MindSpore 1.9 Security Target

Issue	**0.2.2**
Date	**2024-02-26**

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: http://www.huawei.com

Email: support@huawei.com

Change History

Date	Issue	Change Description	Author
2022-07-05	0.1	Updated based on 《CC MindSpore 1.2-ST_V1.3.6_BR.docx》. 1. change the MindSpore Version and the third party	Liu Liu
2022-12-21	0.1.1	Update version number of MindSpore; Update format of SFR description	Liu Liu
2023-08-18	0.1.2	Update version reference of TOE	Liu Liu
2023-09-04	0.1.3	Update version reference of TOE	Liu Liu
2023-11-08	0.1.4	Update test results in 6.1.2	Xiulang Jin
2023-11-28	0.2.0	Change version number	Xiulang Jin
2023-12-01	0.2.1	Update reference version number	Xiulang Jin
2024-02-26	0.2.2	Update version number	Xiulang Jin

Contents

List of Figures

1 Security Target introduction

The ST describes what is evaluated, including the exact security properties of the TOE in a manner that the potential consumer can rely on.

1.1 Security Target reference

Title:	MindSpore 1.9 Security Target
Version:	0.2.2
Author:	Huawei

1.2 TOE reference

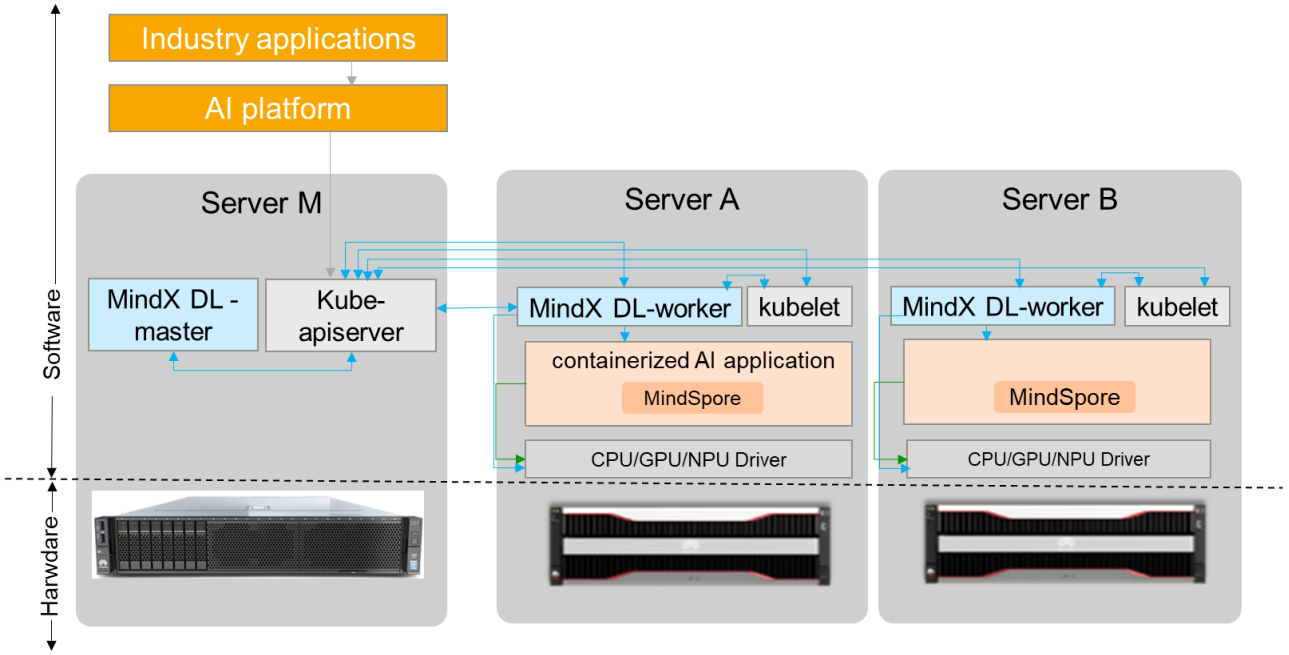
TOE name:	MindSpore
TOE version:	1.9
TOE developer:	Huawei
TOE components:	MindSpore -MindSpore version 1.9.0 MindSpore -MindArmour version 1.9.1 MindSpore -MindInsight version 1.9.0

1.3 TOE Overview

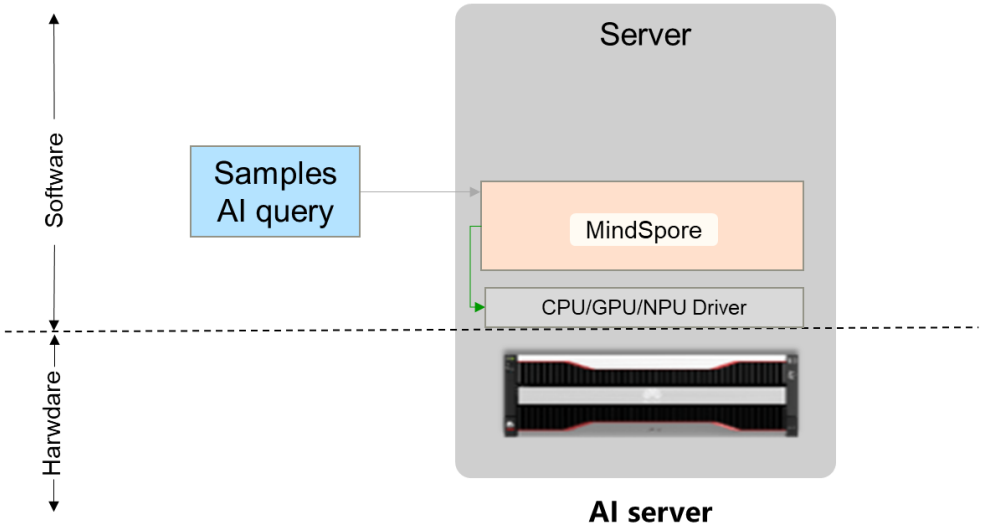
The Target of Evaluation is an open source deep learning training/inference framework software developed by Huawei. It will be referred to as the TOE throughout this document.

MindSpore is provided as a library that is used by AI Application developers to provide AI services with different deployments modes.

1. distributed mode: MindSpore may be deployed with distributed deployment software, such as MindX which is also from Huawei and used for distributed resource management and monitoring, generation of communication configuration for distributed training sets and so on:



2. standalone mode:



The two main out of scope unsecured use cases for MindSpore are:

1. create AI Applications that define and train a deep learning model with a training dataset and MindSpore:

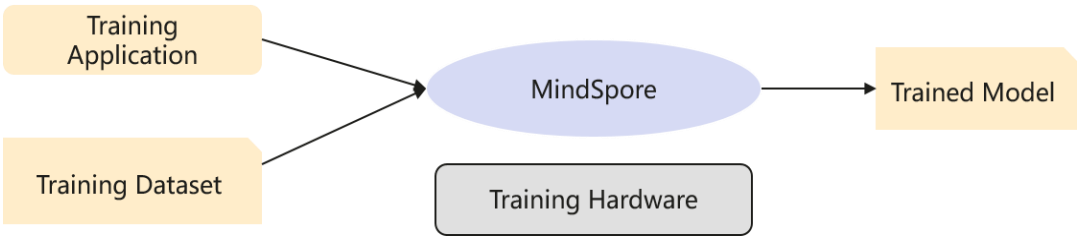


Figure 1 Unsecured training use case

2. create AI Applications that use trained models and MindSpore to conduct inference of samples:

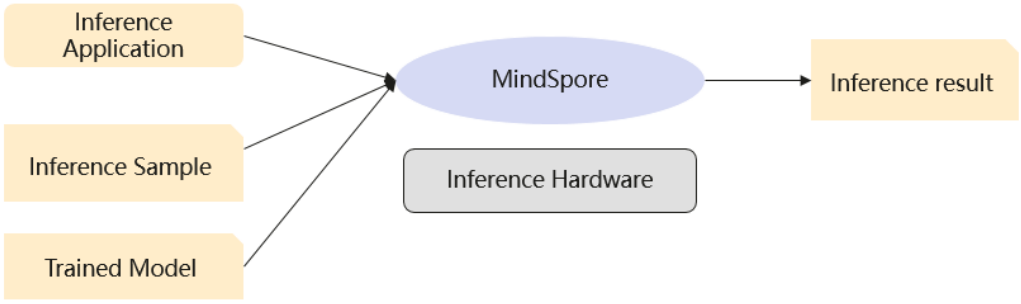


Figure 2 Unsecured inference use case

In the evaluated configuration, the AI Application developer will use MindSpore-MindSpore along with the MindSpore-MindArmour library to improve the generated AI Application deep learning models into a protected model or to attack the original model in the field of computer vision. The security functionality allows for evaluation and comparison of the robustness of models, leading to model design improvements; it is therefore an important part of the security functionality.

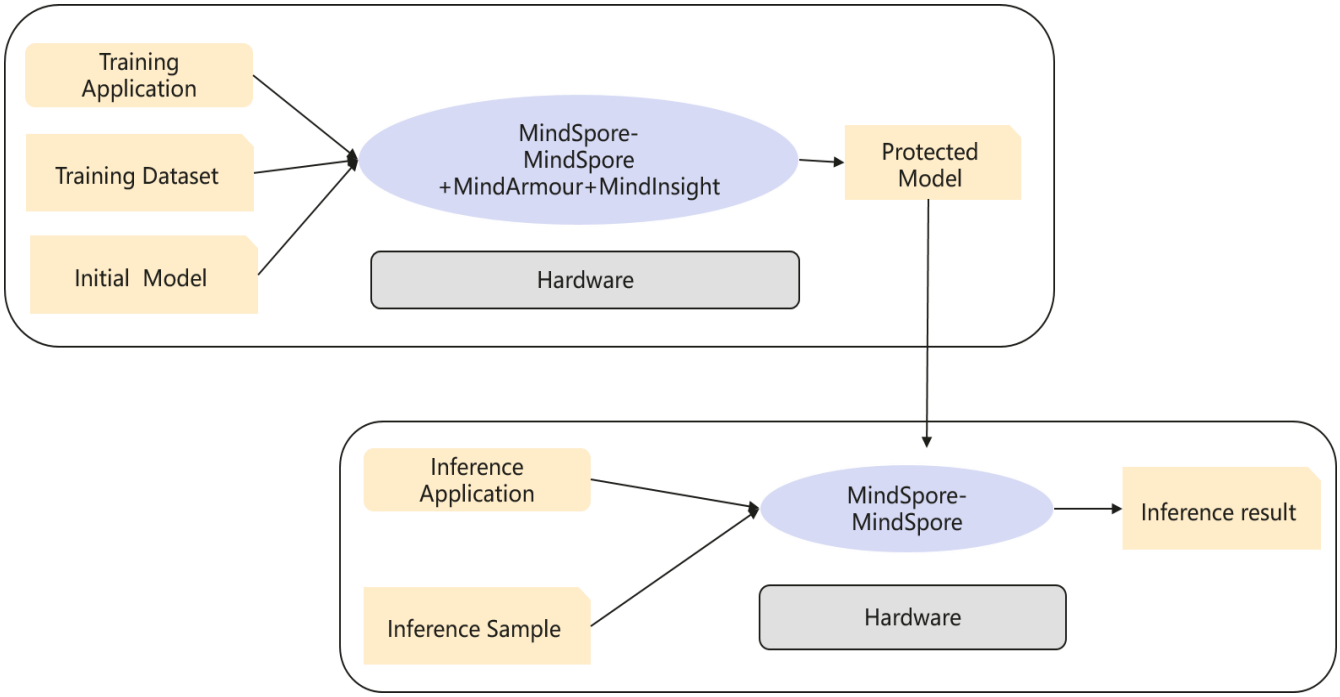


Figure 3 Model hardening use case

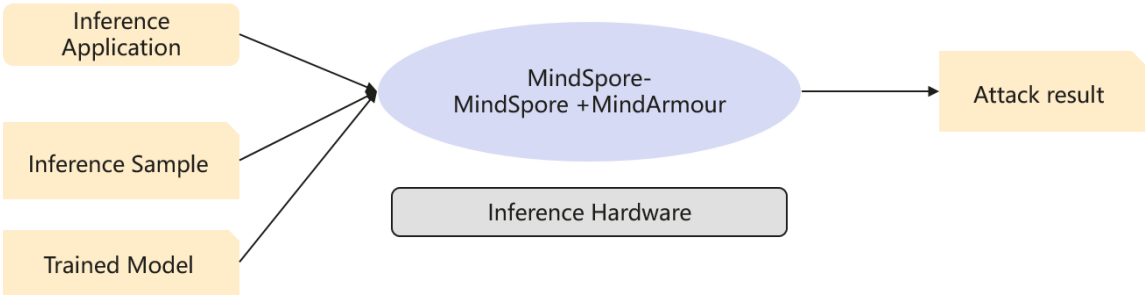


Figure 4 Attack use case

The MindSpore supports other out of scope functionalities and is provided along with other out of scope developer supporting tools:

- The AI Application developer can use the supporting tool MindSpore-MindInsight to support the development by providing visualization of the training process, training performance and training result traceability, thereby simplifying the optimization of the model.
- The AI Application developer can also extract robustness and privacy metrics when using a model that are valuable to judge third party models.

Both unsecured, hardening and attack use cases can be deployed in two types of environment:

1. In local single node environments of the following types:
   - CPU based
   - GPU based (NVIDIA CUDA)
   - Huawei Ascend based

2. In heterogeneous cluster environments of the mentioned node types. When nodes are of the GPU or CPU type, they communicate through OpenMPI and NCCL; when nodes are of the Ascent type, they communicate using proprietary software embedded in the node.

The choice of nodes will depend on the AI developer preference and availability and will impact the performance of the TOE, but not the security functionalities in scope.

Figure 5 shows in purple the components in scope of the security evaluation and in orange and grey the components out of the security evaluation scope. Note that the focus of the evaluation is in generalized attack and hardening of generated models, not the individual models themselves.

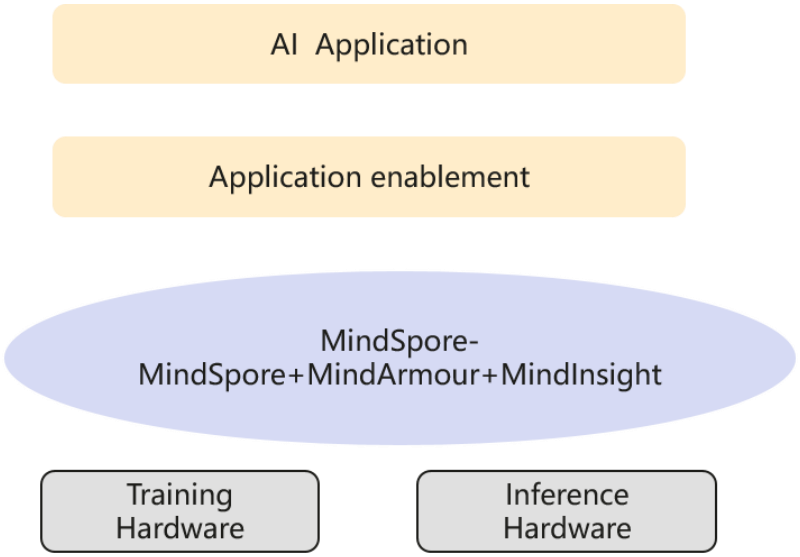


Figure 5 TOE Scope

From the node software point of view, the MindSpore-MindSpore component has the following non-TOE software dependencies:

- Python version >=3.7 and the following libraries:
  - asttokens >= 1.1.13
  - astunparse >= 1.6.3
  - cffi >= 1.12.3
  - decorator >= 4.4.0
  - easydict >= 1.9
  - numpy >= 1.17.0, <= 1.17.5
  - packaging >= 20.0
  - pillow >= 6.2.0
  - protobuf >= 3.8.0
  - scipy >= 1.5.2
  - setuptools >= 40.8.0
  - sympy >= 1.4
  - wheel >= 0.32.0
  - psutil >= 5.6.1
- CUDA 10.1 (for GPU nodes).

- CuDNN >= 7.6 (for GPU nodes).
- OpenMPI 4.0.3 (for single-node/multi-GPU and multi-node/multi-GPU training).
- NCCL 2.7.6-1 (for single-node/multi-GPU and multi-node/multi-GPU training).
- Ascend 910 AI processor software package version: Atlas Data Center Solution 21.0.1 (for Ascend nodes).
- gmp 6.1.2 (for Ascend nodes).

From the node point of view the following non-TOE hardware dependencies are needed:

- For the CPU node case a computer with Ubuntu 18.04 x86_64, Ubuntu 18.04 aarch64 or Windows 10 x86_64.
- For the GPU node case a computer with Ubuntu 18.04 x86_64 and one or more NVIDIA GPUs with CUDA 10.1 support.
- For the Ascend node case a computer with either Ubuntu 18.04 aarch64, Ubuntu 18.04 x86_64, EulerOS 2.8 aarch64 or EulerOS 2.5 x86_64 and one or more Ascend 910 AI processors.

In case the deployment is on a computing cluster, individual nodes can be a heterogeneous combination of the aforementioned types and the following additional requirements are needed:

- Standard network equipment and cabling to interconnect the nodes.

MindArmour depends on MindSpore and the following python libraries:

- matplotlib >= 3.2.1
- numpy >= 1.17.0
- Pillow >= 2.0.0
- pytest >= 4.3.1
- scikit-learn >= 0.23.1
- scipy >= 1.5.3
- setuptools >= 40.8.0
- wheel >= 0.32.0

MindInsight depends on MindSpore and the following python libraries:

- Click>=7.0
- Flask>=1.1.1
- Flask-Cors>=3.0.8
- google-pasta>=0.1.8
- grpcio>=1.35.0
- gunicorn>=20.0.4
- itsdangerous>=1.1.0
- Jinja2>=2.10.1
- MarkupSafe>=1.1.1
- marshmallow>=3.10.0
- numpy>=1.17.0
- pandas>=1.0.4
- pillow>=6.2.0
- protobuf>=3.8.0
- psutil>=5.7.0
- pyyaml>=5.3.1
- scikit-learn>=0.23.1
- scipy>=1.5.2

- six>=1.12.0
- treelib>=1.6.1
- Werkzeug>=1.0.0
- yapf>=0.30.0
- XlsxWriter>=1.3.2

1.4 TOE Description

The TOE is purely a software TOE and delivered as the following components tagged with the same version as the whole framework:

- MindSpore 1.9.0
- MindArmour 1.9.1
- MindInsight 1.9.0

The software components are used together to generate, attack, defend and apply inference to computer vision deep learning models.

The software components are delivered as prebuilt individual packages and are available from the download website https://www.mindspore.cn/install/en. Note that the components source code can be individually obtained from https://gitee.com/mindspore by cloning the mindspore, mindarmour and mindinsight git repositories using the evaluated version tag.

- MindSpore-MindSpore
    - Platform: Ascend 910 and Ascend310
        - OS: Linux-aarch64, mindspore_ascend-1.9.0-cp37-cp37m-linux_aarch64.whl, SHA256: 13967c4f9eaf4f17e04d186ffb8aae73fecc9177877b777c20509ac1c5dd4542
        - OS: Linux-x86_64, mindspore_ascend-1.9.0-cp37-cp37m-linux_x86_64.whl, SHA256: a4f6e6c8a470e1c0086d3d97b447d2dc639e8580e72c061da30b3f4f3b302295
    - Platform: GPU CUDA 10.1:
        - OS: Linux-x86_64, mindspore_gpu-1.9.0-cp37-cp37m-linux_x86_64.whl, SHA256: e990ffb81ccc939553e256cd2845838a353471b60b72098ab33ff41f18dfbed6
    - Platform: GPU CUDA 11.1:
        - OS: Linux-x86_64, mindspore_gpu-1.9.0-cp37-cp37m-linux_x86_64.whl, SHA256: db5b20e66de2fcf8433cc3193aefdf0a9c88000225314dd42b47bb97fdfa9eb6
    - Platform: CPU:
        - OS: Linux-aarch64, mindspore-1.9.0-cp37-cp37m-linux_aarch64.whl, SHA256: e0aec18d84484a5d33bdb02562a1fda4be576b8227b78fe4f435fac270bb712e
        - OS: Linux-x86_64, mindspore-1.9.0-cp37-cp37m-linux_x86_64.whl, SHA256: 16ed2fb42d1197bcbee1e68bb23dc0b41256b8836c1134ac4d5b11356a02bd29

- MindSpore-MindInsight:
    - Platform: Any:
        - OS: Any, mindinsight-1.9.0-py3-none-any.whl,SHA256: d401ec851c34f8b0e86c1435c7a76989520c1dab274fcd82f0fd291a19881de1
- MindSpore-MindArmour:
    - Platform Any:
        - OS: Any, mindarmour-1.9.1-py3-none-any.whl ,SHA256: 9115ab2fbe2337616f1046efaff920bf043a80fb914d92377e38be30d74f6dbd

1.4.1 MindSpore-MindSpore

MindSpore is a new open source deep learning training/inference framework that could be used for mobile, edge and cloud scenarios. MindSpore is designed to provide development experience with friendly design and efficient execution for the data scientists and algorithmic engineers, native support for Ascend AI processor, and software hardware co-optimization. At the meantime MindSpore as a global AI open source community, aims to further advance the development and enrichment of the AI software/hardware application ecosystem.

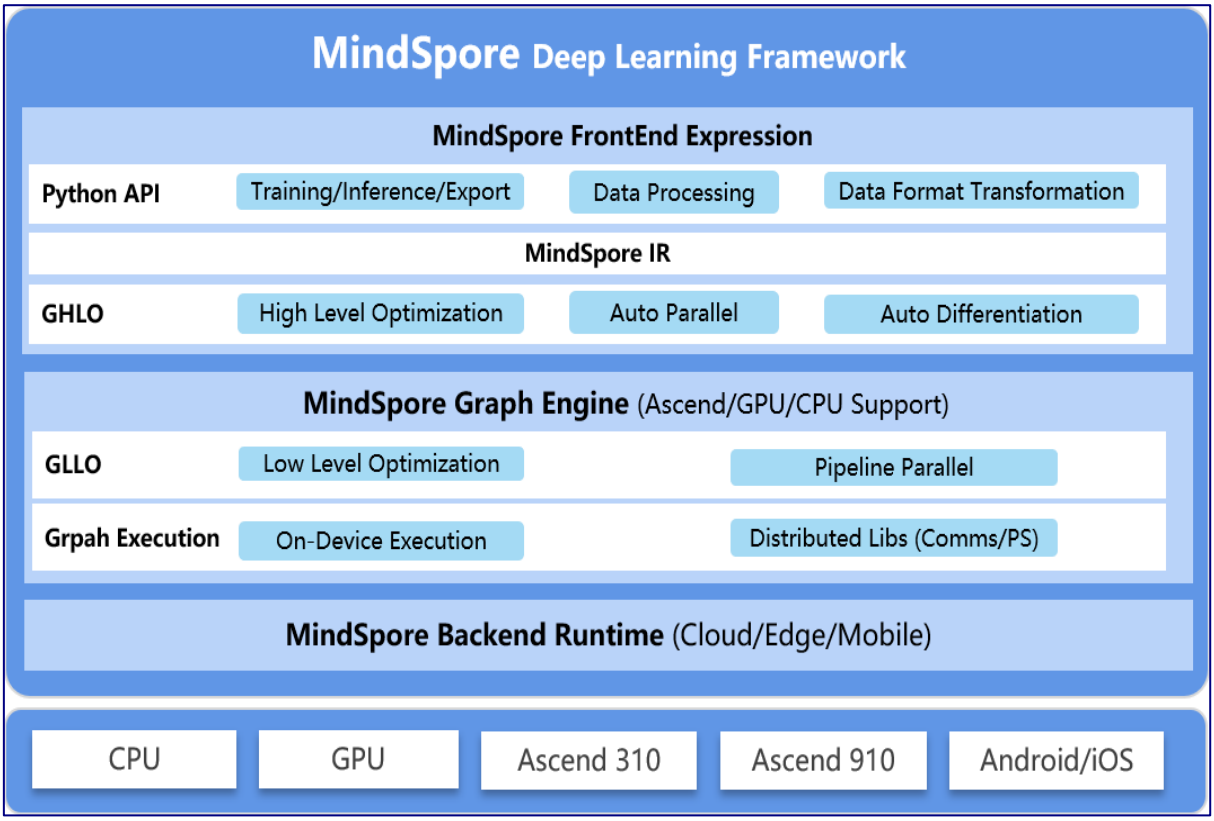


Figure 6 MindSpore framework overview

1.4.1.1 Automatic Differentiation

There are currently three automatic differentiation techniques in mainstream deep learning frameworks:

- **Conversion based on static compute graph**: Convert the network into a static data flow graph at compile time, then turn the chain rule into a data flow graph to implement automatic differentiation.
- **Conversion based on dynamic compute graph**: Record the operation trajectory of the network during forward execution in an operator overloaded manner, then apply the chain rule to the dynamically generated data flow graph to implement automatic differentiation.
- **Conversion based on source code**: This technology is evolving from the functional programming framework and performs automatic differential transformation on the intermediate expression (the expression form of the program during the compilation process) in the form of just-in-time compilation (JIT), supporting complex control flow scenarios, higher-order functions and closures.

> TensorFlow adopted static calculation diagrams in the early days, whereas PyTorch used dynamic calculation diagrams. Static maps can utilize static compilation technology to optimize network performance, however, building a network or debugging it is very complicated. The use of dynamic graphics is very convenient, but it is difficult to achieve extreme optimization in performance.
>
> But MindSpore finds another way, automatic differentiation based on source code conversion. On the one hand, it supports automatic differentiation of automatic control flow, so it is quite convenient to build models like PyTorch. On the other hand, MindSpore can perform static compilation optimization on neural networks to achieve great performance.

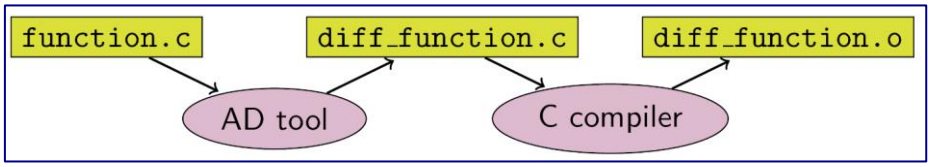


Figure 7 Automatic Differentiation overview

The implementation of MindSpore automatic differentiation can be understood as the symbolic differentiation of the program itself. Because MindSpore IR is a functional intermediate expression, it has an intuitive correspondence with the composite function in basic algebra. The derivation formula of the composite function composed of arbitrary basic functions can be derived. Each primitive operation in MindSpore IR can correspond to the basic functions in basic algebra, which can build more complex flow control.

1.4.1.2 Automatic Parallel

The goal of MindSpore automatic parallel is to build a training method that combines data parallelism, model parallelism, and hybrid parallelism. It can automatically select a least cost model splitting strategy to achieve automatic distributed parallel training.

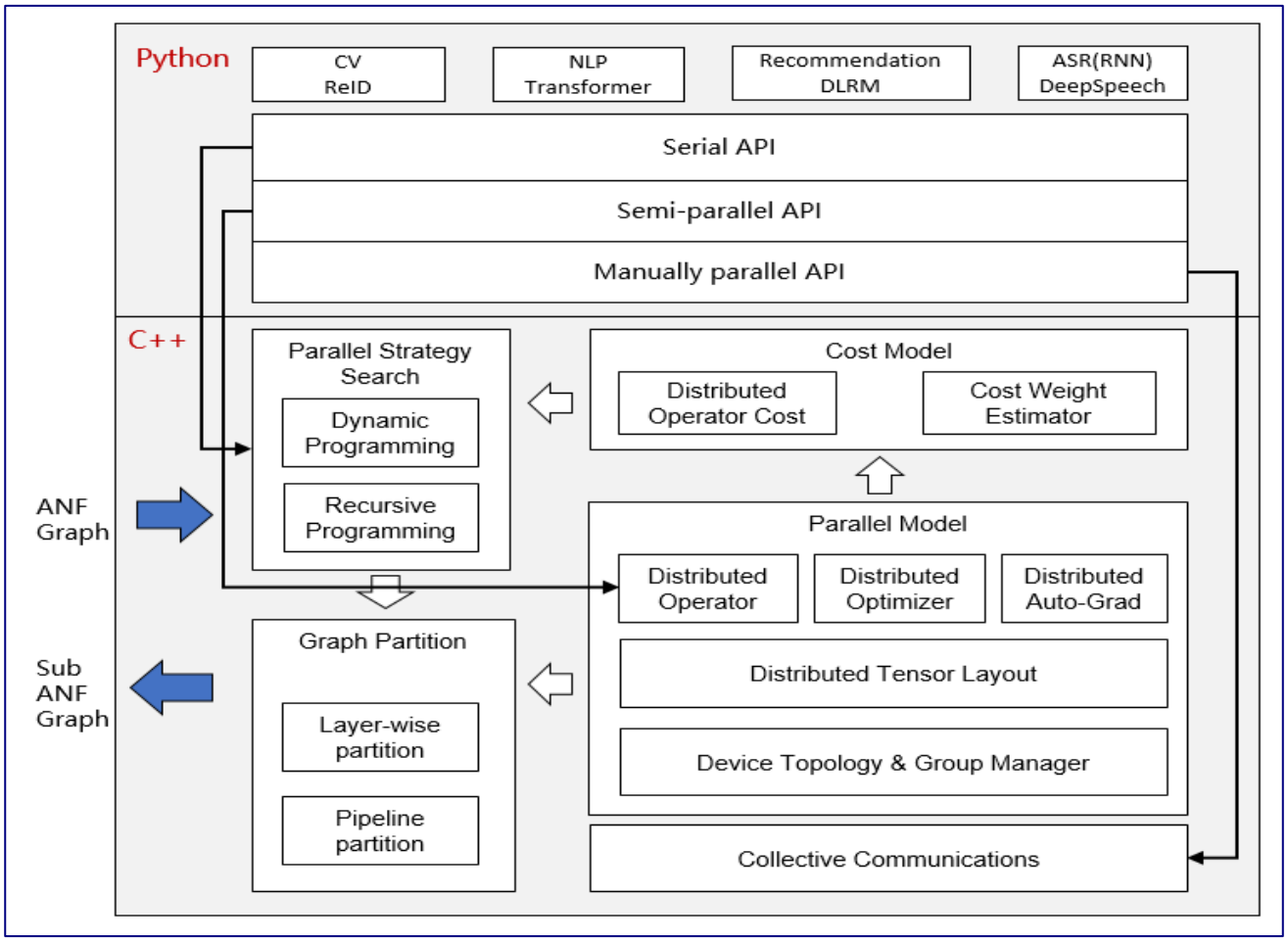


Figure 8 Automatic Parallel overview

At present, MindSpore uses a fine-grained parallel strategy of splitting operators, that is, each operator in the figure is split into a cluster to complete parallel operations. The splitting strategy during this period may be very complicated, but as a developer advocating Pythonic, you don't need to care about the underlying implementation, as long as the top-level API compute is efficient.

1.4.2 MindSpore-MindArmour

MindArmour focus on security and privacy of artificial intelligence. MindArmour can be used as a tool box for MindSpore users to enhance model security and trustworthiness and protect privacy data. MindArmour contains three module: Adversarial Robustness Module, Fuzz Testing Module, Privacy Protection and Evaluation Module.

1.4.2.1 Adversarial Robustness Module

Adversarial robustness module is designed for evaluating the robustness of the model against adversarial examples, and provides model enhancement methods to enhance the model's ability to resist the adversarial attack and improve the model's robustness. This module includes four submodule: Adversarial Examples Generation, Adversarial Examples Detection, Model Defense and Evaluation.

The architecture is shown as follow：

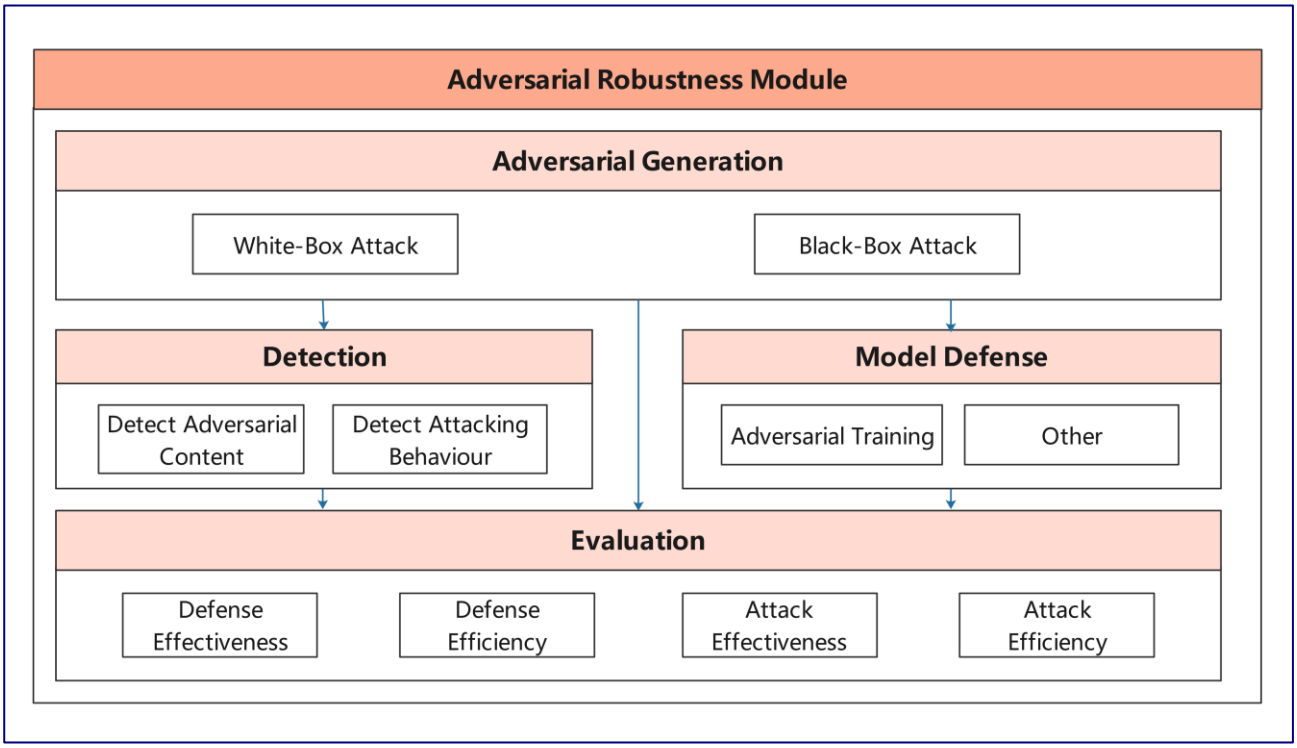


Figure 9 Adversarial Robustness overview

1.4.2.2 Fuzz Testing Module

Fuzz Testing module is a security test for AI models. We introduce neuron coverage gain as a guide to fuzz testing according to the characteristics of neural networks. Fuzz testing is guided to generate samples in the direction of increasing neuron coverage rate, so that the input can activate more neurons and neuron values have a wider distribution range to fully test neural networks and explore different types of model output results and wrong behaviors.

The architecture is shown as follow：

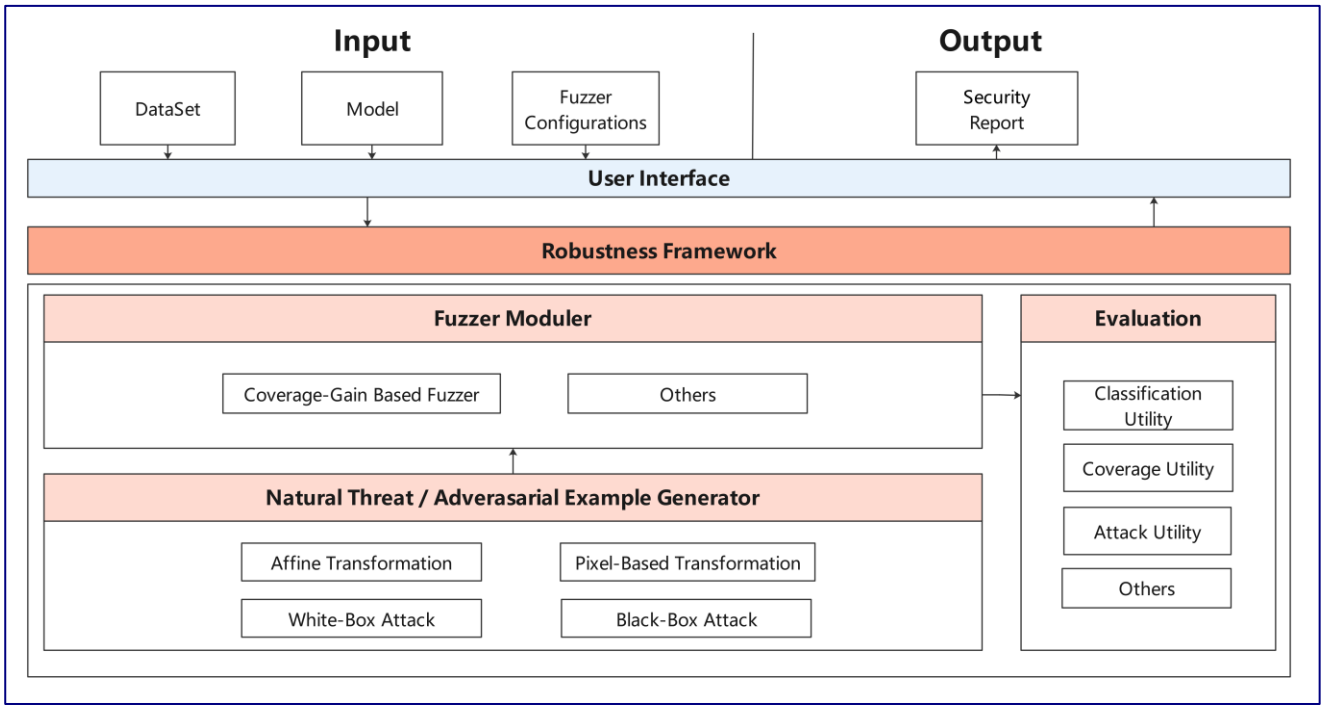


Figure 10 Fuzz testing overview

1.4.2.3 Privacy Protection and Evaluation Module

Privacy Protection and Evaluation Module includes two modules: Differential Privacy Training Module and Privacy Leakage Evaluation Module.

1.4.2.4 Differential Privacy Training Module

Differential Privacy Training Module implements the differential privacy optimizer. Currently, SGD, Momentum and Adam are supported. They are differential privacy optimizers based on the Gaussian mechanism. This mechanism supports both non-adaptive and adaptive policy. Rényi differential privacy (RDP) and Zero-Concentrated differential privacy（ZCDP） are provided to monitor differential privacy budgets.

The architecture is shown as follow：

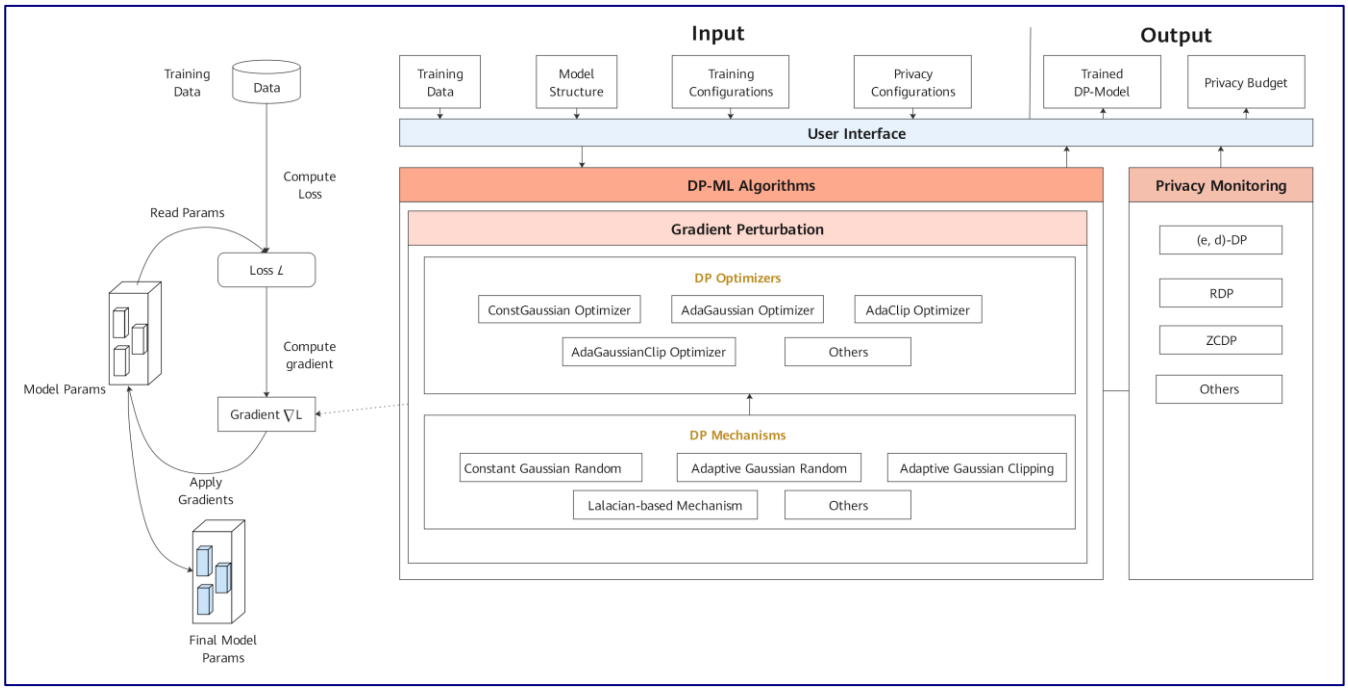


Figure 11 Differential Privacy overview

1.4.2.5 Privacy Leakage Evaluation Module

Privacy Leakage Evaluation Module is used to assess the risk of a model revealing user privacy. The privacy data security of the deep learning model is evaluated by using membership inference method to infer whether the sample belongs to training dataset.

The architecture is shown as follow：

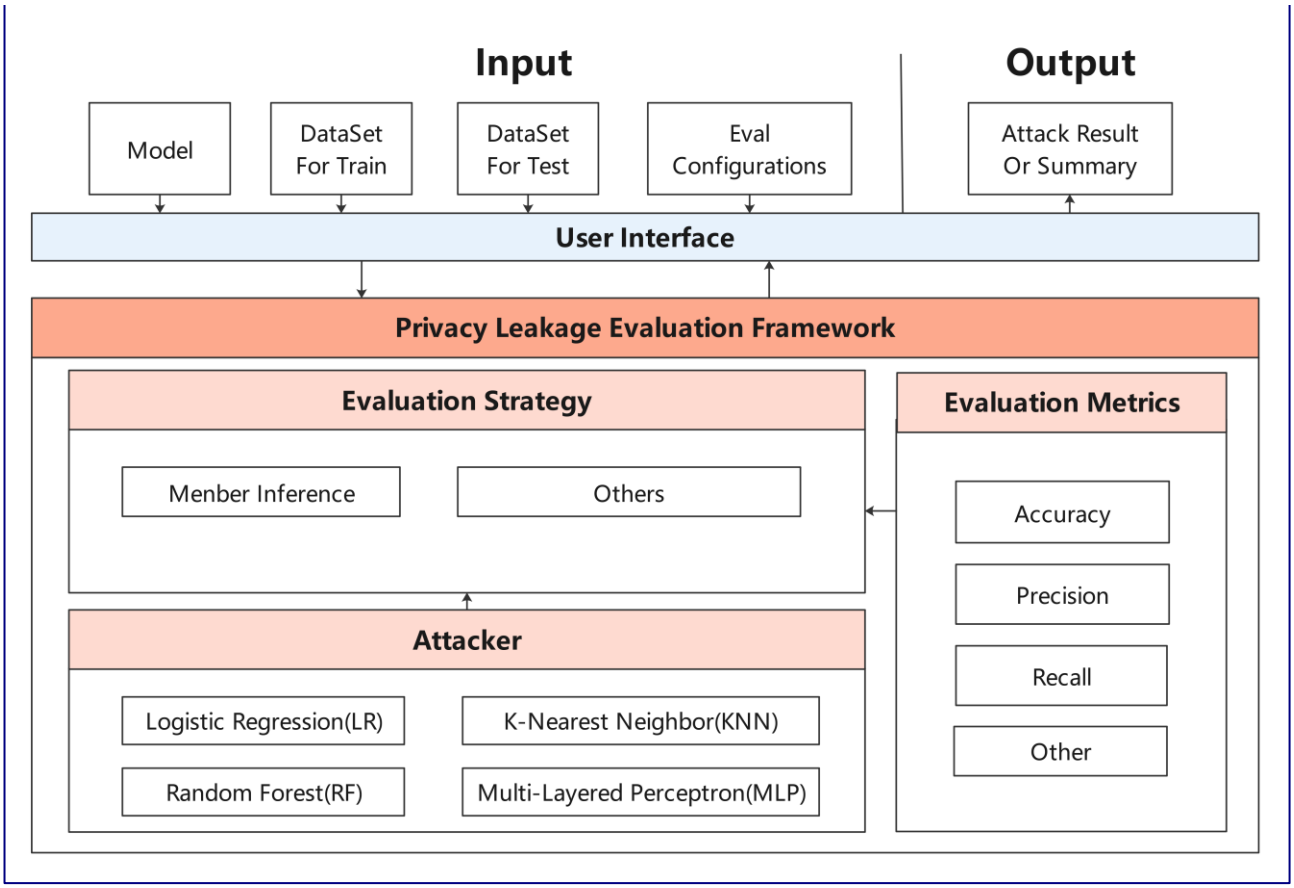


Figure 12 Privacy leakage overview

1.4.3 MindSpore-MindInsight

MindInsight provides MindSpore with easy-to-use debugging and tuning capabilities. During the training, data such as scalar, tensor, image, computational graph, model hyper parameter and training's execution time can be recorded in the file for viewing and analysis through the visual page of MindInsight.

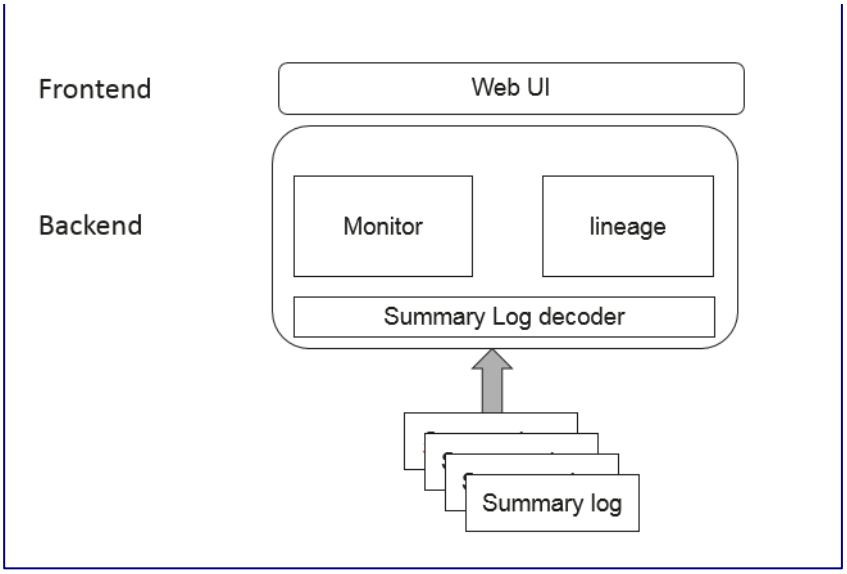


Figure 13 MindInsight overview

1.5 Physical Scope

The MindSpore is a software-only TOE, the physical scope of the TOE includes TOE Binary and TOE Guides.

1.5.1 TOE Binary

The MindSpore software package is released on MindSpore's official website (https://mindspore.cn/versions/en), which show the release list of all version, include MindSpore 1.9.0. The user can download the software, hash value, and related documents from the official website.

1.5.2 TOE Guides

The following product guidance documents are provided with the TOE.

Documents Name	Version	Type	Desc.
MindSpore_EAL4+_AGD_OPE_V0.2.1	0.2.1	Documents	User operation guidance
MindSpore_EAL4+_AGD_PRE_V0.2.1	0.2.1	Documents	Preparation procedure

You can also obtain the product interface document and user guide from MindSpore's official website at (https://www.mindspore.cn/docs/en/r1.9/index.html).

Documents Name	Version	Type	Desc.
MindSpore API References	1.9.0	Documents	https://mindspore-website.obs.cn-north-4.myhuaweicloud.com/pdf/r1.9/en/mindspore.pdf

MindArmour API References	1.9.1	Documents	https://mindspore-website.obs.cn-north-4.myhuaweicloud.com/pdf/r1.9/en/mindarmour.pdf
MindInsight API References	1.9.0	Documents	https://mindspore-website.obs.cn-north-4.myhuaweicloud.com/pdf/r1.9/en/mindinsight.pdf

2 Conformance claims

2.1 Common Criteria conformance claim

This Security Target conforms to CC Part 2 extended and Part 3 conformant, with a claimed Evaluation

Assurance Level of EAL 4, augmented by ALC FLR.2.

This Security Target claims conformance to the following specifications:

Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements, Version 3.1, Revision 5, April 2017.
Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements, Version 3.1, Revision 5, April 2017;

2.2 Protection Profile claim

This Security Target does not claim conformance to any Protection Profile.

3 Security problem definition

The security problem definition defines the security problem that is to be addressed in terms of threats, organization security policies and assumptions.

Note that, as the TOE is a framework, the AI Application developer may choose not to consider some of the threats as they may not be relevant to the generated model use case. Similarly, the AI Application developer may choose not to enforce some of the attack policies as the application developer may not be interested in attacking the model. The security functionality is still available in any case in the TOE.

3.1 Threats

T.EVASION_ATTACK: An attacker may maliciously manipulate a sample before it is presented to a protected computer vision model for inference in a way that would look the same to a human viewer but would be misclassified by the model. Depending on the type of the attack, the attacker may require knowledge of the model and its parameters.

3.2 Organizational Security Policies

P.EVASION_ATTACK: The TOE must be able to generate evasion attack samples targeting the original unprotected computer vision model.

3.3 Assumptions

A.TRUSTED_IT_ENVIRONMENT: The IT environment where the TOE is deployed is configured securely, preventing an attacker to get logical and physical access to the assets stored and transmitted within and between computation nodes.

A.TRUSTED_AI_APPLICATION: The AI Application model shall be properly use the MindArmour functionalities appropriate for its use case and to use a trusted training dataset. At inference time, the AI Application must also apply input format validation to the data before it is processed by the model.

A.RETRAINING: The AI Application developer does not use the model retraining functionality during inference time.

4 Security Objectives

The security objectives are a concise and abstract statement of the intended solution to the problem defined by the security problem definition. The security objectives show which security concerns are addressed by the TOE, and which security concerns are addressed by the environment.

4.1 Security Objectives for the TOE

Note that, as the TOE is a framework, the AI Application developer may choose not to consider some of the security objectives for the TOE as they may not be relevant to the generated model use case. The security functionality is still available in any case in the TOE.

O.ACCURACY_DEFENCE: The TOE shall provide a model with high image classification accuracy under the adversarial sample attacks. The TOE shall provide the possibility to retrain the model with different types of adversarial samples during model development.

O.ACCURACY_ATTACK: The TOE shall be able to generate adversarial samples that will be misclassified during the inference stage by the original unprotected image classification model.

4.2 Security Objectives for the Environment

OE. TRUSTED_IT_ENVIRONMENT: The IT environment where the TOE is deployed shall be configured securely, preventing an attacker to get access to the assets stored and transmitted within and between computation nodes.

OE.TRUSTED_AI_APPLICATION: The AI Application model shall be securely programmed, using the MindArmour functionalities appropriate for its' use case and to use a trusted training dataset. At inference time, the AI Application shall apply input format validation to the data before it is processed by the model.

OE.RETRAIN: The AI Application developer shall not use the model retraining functionalities at the inference stage.

4.3 Security Objectives rationale

The following tables provides a mapping of security objectives to threats, OSPs and assumptions.

Note that, as the TOE is a framework. When a SPD element is not chosen for a specific use case, its related security objectives can be ignored.

Table 1 Coverage of the objectives for the TOE to the SPD

SPD	Security objectives for the TOE	Coverage rationale
T.EVASION_ATTACK	O.ACCURACY_DEFENCE	Direct coverage
P.EVASION_ATTACK	O.ACCURACY_ATTACK	Direct coverage

Table 2 Coverage of the objectives for the Environment to the SPD

SPD	Security objectives for the environment	Coverage rationale
A.TRUSTED_IT_ENVIRONMENT	OE.TRUSTED_IT_ENVIRONMENT	Direct coverage
A.TRUSTED_AI_APPLICATION	OE.TRUSTED_AI_APPLICATION	Direct coverage
A.RETRAINING	OE.RETRAIN	Direct coverage

5 Extended Components Definition

This ST defines an extended SFR class with 2 Families with 2 components

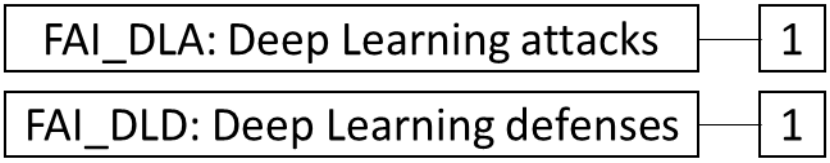
5.1 Class FAI: Artificial Intelligence

Artificial Intelligence technologies allow intelligent agent devices to perceive their environment and take actions that maximize their chance of successfully achieve their goals.

Artificial Intelligence methods vary in terms of approach and application and the families in this class can be applied to both individual agent devices and to methods that are used to generate individual intelligent agents by using specific generation tools and techniques.

Some families in this class consider TSFs that allow an agent user to attack with varying degrees of capabilities different types of Artificial Intelligence agents, with the goal of minimizing the agent chance of success. Other families in this class consider TSFs that allow an agent user to improve and harden the agent in order to withstand such attackers.

The FAI: Artificial Intelligence class is composed of two families: Deep Learning attacks (FAI_DLA) and Deep Learning defenses (FAI_DLD). The FAI_DLA family supports attack functionalities for an attacker user in a Deep Learning Artificial Intelligence type of agent. The FAI_DLD family supports functionalities for the hardening of the model defenses in a Deep Learning Artificial Intelligence type of agent.



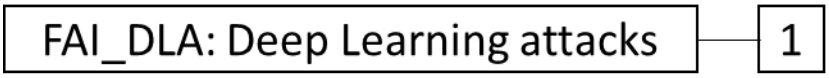
5.1.1 Deep Learning attacks (FAI_DLA)

5.1.1.1 Family Behavior

A user aiming to attack Deep Learning models will target a disruption of one or multiple of the agent’s goals. This goals may include the accuracy of the deep learning model inference, the privacy of the data used to train the model or the confidentiality of the model itself.

The attacker user may use the TOE to generate attacks targeting different types of deep learning networks with different usage scenarios. Note that the attack technique TSFs along with its prerequisites must be specified in the components.

5.1.1.2 Component levelling



FAI_DLA.1: Defines TSFs for an attacker user to target the inference accuracy of the model in a Deep Learning Artificial Intelligence type of agent, lowering the accuracy when applying inference to attacker provided samples.

5.1.1.3 Management

There are no management activities foreseen

5.1.1.4 Audit

There are no audit activities foreseen.

5.1.1.5 FAI_DLA.1 Deep learning accuracy attack

Hierarchical to: No other components

Dependencies: No dependencies

FAI_DLA.1.1 The TSF shall implement the inference accuracy attack **[assignment: attack technique]**, applicable to **[assignment: list of networks]** based models, when the following prerequisites are met: **[assignment: list of model knowledge and model access prerequisites]**.

FAI_DLA.1.2 The TSF shall ensure that the accuracy of the model decreases by **[assignment: accuracy decrease metric]**

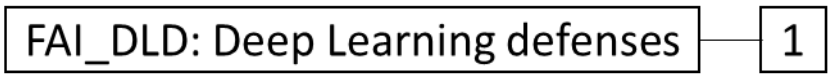
5.1.2 Deep Learning defenses (FAI_DLD)

5.1.2.1 Family Behavior

A user aiming to protect Deep Learning models will harden against the disruption of one or multiple of the agent's goals. This goals may include the accuracy of the deep learning model inference, the privacy of the data used to train the model or the confidentiality of the model itself.

The user may use the TOE to harden different types of deep learning networks with different usage scenarios. Note that the defense technique TSFs along with its prerequisites must be specified in the components.

5.1.2.2 Component levelling



FAI_DLD.1: Defines TSFs for a model developer user to improve the inference accuracy of the model in a Deep Learning Artificial Intelligence type of agent, increasing the accuracy when applying inference to attacker provided samples.

5.1.2.3 Management

There are no management activities foreseen

5.1.2.4 Audit

There are no audit activities foreseen.

5.1.2.3 FAI_DLD.1 Deep learning accuracy defense

Hierarchical to: No other components

Dependencies: No dependencies

FAI_DLD.1.1 The TSF shall implement the inference accuracy defense *[selection:**[assignment: defense technique]***, *any necessary technique]*, applicable to **[assignment: list of networks]** based models, when the following prerequisites are met: **[assignment: list of model knowledge and model access prerequisites]**.

FAI_DLD.1.2 The TSF shall ensure that inference accuracy is maintained by **[assignment: accuracy defense metrics for the augmented protected model]**

6 Security Requirements

6.1 Security Functional Requirements

This section describes the security functional requirements for the TOE. These requirements are the basis on which the TOE is evaluated.

The operations performed on the SFRs are identified as follows:

- Selection: *chosen selection*
- Assignment: **performed assignment**
- Refinement: Application note: details
- Iteration: / Iteration is added to the SFR identifier

6.1.1 FAI_DLA.1 Deep learning accuracy attack / White box

FAI_DLA.1.1 The TSF shall implement the inference accuracy attack **[assignment: see table],** applicable to **[assignment: see table]** based models, when the following prerequisites are met: **[assignment: trained model parameters are known and an approximation of the loss function is known].**

FAI_DLA.1.2 The TSF shall ensure that the accuracy of the model decreases b**y [assignment: see table]**

Note: The metrics are defined in terms of the effects in standardized datasets and networks. It is a common practice within academia and the industry for framework products.

Accuracy is used to evaluate whether the model in the adversarial samples after the attack.

Accuracy Threshold = x+(1-x)*0.2 where:

- X is the actual accuracy after attack.
- (1-X) *0.2 indicates are the tolerance of the current attack effect.
- x + (1-x) *0.2 indicates are the accuracy threshold for evaluating the effectiveness of attack.

Attack	Networks (Image classification)	Accuracy decrease metric

FastGradientSignMethod: I. J. Goodfellow, J. Shlens, C. Szegedy, "Explaining and Harnessing Adversarial Examples"	LeNet5 ResNet50	Prerequisites: Net: LeNet5 Dataset: MNIST Original Accuracy:0.9873 Eps=0.2 After attack: AA=Accuracy in the adversarial samples :0.3766 AT=Accuracy Threshold:0.5013 If under the same prerequisites, the test result of the AA <AT, the MindArmour attack is effective. Prerequisites: Net:ResNet50 Dataset:CIFAR10 Original Accuracy: 0.9493 Eps=8/255 After attack: AA=Accuracy in the adversarial samples : 0.1598 AT=Accuracy Threshold:0.32784 If under the same prerequisites ,the test result of the AA <AT, the MindArmour attack is effective
JSMA-Attack: N. Papernot, P. McDaniel, et al., "The Limitations of Deep Learning in Adversarial Settings"	LeNet5	Prerequisites: Net: LeNet Dataset: MNIST Original Accuracy:0.9873 Max_iteration=100 Theta=1.0 After attack: AA=Accuracy in the adversarial samples: 0.1450 AT=Accuracy Threshold: 0.3160 If under the same prerequisites, the test result of the AA < AT, the MindArmour attack is effective.

CarliniWagnerL2-Attack: N. Carlini, D. Wagner, et al., "Towards Evaluating the Robustness of Neural Networks"	- LeNet5 - ResNet50	Prerequisites: Net: LeNet5 Dataset: MNIST Original Accuracy:0.9873 learning_rate=0.005 After attack: AA=Accuracy in the adversarial samples:0.0000 AT=Accuracy Threshold:0.2000 Actual Accuracy < Accuracy Threshod If under the same prerequisites, the test result of the AA < AT, the MindArmour attack is effective. Prerequisites: Net:ResNet50 Dataset:CIFAR10 Original Accuracy: 0.9493 learning_rate=0.005 After attack: AA=Accuracy in the adversarial samples: 0.6438 AT=Accuracy Threshold: 0.7150 Actual Accuracy < Accuracy Threshold If under the same prerequisites, the test result of the AA < AT, the MindArmour attack is effective.

DeepFool Attack: S. Moosavi-Dezfooli, A. Fawzi, et al., "DeepFool: a simple and accurate method to fool deep neural networks"	- LeNet5 - ResNet50	Prerequisites: Net:LeNet5 Dataset:MNIST Original Accuracy:0.9873 overshoot=0.03 After attack: AA=Accuracy in the adversarial samples: 0.3081 AT=Accuracy Threshold: 0.4465 If under the same prerequisites ,the test result of the AA < AT, the MindArmour attack is effective. Prerequisites: Net:ResNet50 Dataset:CIFAR10 Original Accuracy: 0.9493 overshoot=0.03 After attack: AA=Accuracy in the adversarial samples: 0.2047 AT=Accuracy Threshold: 0.3638 If under the same prerequisites ,the test result of the AA < AT, the MindArmour attack is effective.

Projected Gradient Descent Attack: A. Madry, et al., "Towards deep learning models resistant to adversarial attacks"	-LeNet5 -ResNet50	Prerequisites: Net:LeNet5 Dataset:MNIST Original Accuracy:0.9873 eps=0.2 eps_iter=0.1 iter_num=5 After attack: AA=Accuracy in the adversarial samples: 0.1152 AT=Accuracy Threshold: 0.2921 If under the same prerequisites ,the test result of the AA < AT, the MindArmour attack is effective. Prerequisites: Net:ResNet50 Dataset:CIFAR10 Original Accuracy: 0.9493 eps=8/255 eps_iter=4/255 iter_num=5 After attack: AA=Accuracy in the adversarial samples: 0.0000 AT=Accuracy Threshold: 0.2000 If under the same prerequisites ,the test result of the AA < AT, the MindArmour attack is effective

6.1.2 FAI_DLA.1 Deep learning accuracy attack / Black box

FAI_DLA.1.1 The TSF shall implement the inference accuracy attack **[assignment: see table]**, applicable to **[assignment: see table]** based models, when the following prerequisites are met: **[assignment: the logits of samples can be predicted]**.

FAI_DLA.1.2 The TSF shall ensure that the accuracy of the model decreases by **[assignment: see table]**

Note: The metrics are defined in terms of the effects in standardized datasets and networks. It is a common practice within academia and the industry for framework products.

Accuracy is used to evaluate whether the model in the adversarial samples after the attack.

Accuracy Threshold = x+(1-x)*0.2 where:

- X is the actual accuracy after attack.

- (1-X) *0.2 indicates are the tolerance of the current attack effect.
- x + (1-x) *0.2 indicates are the accuracy threshold for evaluating the effectiveness of attack.

Attack	Networks (Image classification)	Accuracy decrease metric
Genetic Attack: M. Alzantot, Y. Sharma, et al., "GeneticAttack: Practical Black-box Attacks with Gradient-FreeOptimization"	LeNet5	Prerequisites: Net: LeNet5 Dataset: MNIST Original Accuracy:0.9873 step_size=0.3 max_steps=100 After attack: AA=Accuracy in the adversarial samples: 0.1068 AT=Accuracy Threshold: 0.2854 If under the same prerequisites, the test result of the AA < AT, the MindArmour attack is effective.
PSO Attack: R. Mosli, M. Wright, et al., "They Might NOT Be Giants: Crafting Black-Box Adversarial Examples with Fewer Queries Using Particle Swarm Optimization"	- LeNet5 - ResNet50	Prerequisites: Net: LeNet5 Dataset: MNIST Original Accuracy:0.9873 step_size=0.3 t_max =100 After attack: AA=Accuracy in the adversarial samples: 0.0786 AT=Accuracy Threshold: 0.2629 If under the same prerequisites, the test result of the AA < AT, the MindArmour attack is effective. Prerequisites: Net:ResNet50 Dataset:CIFAR10 Original Accuracy: 0.9493 step_size=0.5 t_max =100 After attack: AA=Accuracy in the adversarial samples: 0.2360 AT=Accuracy Threshold: 0.3888 If under the same prerequisites ,the test result of the AA < AT, the MindArmour attack is effective.
HopSkipJumpAttack: C. J, M. I. Jordan, et al., "HopSkipJumpAttack: A	LeNet5	Prerequisites: Net: LeNet5 Dataset: MNIST Original Accuracy:0.9873

Query-Efficient Decision-Based Attack"		gamma=1.0 After attack: AA=Accuracy in the adversarial samples: 0.1125 AT=Accuracy Threshold: 0.2900 If under the same prerequisites, the test result of the AA < AT, the MindArmour attack is effective.
Natural-evolutionary-strategy Attack: A. Ilyas, L. Engstrom, et al., "Black-box adversarial attacks with limited queries and information"	- LeNet5	Prerequisites: Net: LeNet5 Dataset: MNIST Original Accuracy:0.9873 max_queries=1000 scene=Label_Only top_k=5 epsilon=0.5 After attack: AA=Accuracy in the adversarial samples: 0.1144 AT=Accuracy Threshold: 0.2915 If under the same prerequisites, the test result of the AA < AT, the MindArmour attack is effective.
Pointwise Attack: L. Schott, J. Rauber, et al., "Towards the first adversarially robust neural network model on MNIST"	- LeNet - ResNet50	Prerequisites: Net:LeNet5 Dataset:MNIST Original Accuracy:0.9873 search_iter=15 max_iter=100 After attack: AA=Accuracy in the adversarial samples: 0.0728 AT=Accuracy Threshold: 0.2583 If under the same prerequisites ,the test result of the AA < AT, the MindArmour attack is effective. Prerequisites: Net:ResNet50 Dataset:CIFAR10 Original Accuracy: 0.9493 search_iter=15 max_iter=100 After attack: AA=Accuracy in the adversarial samples: 0.0813 AT=Accuracy Threshold: 0.2650 If under the same prerequisites ,the test result of the AA < AT, the MindArmour attack is effective.
Salt and pepper attack: It is a black box attack	- LeNet5 - ResNet50	Prerequisites: Net:LeNet5

algorithm, in which benign samples are added with salt and pepper noise to mislead the AI model. The added noise decreases with the number of iterations, so as to reduce the difference between adversarial samples and benign samples.		Dataset:MNIST Original Accuracy:0.9873 max_iter=100 After attack: AA=Accuracy in the adversarial samples: 0.0318 AT=Accuracy Threshold: 0.2254 If under the same prerequisites ,the test result of the AA < AT, the MindArmour attack is effective. Prerequisites: Net:ResNet50 Dataset:CIFAR10 Original Accuracy: 0.9493 max_iter=100 After attack: AA=Accuracy in the adversarial samples: 0.0962 AT=Accuracy Threshold: 0.2769 If under the same prerequisites ,the test result of the AA < AT, the MindArmour attack is effective

6.1.3 FAI_DLD.1 Deep learning accuracy defense

FAI_DLD.1.1 The TSF shall implement the inference accuracy defense *[selection:**[assignment: see list]**]*, applicable to **[assignment: see list]** based models, when the following prerequisites are met: **[assignment: the trained model and tagged samples are known]**.

FAI_DLD.1.2 The TSF shall ensure that inference accuracy is maintained by **[assignment: see list]**

Note: The metrics are defined in terms of the effects in standardized datasets and networks. It is a common practice within academia and the industry for framework products.

Note: Two accuracy metrics are provided to evaluate the defense effectiveness. One is the accuracy of the model in the original dataset after defense, and the other is the accuracy of the model in the adversarial sample after defense. A threshold is provided for both metrics that needs to be within a 80% tolerance factor of the calculated accuracy (i.e., if x is the accuracy after defense, 0.8 is the tolerance and x * 0.8 is the accuracy threshold).

Defense	Networks	Accuracy defense metric
natural adversarial defense: A. Kurakin, et al., "Adversarial machine learning at scale"	LeNet5 ResNet50	Prerequisites: Net: LeNet5 Dataset: MNIST Original Accuracy:0.9873 Eps=0.2 After attack with fgsm the accuracy of adversarial samples is 0.3766 Epoch:5 Batchsize:32

		After defense: AO=Accuracy of original samples after defense is : 0.9835 AT1=Accuracy Threshold of original samples after defense: 0.7868 AD=Accuracy of adversarial samples after defense is : 0.9351 AT2=Accuracy Threshold of adversarial samples after defense: 0.7481 If under the same prerequisites, the test result on the AO > AT1 && AD > AT2, MindArmour defense is effective. Prerequisites: Net:ResNet50 Dataset:CIFAR10 Original Accuracy: 0.9493 Eps=8/255 After attack with fgsm: Acc of adversarial samples is 0.1598 Epoch:5 Batchsize:32 After defense: AO=Accuracy of original samples after defense is : 0.8562 AT1=Accuracy Threshold of original samples after defense: 0.6845 AD=Accuracy of adversarial samples after defense is : 0.6041 AT2=Accuracy Threshold of adversarial samples after defense: 0.4833 If under the same prerequisites, the test result on the AO > AT1 && AD > AT2, MindArmour defense is effective.
Projected adversarial defense：A. Madry, et al., "Towards deep learning models resistant to adversarial attacks"	ResNet50 LeNet5	Prerequisites: Net:LeNet5 Dataset:MNIST Original Accuracy:0.9873 Eps=0.2 Eps_iter=0.1 Iter_num=5

		After attack with pgd the accuracy of adversarial samples is 0.1152 Epoch:5 Batchsize:32 After defense: AO=Accuracy of original samples after defense is : 0.9786 AT1=Accuracy Threshold of original samples after defense: 0.7829 AD=Accuracy of adversarial samples after defense is : 0.9393 AT2=Accuracy Threshold of adversarial samples after defense: 0.7514 If under the same prerequisites, the test result on the AO > AT1 && AD > AT2, MindArmour defense is effective. Prerequisites: Net:ResNet50 Dataset:CIFAR10 Original Accuracy: 0.9493 Eps=8/255 Eps_iter=4/255 Iter_num=5 After attack with pgd: Acc of adversarial samples is 0.0000 Epoch:5 Batchsize:32 After defense: AO=Accuracy of original samples after defense is : 0.8349 AT1=Accuracy Threshold of original samples after defense: 0.6679 AD=Accuracy of adversarial samples after defense is : 0.4355 AT2=Accuracy Threshold of adversarial samples after defense: 0.3484 If under the same prerequisites, the test result on the AO > AT1 && AD > AT2, MindArmour defense is effective.

6.2 Security assurance requirements

The set of security assurance requirements are those of EAL4 augmented with ALC_FLR.2.

6.3 Security requirements rationale

6.3.1 Security functional requirements rationale

Security functional requirements tracing table

Table 3 Security functional requirements dependencies

SFR	Dependencies	Rationale
FAI_DLA.1 / White box	No dependencies	Requirements met
FAI_DLA.1 / Black box	No dependencies	Requirements met
FAI_DLD.1	No dependencies	Requirements met

Table 4 Coverage of the objectives for the SFR to the SOT

Security Objectives for the TOE	Security Functional Requirements	Coverage rationale
O.ACCURACY_ATTACK	FAI_DLA.1 / White box	Direct coverage
O.ACCURACY_ATTACK	FAI_DLA.1 / Black box	Direct coverage
O.ACCURACY_DEFENCE	FAI_DLD.1	Direct coverage

6.3.2 Security assurance requirements rationale

The chosen SARs are the ones of EAL4 augmented with ALC_FLR.2. This set is chosen because it is internally consistent and provides an appropriate level of assurance for a deep learning framework that will be used by a security-aware application developer.

7 TOE summary specification

The TSFs trace back to the SFRs as follows:

Table 5 SFR to TSF tracing

	TSF.ModelDefense	TSF.ModelAttack
FAI_DLA.1 / White box		X
FAI_DLA.1 / Black box		X
FAI_DLD.1	X	

7.1 TSF.ModelAttack

The TOE implements the accuracy model attack techniques mentioned in the FAI_DLA.1 / White box and FAI_DLA.1 / Black box SFRs. Such attack techniques are implemented in the MindArmour component.

Specifically, the deep learning accuracy attack techniques are implemented in the Adversarial Examples Generation Module. The module slightly modifies the sample input so that the AI model cannot correctly identify or process the sample input, but a human can still correctly judge the sample input.

Additionally, the Model Defense and Evaluation module can be used to support the choice of attack mechanisms in a white box attack scenario where the goal is to later on strengthen the model.

7.2 TSF.ModelDefense

The TOE implements the accuracy defense techniques mentioned in the FAI_DLD.1 SFRs. Such defense techniques are implemented in the MindArmour component.

Specifically, the deep learning accuracy defense techniques are implemented in the Adversarial Examples Detection Module. The module mixes some small disturbances in the sample, and then makes the neural network adapt to the change, resulting in stronger robustness to the adversarial samples.

Additionally, the Model Defense and Evaluation module can be used to support the choice of defense mechanisms.

8 Glossary of terms

AI: Artificial Intelligence

CPU: Central Processing Unit

GPU: Graphics Processing Unit

ST: Security Target

TOE: Target of Evaluation

9 References

[CC1] Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017. Part 1: Introduction and general model.

[CC2] Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017. Part 2: Security functional components.

[CC3] Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017. Part 3: Security assurance components.

