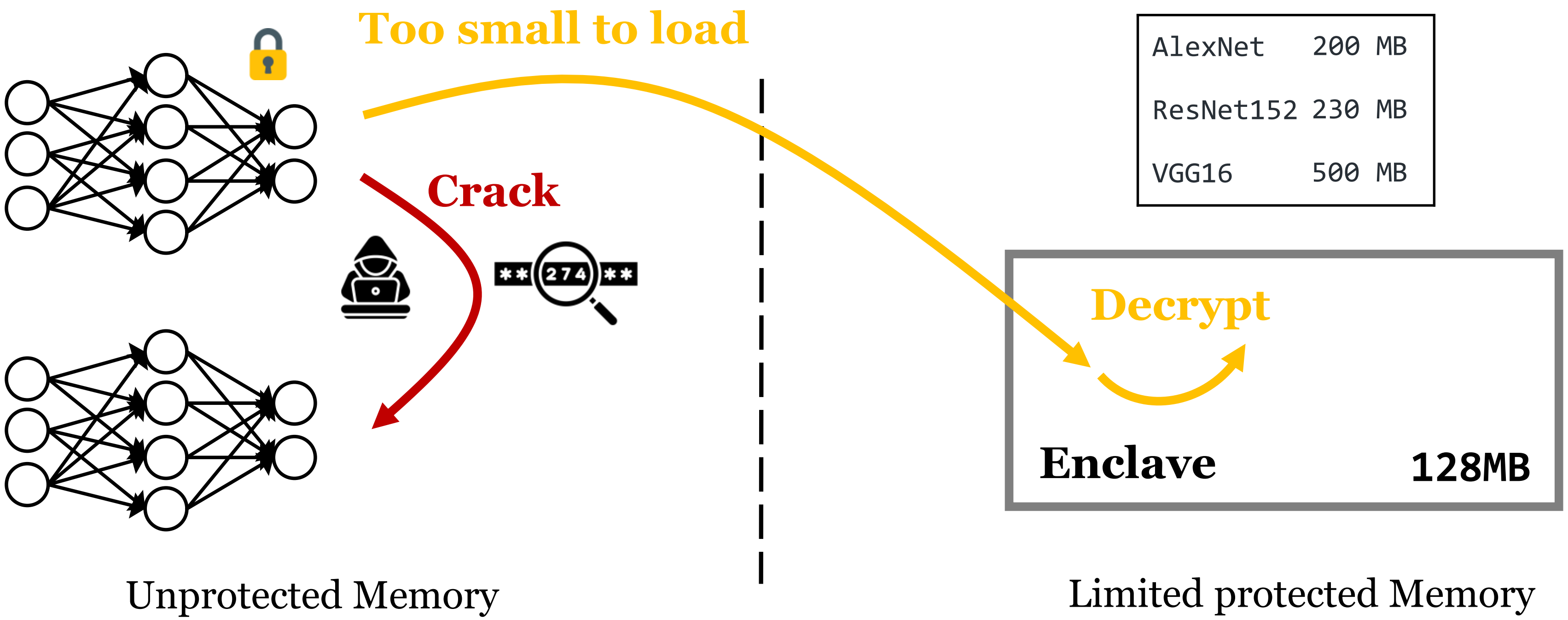
- Model confidentiality
  - Original models cannot be exposed to authorized users
  - Encrypted models cannot be restored effortlessly
- **Model controllability**
  - Conduct inference as agreed upon in the contract
  - Terminate the authorization in case of any breach of the contract

Design Goals

- Model confidentiality
    - Original models cannot be exposed to authorized users
    - Encrypted models cannot be restored effortlessly
- Model controllability
    - Conduct inference as agreed upon in the contract
    - Terminate the authorization in case of any breach of the contract
- **Minimal latency and resource consumption**
    - Satisfy the response-time requirements of real-life applications
    - Applicable on resource-constrained devices

Challenge

- Difficulty in confidentiality and efficient execution of the deployed model
    - Existing methods cannot resist cracking or fine-tuning attacks
    - Cannot decrypt the entire model straightforwardly within TEE since the limited memory

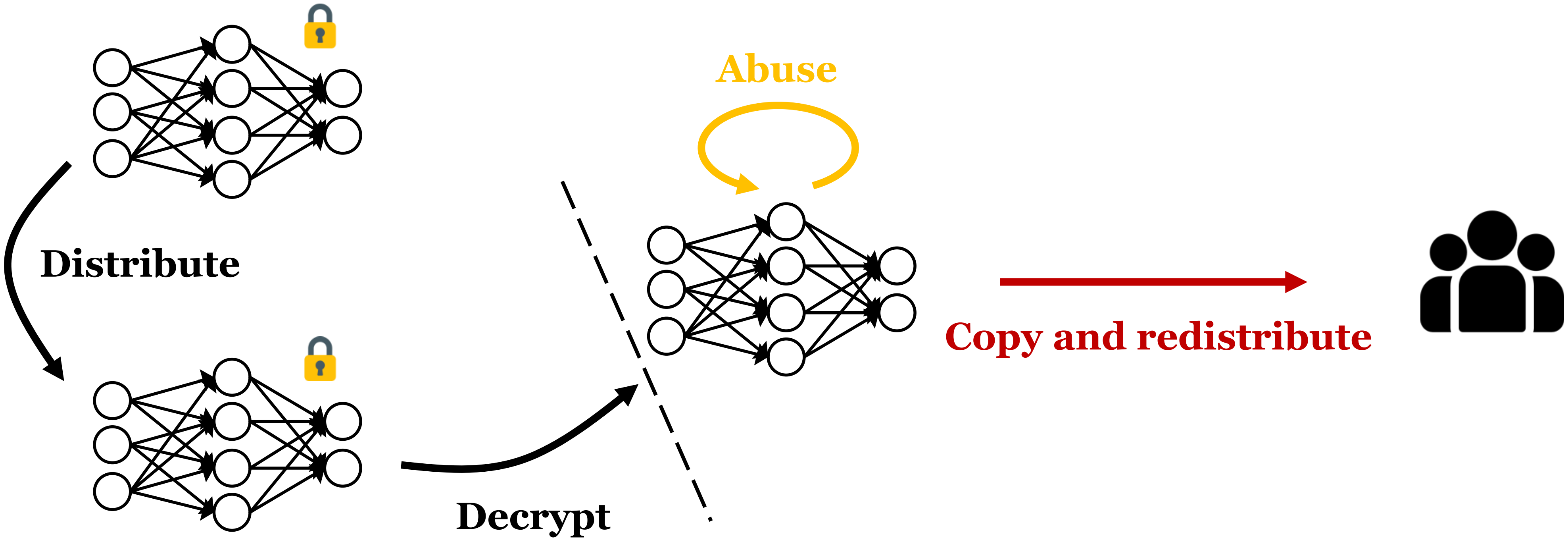


Model size

AlexNet	200 MB
ResNet152	230 MB
VGG16	500 MB

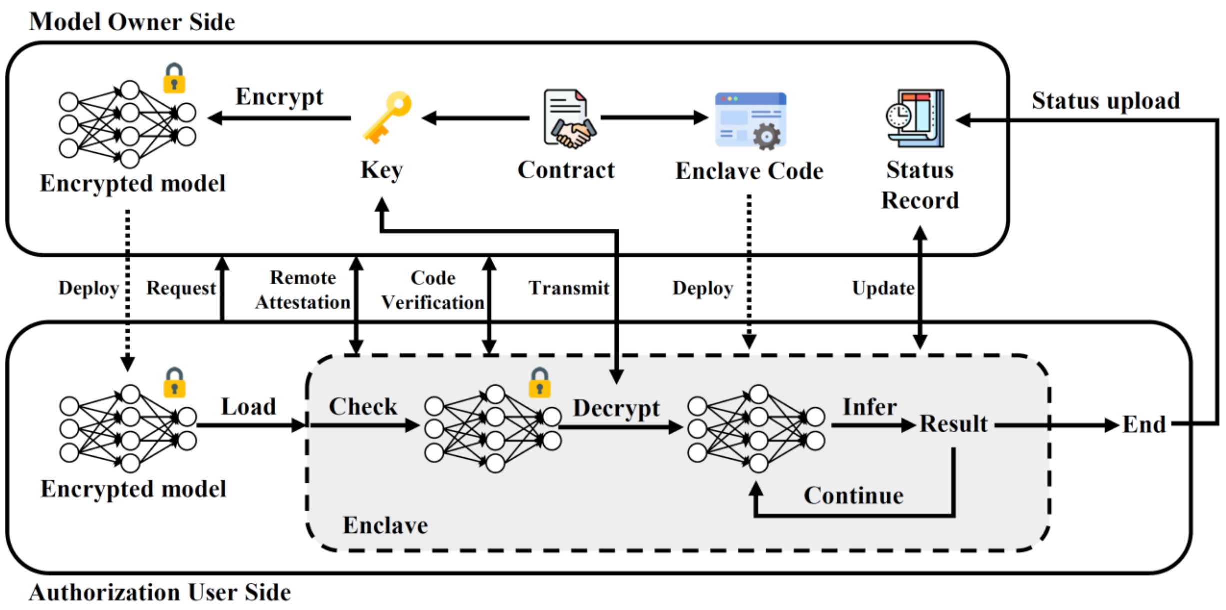
Challenge

- Uninterrupted and inescapable model controllability on remote devices
  - Existing works cannot offer such controllability after the distribution of models
  - The owner needs to maintain the connection with the decrypted model



System Overview

- Generates deployment materials on the owner's side
  - Pre-signed contract -> Encryption key & Encrypted model & Enclave code
- Performs controlled inference on the user's side
  - Enclave initialization -> Inference for a specified period as per the contract



Confidentiality

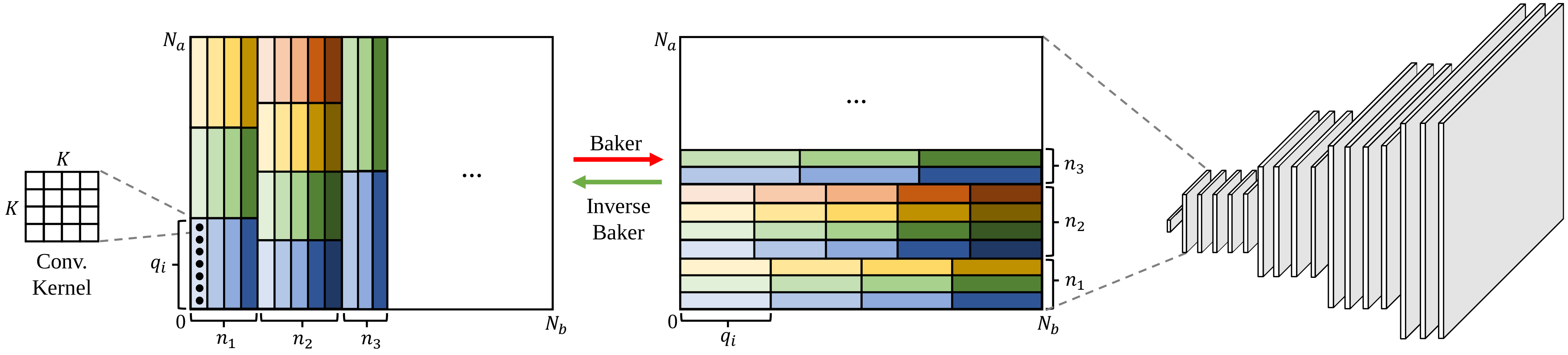
- Encryption requirements of controllable authorization
  - Compatible with the SGX-based DL inference
  - No loss of inference accuracy after decryption
  - Uncrackable with reasonable time and effort cost

Layer-wise model encryption with baker mapping

Confidentiality

- Encryption requirements of controllable authorization
  - Compatible with the SGX-based DL inference
  - No loss of inference accuracy after decryption
  - Uncrackable with reasonable time and effort cost

Layer-wise model encryption with baker mapping

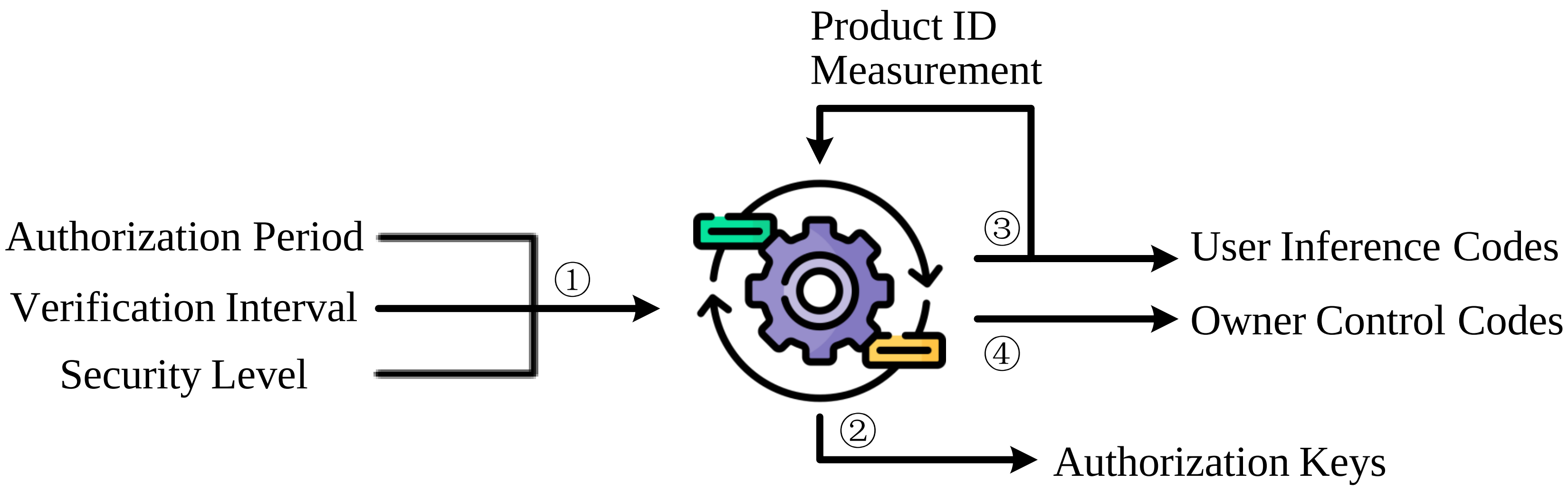


Controllability

- Contract-based code generation
  - Ensure the remote device performs a series of intended operations
  - Ensure the corresponding user codes are not tampered with
  - Pre-generated and verifiable enclave codes

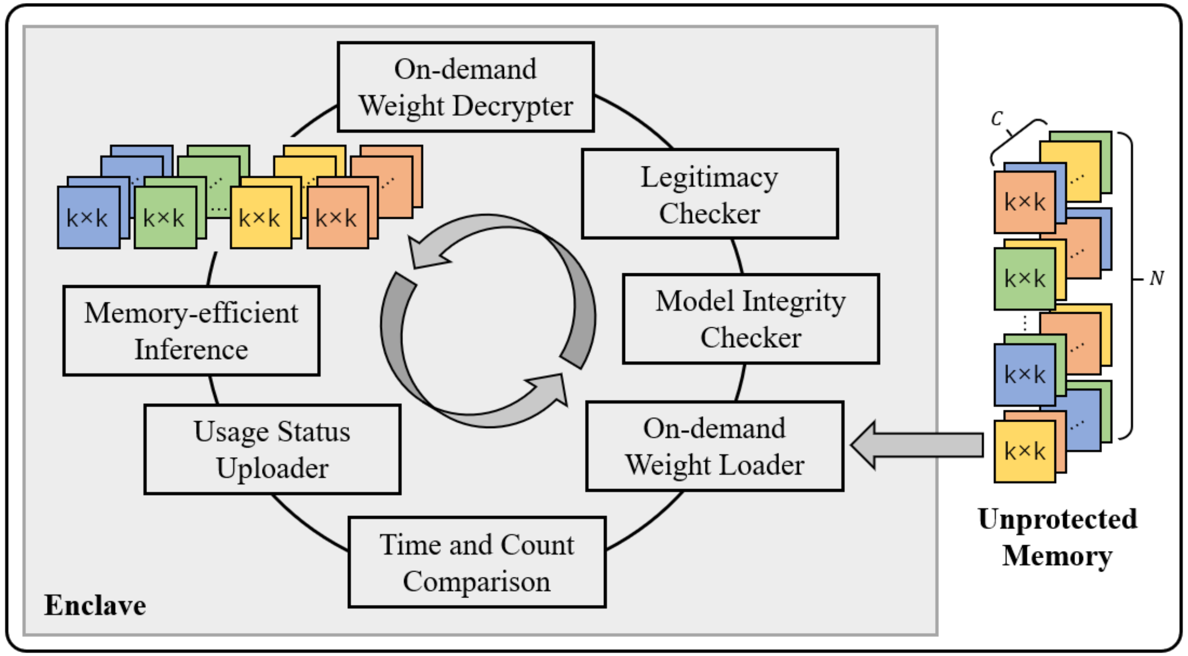
Controllability

- Contract-based code generation
  - Ensure the remote device performs a series of intended operations
  - Ensure the corresponding user codes are not tampered with
  - Pre-generated and verifiable enclave codes



Controllability

- Controlled model inference
  - Dynamically load the needed encrypted weights
  - Parallelly pipeline: integrity check / decryption/ inference
  - Promptly upload the current usage status



Evaluation

1) How is the efficiency and security of model encryption?

- Baseline 1: straightforward encryption method: Deep Lock
- Baseline 2: mapping encryption method: Chaotic Weights

2) Can DeepContract run DNN within SGX's memory limit?

3) How much is the overhead of DeepContract?

- Baseline 1: in-enclave inference: Occlumency
- Baseline 2: secure two-party computation using HE/MPC

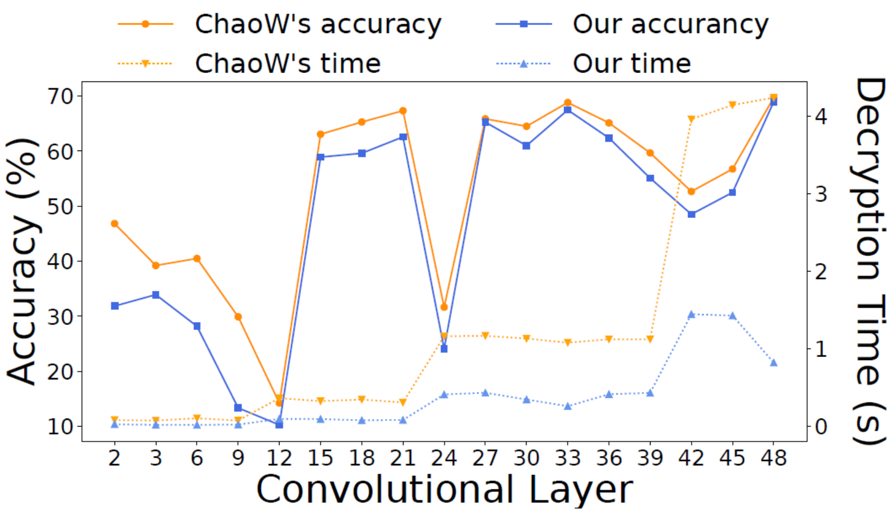
Evaluation

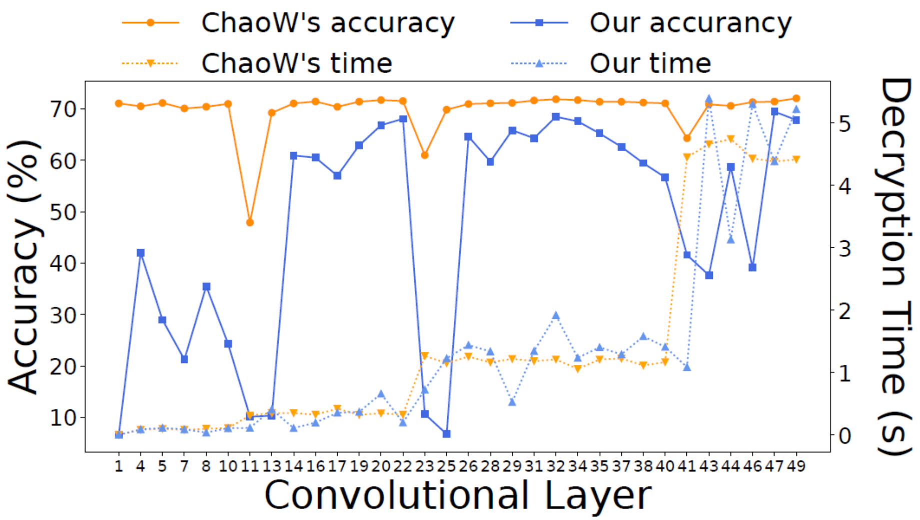
1) How is the efficiency and security of model encryption?

Decryption Speed

- **8.9x** faster than **DeepLock**
- **2.4x** faster than **ChaoW**

Scheme	VGG16	ResNet18	ResNet50	ResNet101
DeepLock [2]	23.53	17.60	37.52	69.41
ChaoW [30]	5.14	5.22	13.43	20.23
DeepContract	1.58	1.42	9.82	15.21

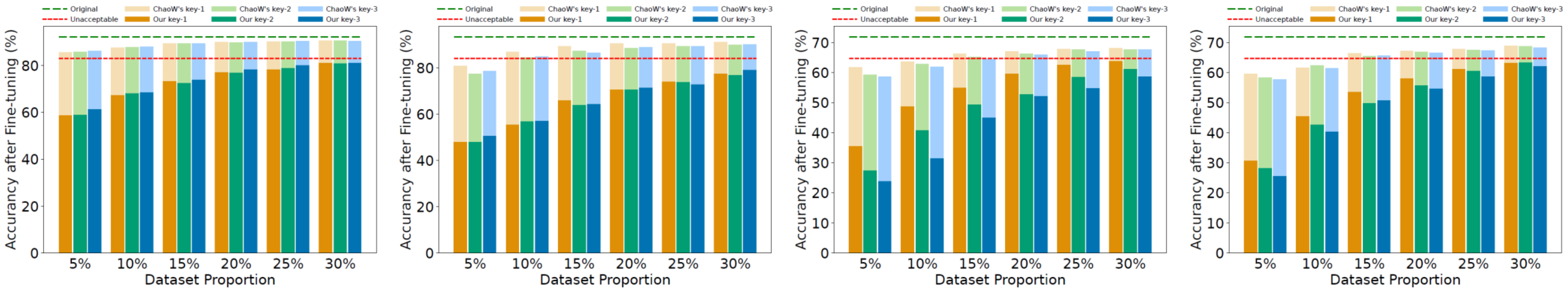




Evaluation

1) How is the efficiency and security of model encryption?

Resistance to fine-tuning attacks



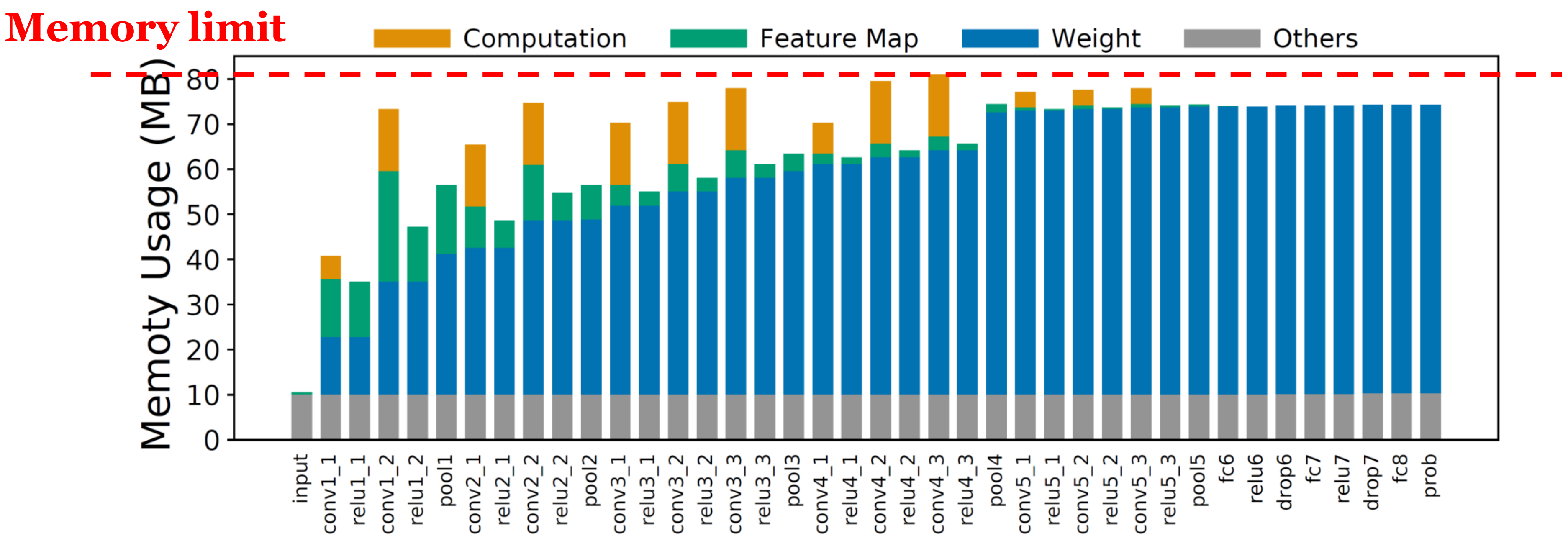
(a) VGG-16 (b) ResNet-18 (c) ResNet-50 (d) ResNet-101

Encrypted models cannot be restored to unacceptable accuracy

even with staggering proportion **(25%-30%)** of the training dataset!

Evaluation

2) Can DeepContract run DNN within SGX's memory limit?



The maximum memory usage is always under

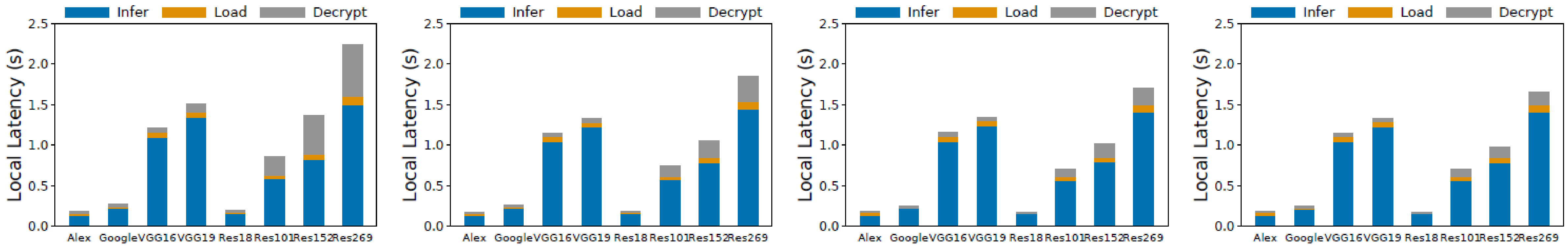
the available Enclave Page Cache memory size of SGXv1.

Evaluation

3) How much is the overhead of DeepContract?

Inference Speed

- **23%** slower than **Occlumency** (Not protecting model weights)
- **8%-13%** slower at more relaxed security levels



(a) 30% of the dataset (b) 20% of the dataset (c) 10% of the dataset (d) 5% of the dataset

Evaluation

3) How much is the overhead of DeepContract?

Compared to cryptographic methods

- More **real-time** inference
- Only **minor data transfer** required

Scheme	Framework	MNIST		CIFAR-10	
		Run Time (s)	Data Transfer (MB)	Run Time (s)	Data Transfer (MB)
HE	SHE [33]	9.3	123	2258	160
HE	LoLa [4]	2.2	18	730	370
MPC	EzPC [7]	5.1	501	265.6	40683
MPC	Chameleon [40]	2.24	11	52.67	2650
MPC	XONN [39]	0.15	32	5.79	2599
HE-MPC	nGraph-HE2 [3]	64.32	51	1824	3775
HE-MPC	MiniONN [32]	9.32	658	544	9272
HE-MPC	Gazelle [23]	0.81	70	12.9	1236
TEE	DeepContract	0.13	0.0041	0.18	0.0045

Stage	Data	Size	Frequency
Deployment	Encrypted Model	90.7 MB	Once in an authorization
	Enclave Codes	22.6 MB	
Transmission	Authorization Key	1.5 KB	Once in an authorization
	Hash Values	2.1 KB	
	Attestation Message	3.1 KB	Once in a verification cycle
	Usage Status	0.2 KB	

Thank You !

DeepContract: Controllable Authorization of Deep Learning Models

Xirong Zhuang[1], Lan Zhang[1,2], Chen Tang[1], Huiqi Liu[1], Bin Wang[3], Yan Zheng[3], Bo Ren[3].

1 University of Science and Technology of China,

2 Institute of Dataspace, Hefei Comprehensive National Science Center,

3 Tencent YouTu Lab.

中国科学技术大学
University of Science and Technology of China

IDATA 合肥综合性国家科学中心 数据空间研究院

腾讯优图