Machine Learning System Security Guidelines, Part I. “Security Measures Procedures”

Version 1.03a
March 15, 2023

Editing Committee of Machine Learning System Security Guidelines
Security Working Group on Machine Learning System

Machine Learning Systems Engineering (MLSE)
Japan Society for Software Science and Technology

Contents

I-1. Summary

I-1.1. Overview

This document organizes security procedures against machine learning-specific attacks (MLSAs) for developers and service providers of machine learning systems. The purpose of this document is to clarify effective countermeasures according to the situation and to help developers and service providers communicate with machine learning security experts when implementing security measures. This guideline is a self-assessment tool and does not necessarily need to be followed. Any damage resulting from the implementation of the contents of this document is not the responsibility of the authors.

These guidelines comprise “Security Measures Procedures,” “Risk Assessment,” and “Overview of Detection Techniques for Machine Learning-Specific Attacks” (Figure I- 1).

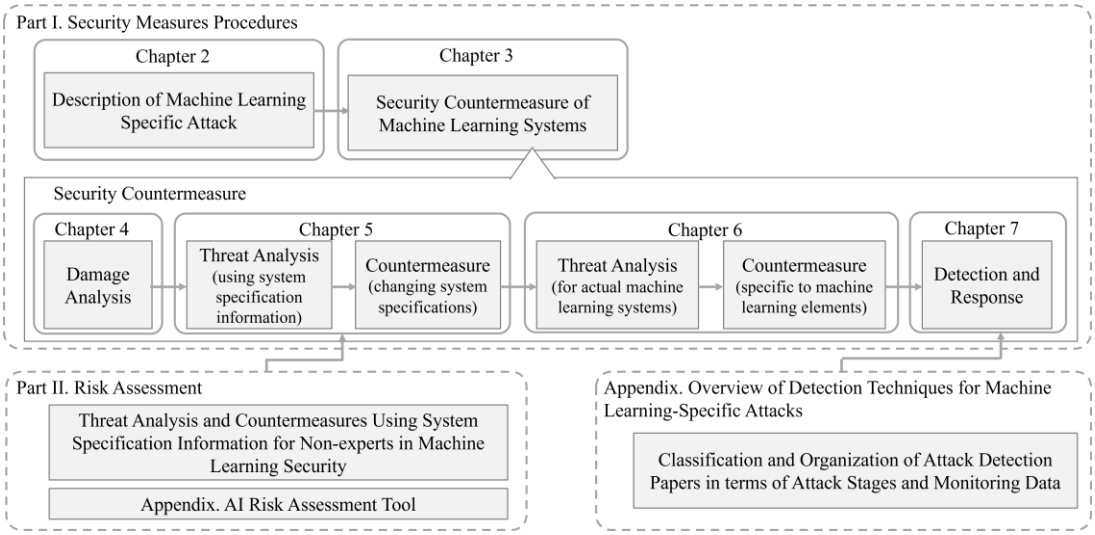

Figure I- 1. Overview of the Guidelines

In Part I, the security measures procedures, the procedures of security countermeasures against MLSAs are introduced. Chapter I-2 describes the MLSAs, and Chapter I-3 provides an overview of security countermeasures against MLSAs. Chapters I-4 to I-7 describe the specific implementation of each procedure introduced in Chapter I-3.

In Part II, the risk assessment, methods for system developers who may not have expertise in machine learning security to assess “threat analysis and countermeasures using system specification information” described in Chapter I-5 of Part I are introduced in detail.

The appendix, overview of detection techniques for machine learning-specific attacks, surveys papers related to detection technology against MLSAs, which are described in Chapter I-7 and classified and summarized in terms of “categories of attacks,” “state of attack detection,” and “data to be monitored.” This document can be used as a reference when building a detection system.

I-1.2. Purposes and Background

Recently, with the development and widespread use of machine learning, many previously unobtainable machine learning-based systems with advanced functions, such as image recognition and natural language processing, have been developed (hereinafter called “AI systems”). In the future, such systems are expected to replace humans, such as self-driving cars and automated financial transactions. However, when they become an integral part of the social infrastructure, security risks increase.

Security assessments and countermeasures have been conducted even in conventional systems that do not use machine learning. However, machine learning has specific vulnerabilities, such as adversarial examples that intentionally malfunction train models; thus, security analysis and countermeasures specific to machine learning systems are required.

Since the discovery of the adversarial examples in 2014, extensive research has been conducted on MLSAs and defensive techniques against them. Although some papers collect research trends, the assumptions and evaluation metrics for the attacks and defenses proposed in each study have not been defined. Therefore, it is unclear how to determine what attacks are applicable and what countermeasures are necessary for system development. Furthermore, to analyze security specific to machine learning systems and implement countermeasures, special knowledge of machine learning and security is required. This requirement makes security measures difficult to implement in development workplaces where few machine learning security experts with knowledge in both areas are available.

Therefore, AI developers who may not know about information security need criteria (guidelines) to judge whether MLSAs will be applicable in their AI systems. These guidelines introduce typical attack methods and enable AI developers to judge whether the conditions required for those attacks to be executed on a developed AI system are satisfied. They will be able to determine whether they should work with machine learning security experts according to the results of their analysis of the likelihood of such attacks.

In organizing these guidelines, a Formulation Committee of Machine Learning System Security Guidelines was established in July 2021 within the Safety and Security Working Group on Machine Learning System of the Machine Learning Systems Engineering Research Group (MLSE). This committee comprises members from industry and academia selected from the public by this study group and aims to provide useful information for AI developers as guidelines. These guidelines summarize the results of the discussions and deliberations of this formulation committee.

Some governments and international organizations have published guidelines on developing and using AI. In 2019, the OECD released the multi-country agreed-upon AI Principles [I-1], which include inclusive growth, sustainable development and well-being, human-centered values and equity, transparency and accountability, robustness, security and safety, and accountability. In Japan, the

“Human-centered AI Society Principles” [I-2] set forth the principles of human-centeredness, education and literacy, ensuring privacy, ensuring security, ensuring fair competition, fairness, accountability and transparency, and innovation.

Although the various elements that comprise the AI Principles are grouped in different ways, they fall into eight themes: privacy, accountability, safety and security, transparency and accountability, fairness and non-discrimination, human control of technology, professional responsibility, and promotion of human values.

To put the above principles into practice, METI's “The State of AI Governance in Japan” [I-3] organized the structure of AI governance as shown in Figure I- 2.

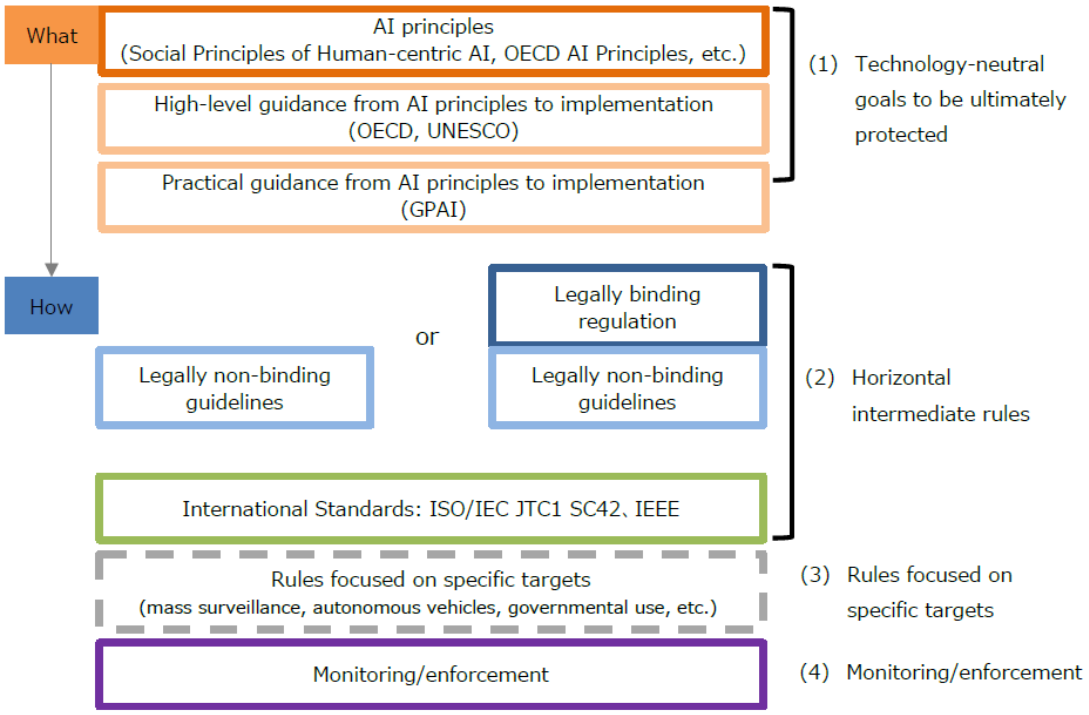

Figure I- 2. The structure of AI governance

These guidelines correspond to the “legally non-binding guidelines” shown in Figure I- 2. Additionally, there are an effort to emphasize “security” as addressed in the AI Principles.

I-1.3. Scope of the Guidelines

This section describes the intended audience for these guidelines, how the guidelines will be used, and their relationship to related literature.

I-1.3.1. Expected Target of the Guidelines

The intended audience for these guidelines includes developers of machine learning systems and service providers using machine learning systems who may not have expertise in machine learning

security. In this section, what machine learning security experts should do about the security measures for machine learning systems and what non-experts should do are described separately.

Regarding the AI Utilization Guidelines of the Ministry of Internal Affairs and Communications [I-4], the flow of AI utilization is shown in Figure I- 3.

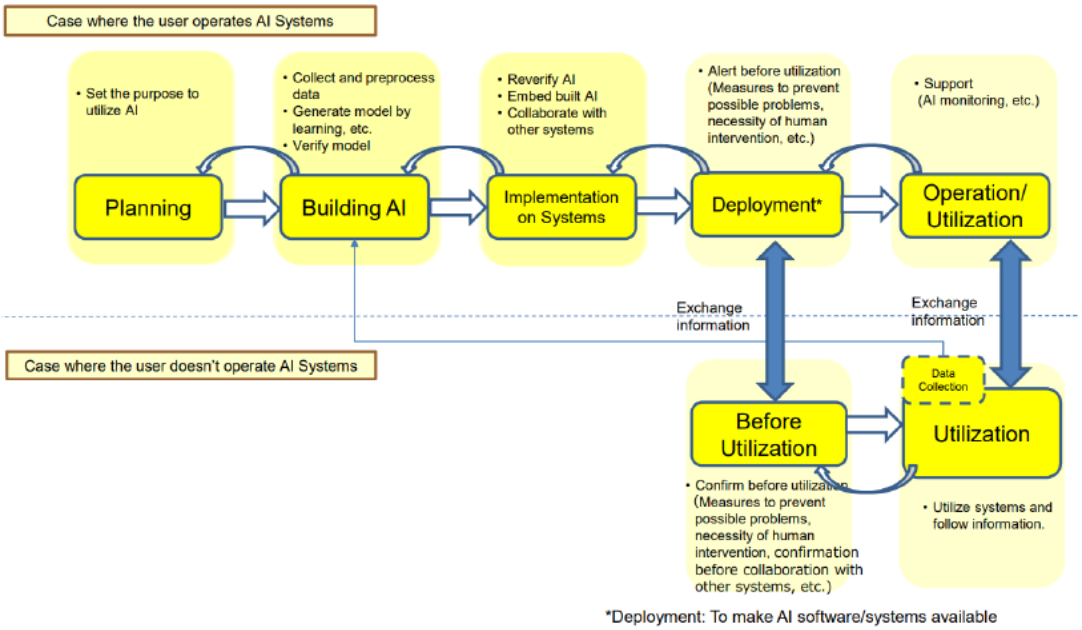

Figure I- 3. General flow of AI utilization for each entity

This document is intended for use in “planning,” “building AI,” and “operation/utilization.”

I-1.3.2. Relationship to Other Machine Learning Security Countermeasure Literature

Various studies on security countermeasures against MLSAs have been published by various government agencies and companies.

The ATLAS [I-5] published by MITRE organizes the tactics and methods of MLSAs. This report summarizes the methods used at each stage of an attack; such as the reconnaissance phase, initial access phase, attack execution phase, and information theft phase. It also summarizes actual attack examples as case studies.

For risk assessment of machine learning systems, ENISA's Artificial Intelligence Cybersecurity Challenges [I-6] discusses assets and damage associated with machine learning systems. Microsoft’s Threat Modeling AI/ML Systems and Dependencies [I-6] and ICO’s AI and data protection risk mitigation and management toolkit [I-8] also mention precautions to be taken when developing AI systems, and the documents help to connect the development system to a threat.

MLSAs and defenses against them have been published by organizations such as NIST [I-9], AIST [I-10], ENISA [I-6], and ICO [I-8], and companies such as Microsoft [I-6] and Mitsui Bussan Secure

Directions [I-11].

On the other hand, a series of security countermeasures in the existing security countermeasure literature for MLSAs had no systematic organization. Therefore, system developers and service providers implement security measures with difficulty.

Under these guidelines, security procedures are explained to developers and service providers of machine learning systems to help them understand what they need to do when implementing security measures, and to help them communicate with machine learning security experts when implementing security measures.

Additionally, to facilitate security measure performance, a method that enables “threat analysis and countermeasures using system specification information” in the security measures procedures to be performed even without expertise in machine learning security is introduced.

I-1.3.3. Relationship to General Information Security

This section describes the relationship between security against MLSAs and general information security. In these guidelines, “MLSAs” are defined as attacks that use machine learning system characteristics and have access to legitimate authority. For example, an attack that duplicates the model of the target machine learning system by repeatedly inputting data into the system and observing the output is included in the MLSAs. Security measures should be considered for general information security and security against MLSAs. It is desirable to consider countermeasures by combining the priorities of general information security and security against MLSAs, referring to the damage caused by attacks and the likelihood of attacks occurring. The ISO 27000 series, ISO/IEC 15408, NIST SP 800 series, NIST Cyber Security Framework, and the Information Technology Promotion Agency, Japan guidelines provide frameworks for general information security risk assessment and countermeasures.

I-1.4. Machine Learning Systems Used in These Guidelines

The target system in Part I is a system using machine learning. The machine learning processing part of a machine learning system generally comprises a training pipeline and an inference pipeline, as shown in Figure I- 4 and Figure I- 5. Some systems have only an inference pipeline because the training pipeline is performed externally. Before the machine learning system is operated, the training pipeline performs training processing using training-related data and generates trained models. Then, the inference pipeline performs inference processing using the inference target data and the trained models to obtain inference results.

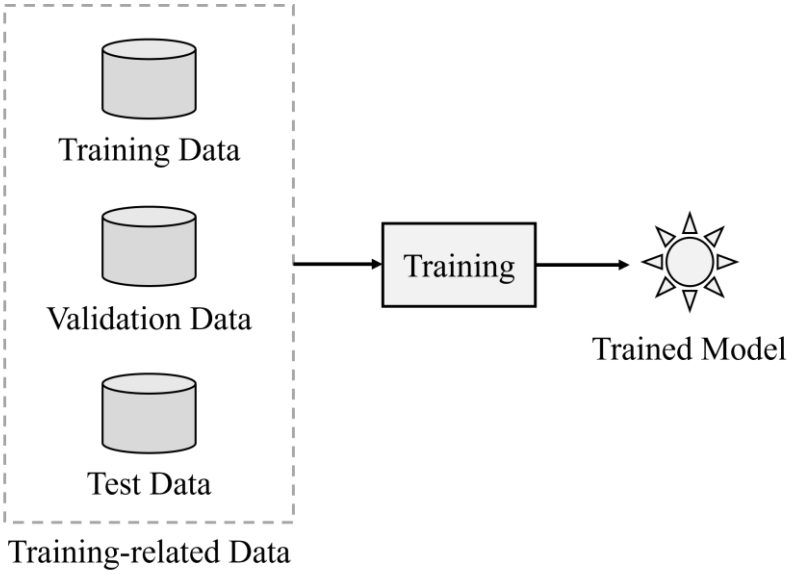

Figure I- 4. Training pipeline of a machine learning processing part

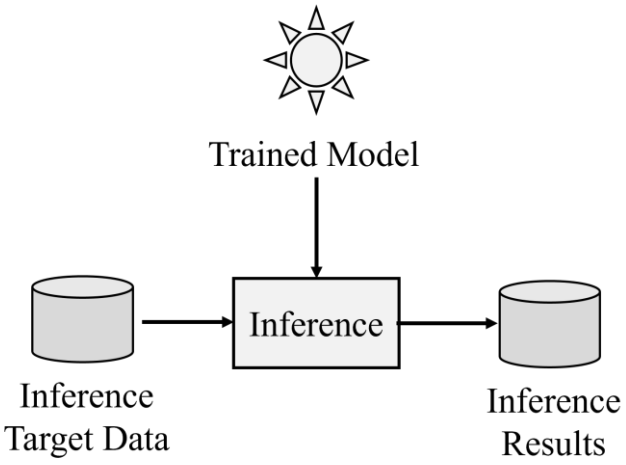

Figure I- 5. Inference pipeline of a machine learning processing part

I-2. Machine Learning Specific Attacks (MLSAs)

This chapter describes the threats posed by MLSAs and the specific attacks that pose threats and concludes with basic ideas for security measures.

MLSAs can be found in NIST’s NIST IR 8269 Draft [I-9], AIST’s Machine Learning Quality Management Guidelines [I-10], and Microsoft’s Threat Modeling AI/ML Systems and Dependencies [I-6].

I-2.1. Threats Specific to Machine Learning Systems

Threats specific to machine learning systems can be broadly classified into three categories. These threats can be caused by authorized access to the machine learning system.

- Model or system malfunctions
- Model theft
- Training data theft

I-2.1.1. Model or System Malfunctions

This threat is the malfunctioning of machine learning system models, which prevents the correct system operation. The malfunction of models and systems can lead to damage such as induced accidents in automatic driving systems (e.g., misidentification of sign recognition systems) and evading malware detection.

Two types of attacks can cause a model or system to malfunction: attacks that are performed by input to the model/system during inference (Section I-2.2.1) and attacks that contaminate training data and training models (Section I-2.2.2). In addition, attacks can not only intentionally change the decision results but also impair only the explanatory function of the model or the system. This result can affect the transparency of the system.

I-2.1.2. Model Theft

This attack is the theft of a model of a machine learning system by an attacker who creates a copy of the model or a model with similar performance. Model theft can lead not only to direct damage to Intellectual Property (IP), such as replicating services, but also to other attacks that use the thieved models.

One type of model theft attack is based on inputs to models and systems (Section I-2.2.3).

I-2.1.3. Training Data Theft

This threat is the theft by an attacker of data used to train models for machine learning systems or the theft of some of the information in the data used for training. This action can lead to privacy protection damage, such as personally identifiable information (PII) leakage.

One type of training data theft attack is based on inputs to models and systems (Section I-2.2.4, Section I-2.2.5).

I-2.2. Threat-causing Attacks

Five typical MLSAs cause the threats listed in Section I-2.1.

I-2.2.1. Evasion Attacks

Evasion attacks cause models and systems to malfunction (Section I-2.1.1).

Maliciously modifying the input data of a machine learning system can induce the system to behave in a way that was not intended. A well-known attack is called an adversarial example. This attack induces model misjudgments by adding a small amount of noise to the input data that are invisible to humans.

I-2.2.2. Poisoning Attacks

Poisoning attacks cause models and systems to malfunction (Section I-2.1.1).

An attacker can cause malfunctions by inserting crafted data and models into data and models used to train models of machine learning systems. This attack not only causes one label to be misjudged by another label, but also includes a backdoor attack that causes a given label to be misjudged by inputting data containing a specific pattern called a trigger.

I-2.2.3. Model Extraction Attacks

Model extraction attacks cause model theft (Section I-2.1.2).

This attack creates a model that has the same performance as the target system's model by analyzing the input-output relationship with the machine learning system.

I-2.2.4. Model Inversion Attacks

Model inversion attacks cause training data theft (Section I-2.1.3).

This attack reconstructs information contained in training data by analyzing the relationship between input and output data on machine learning systems.

I-2.2.5. Membership Inference Attacks

Membership inference attacks cause training data theft (Section I-2.1.3).

This attack identifies whether specific data are included in the training data of a model by analyzing the input and output data to the machine learning system. Unlike model inversion attacks, it does not reconstruct the training data.

I-3. Security of Machine Learning Systems

This chapter describes the procedures for security measures against MLSAs. In machine learning systems, security measures should be taken against MLSAs, in addition to conventional general information security. It is desirable to consider the priority of general information security and security against MLSAs, referring to the amount of damage and the possibility of attacks.

Notably, comprehensively understanding and taking countermeasures against all attacks is usually impossible.

I-3.1. Concept of Machine Learning Security

Machine learning security is considered in the same process as general information security measures, where the possibility of attacks is identified through “risk assessment” and then “countermeasures” are taken based on the results of that analysis. First, the impact of the threat is analyzed by identifying the assets related to the system and linking them to the threats that affect the assets (Section I-2.1. Next, threat analysis determines whether the system has vulnerabilities that allow attacks leading to threats (Section I-2.2).

Figure I- 6 shows the overall picture of the relationship between assets, threats, and attacks, damage analysis, threat analysis, and countermeasures in machine learning.

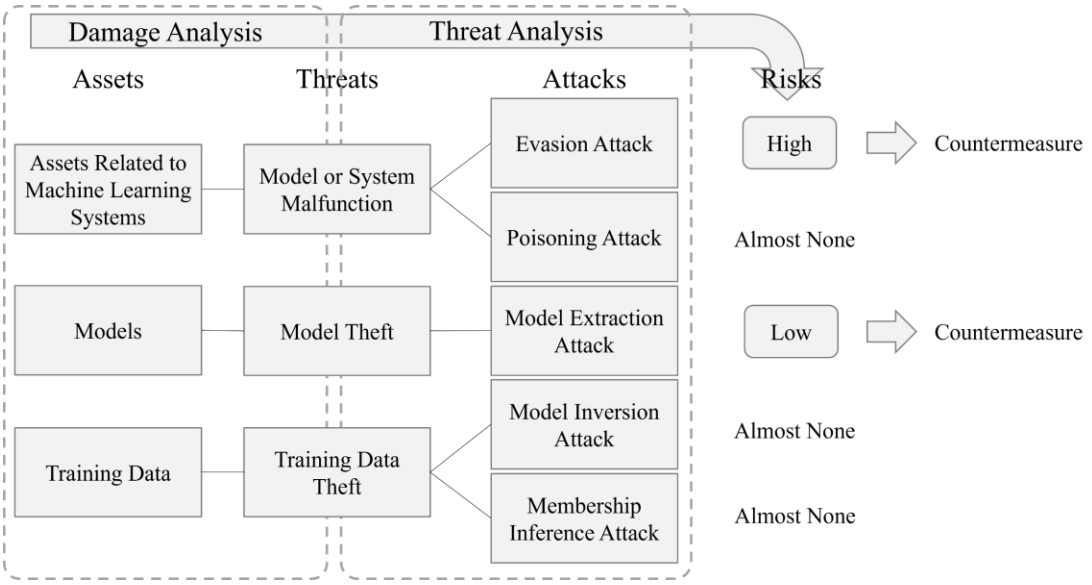

Figure I- 6. Overall picture of machine learning security

Countermeasures are divided into two major categories: “mitigation measures” to make attacks difficult or suppress their effects before operation/utilization, and “detection and response” to detect the occurrence of attacks during operation and take action. Basically, measures should be implemented through “mitigation measures” before operation as much as possible. “Detection and response” are performed for threats that cannot be prevented by mitigation measures, threats that require more focused countermeasures, and unknown threats that cannot be identified in advance.

I-3.2. Machine Learning Security Processes

Figure I- 7 shows an example of how to proceed with each process in the previous section.

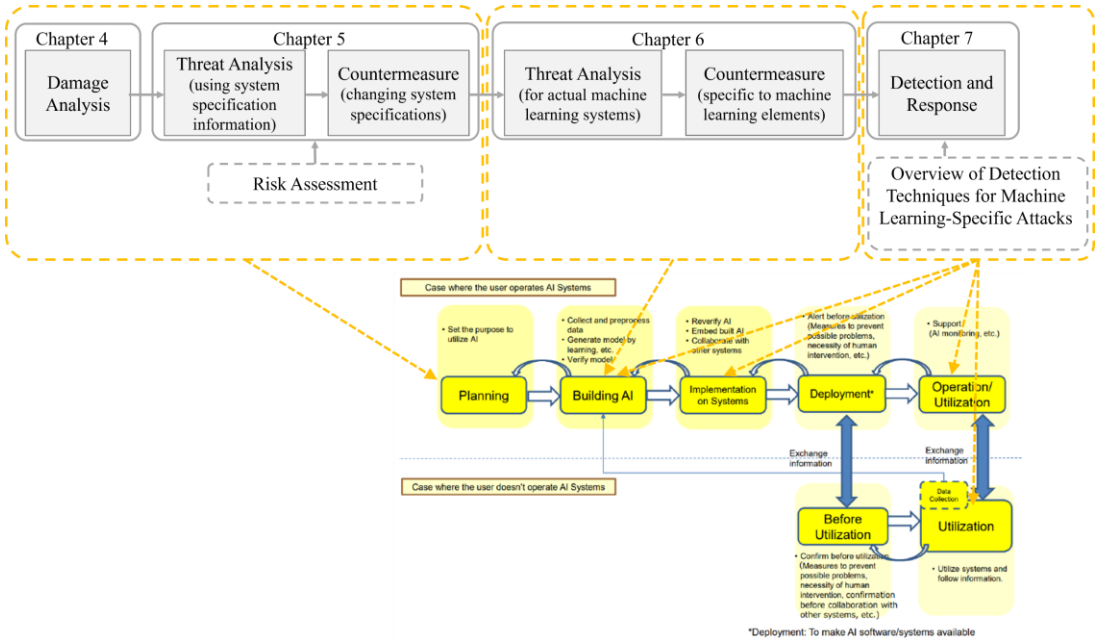

Figure I- 7. Machine learning security processes

First, “damage analysis” should be conducted, followed by “threat analysis and countermeasures using system specification information.” If threats are not sufficiently removed at this stage, “threat analysis and countermeasures for actual machine learning systems” should be conducted. Then, based on the results of the countermeasures, “detection and response” are designed and implemented as security measures for areas where the countermeasures are insufficient. Generally, “detection and response” are implemented during the operation/utilization of the system. However, in some cases, such as countermeasures against contamination of contaminated data and models, they are implemented at the stage of building AI and implementation on systems. In such cases, a system for “detection and response” must be established at the time of “building AI” in the “planning” stage.

In this document, the “Risk Assessment” part describes methods for system developers who do not have expertise in machine learning security to analyze threats in the “Threat Analysis and Countermeasures Using System Specification Information” process described above. In addition, “Overview of Detection Techniques for Machine Learning-Specific Attacks” introduces detection technology papers on MLSAs, which are categorized and organized with “target attack” and “its attack stage” in the above “detection and response” process.

I-3.3. Implementation of Machine Learning Security Processes

The implementation of each process described in the previous section requires the cooperation of two types of personnel, one responsible for actual analysis and countermeasures and the other responsible

for deciding on countermeasures and detection in each process.

The person who performs the detection and countermeasures will basically be the system developer. However, in the case of “detection and response,” preliminary preparations such as log design are done in the “planning” and “implementation on systems” phases, whereas actual detection and response are performed in the “operation/utilization” phase. The former phase is often performed by the system developer and the latter phase by the service provider, so communication between these two is necessary. In addition, since expertise in machine learning security is essential implementing each phase, cooperation with experts is also necessary.

The responsible party is the person who takes responsibility for the risk when it materializes, such as the requester of the system development or the service provider. The results of the damage analysis and threat analysis presented by the implementer are used to determine the threats to be addressed, their priority, and constraints.

I-4. Damage Analysis

This chapter describes the damage analysis for MLSAs. The first step in damage analysis is to identify the assets to be protected related to the machine learning system. Then, the identified assets are linked to the threats posed by MLSAs, and the expected damage is calculated.

I-4.1. Identification of Assets That Should Be Protected

Assets related to the target machine learning system should be identified. Two major types of assets are related to machine learning systems: “assets that constitute machine learning” such as models and training data, and “assets related to machine learning systems” that are affected by the output results of the models.

Examples of “assets related to machine learning systems” for each system are listed below.

- Sign recognition systems ... Automatic driving
- X-ray diagnostic imaging systems ... Medical judgment
- Face recognition gates ... Security of gate installation location
- SNS image filters ... SNS policy
- Malware detection and terminals ...Information security of installed organizations

Assets associated with machine learning systems are also discussed in ENISA's Artificial Intelligence Cybersecurity Challenges [I-6]. Mainly “assets comprising machine learning” are classified and enumerated into six categories, such as data, models, and stakeholders. This document can be used as a reference to identify assets that should be protected.

I-4.2. Organization of Related Entities

Organize the stakeholders connected to these assets and identify these entities affected by the threats. For example, in the case of training data, the data provider is a stakeholder. Notably, “assets related to machine learning systems” may affect parties other than the system users. In the case of an X-ray image diagnosis system, the direct user of the system is the doctor, but the doctor and the patient to be diagnosed are affected by the judgment result of the system.

For reference, Figure I- 8 shows the entities involved in using AI according to the AI Utilization Guidelines [I-4].

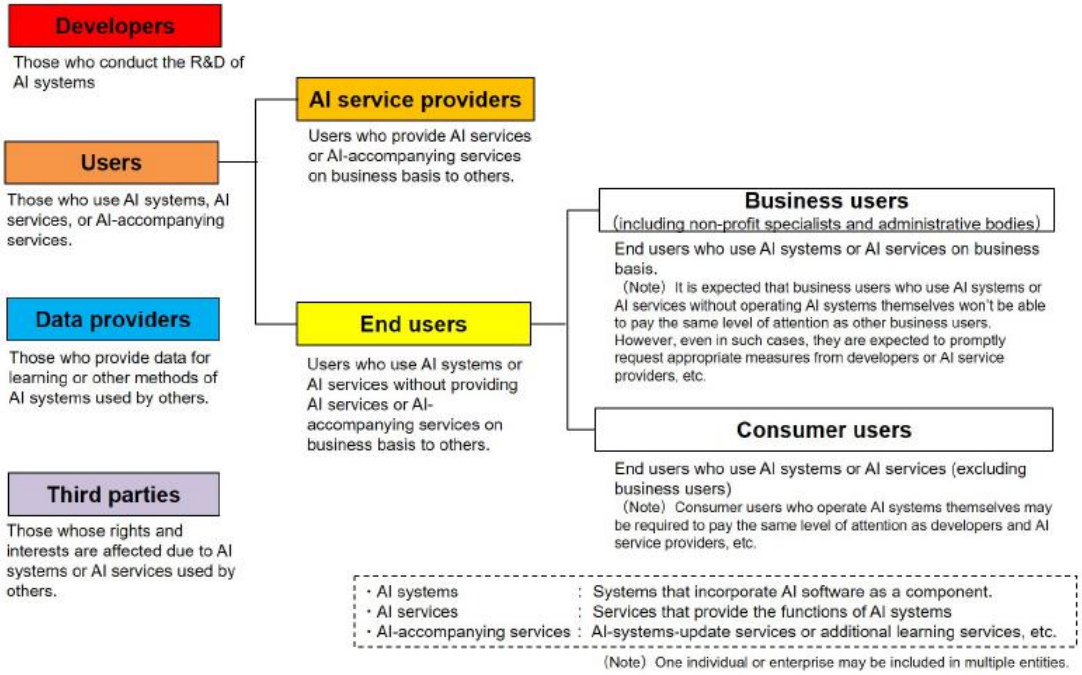

Figure I- 8. Classification of related entities in the use of AI

I-4.3. Damage Analysis of Threats Specific to Machine Learning Systems

The damage analysis identifies the possible damage by linking the assets and related entities identified in the work up to the previous section with the threats shown in Section I-2.1. An example of the linkages is shown in Figure I- 9.

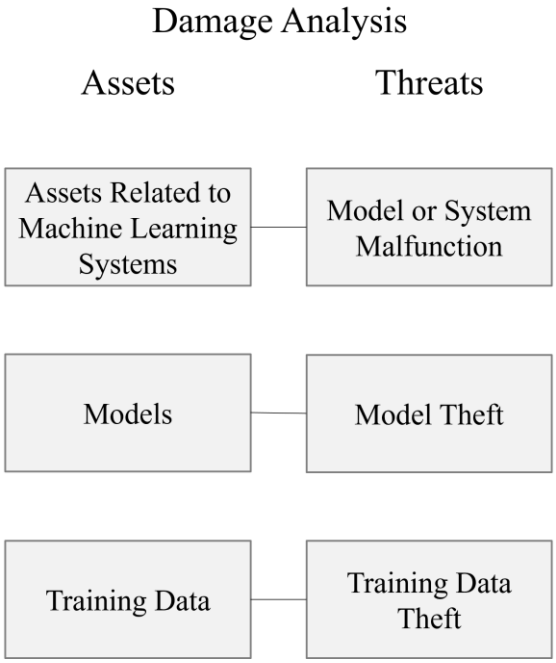

Figure I- 9. Examples of linking assets and threats

To link “assets related to machine learning systems” to “model or system malfunction,” specific threat scenarios should be envisioned. Examples are listed below.

- Misjudging sign recognition system causes accidents in automated vehicles
- Misjudging X-ray image judgment systems causes medical errors
- Misjudging the facial recognition gate results allows non-registered persons to pass through the gate
- Avoiding social networking service's image filtering causes the uploading and publishing of images prohibited by terms and conditions
- Avoiding malware detection allows the target to download malware

AIST's Machine Learning Quality Management Guidelines [I-10] defines seven AI safety levels by dividing risk avoidance into human risks such as injury to humans and other economic risks as damage analysis. Please refer to that document as well.

I-5. Threat Analysis and Countermeasures Using System Specification Information

This chapter describes threat analysis and countermeasures from the specification information of machine learning systems. This analysis and these countermeasures are assumed to be conducted in the “planning” phase (Figure I- 3) in the flow of AI utilization in the Guidelines for AI Utilization [I-4] of the Ministry of Internal Affairs and Communications.

On the basis of the results of the damage analysis in Chapter I-4, threat analysis is conducted for attacks that cannot be overlooked based on the expected damage. By analyzing and taking countermeasures from specification information, the target attacks can be narrowed down and the analysis and countermeasures can be omitted at the time of “building AI” and the “implementation on systems” phase.

The specific implementation of this chapter is summarized in the “Risk Assessment” part as a method that enables system developers and service providers who do not have expertise in machine learning security.

I-5.1. Threat Analysis Using System Specification Information

The target machine learning system is analyzed to determine whether it is vulnerable to attacks that cause threats specific to machine learning systems (Section I-2.2) because of the system specifications. First, a potential attacker is established (Section I-5.1.1) from the entities (Section I-4.2) related to the target machine learning system. Next, the capabilities of the potential attacker are analyzed (Section I-5.1.2) to determine whether they satisfy the conditions for the target attack to be successful. Each process is described below.

I-5.1.1. Setting Potential Attackers

First, the potential attacker is set from the entities related to the target machine learning system (Section I-4.2). Potential attackers are the entities related to the target machine learning system that may become attackers. Each party has different privileges in the target machine learning system. For example, system operators can perform training processes, but system users can only perform inference processes. Therefore, multiple attackers are set up, and each attacker is analyzed to determine if it satisfies the attack success conditions described in Section I-5.1.2.

There are four potential attackers: “system administrators and operators (developers and service providers),” “training data providers,” “system users,” and “third parties that do not directly use the system.” For example, in the case of an X-ray diagnostic imaging system, the following attacker candidates are considered: “system administrator,” “provider of training X-ray image data,” “doctor (user of the system),” and “patient (a third party who does not directly use the system).” Figure I- 8 lists entities that are assumed to be involved using AI [I-4], which can be used as a reference when comprehensively identifying potential attackers.

I-5.1.2. Analysis of Conditions for Attack Formation

Next, the capability of the potential attacker is analyzed regarding whether attacks satisfy the conditions to be successful.

For the five attacks listed in Section I-2.2, the conditions for an attack to be successful are basically the number of inferences and the number of responses of inference results made by the machine learning system. In addition, for poisoning attacks, the training process and the degree of intervention in the training data and model are additional conditions. Factors that mitigate the conditions for the success of each attack include the content of the inference results, the ease of obtaining training data and similar data, and public information about the system.

The specific training and inference conditions for an attack to be successful must be calculated based on the latest research trends. Research that uses the same machine learning algorithm as the target machine learning system or that handles similar data can be used as a reference. Table I- 1 shows examples of factors used to determine the capabilities of a potential attacker.

Table I- 1. Examples of factors for determining potential attacker capability

Decision Factors	Examples
Degree of intervention in training	- Amount of arbitrary data that can be mixed into the training data - Number of arbitrary models that can be mixed into the model used for training - Feasibility of replacing trained models with arbitrary models by the attacker (For example, if the potential attacker can be entrusted with model training, the model can be replaced. In this case, poisoning attacks are possible regardless of the inference conditions.)
Number of processing inference and obtaining inference result	- Availability of trained models (e.g., in-vehicle systems can freely perform inference processing) - Content of inference results that can be obtained ✓ Output label ✓ Confidence score ✓ Internal output of the model during inference ✓ Inference results guessed from system behavior (Even if the inference result cannot be obtained directly, it may possibly by inferred from the system behavior, similar to a sign recognition system)
Availability of relevant data	- Part of training data - Data similar to training data - Statistical information of training data (The presence of such information facilitates model extraction attacks, model inversion attacks, and membership inference attacks.)
Publicly available information on machine learning systems	- Types of using machine learning algorithms - Specification of input data - Preprocessing of model input - Parameters for training

In the analysis, the likelihood of an attack is calculated not only by determining whether the potential attacker's ability satisfies the conditions for a successful attack but also by considering the feasibility of the attack, such as the time required to execute an inference to make the attack possible.

I-5.2. Countermeasures by Changing System Specification

Combining the damage analysis results in Chapter I-4 with the threat analysis results in Section I-5.1, countermeasures against the attacks that shall be prevent are introduced into the system specifications. Basically, a higher priority should be given to countermeasures against attacks that are expected to cause substantial damage or are easy to execute.

Two major types of countermeasures that changes system specifications are taken: system-wide and development process mitigation measures. Both countermeasures should be implemented so that the potential attacker cannot satisfy the attack conditions analyzed in Section I-5.1. Measures to mitigate damage by changing the system usage conditions are also taken. Table I- 2 shows examples of each type of mitigation measure.

Table I- 2. Mitigation measures by changing system specifications

Classification of mitigation measures	Examples
System-wide mitigation measures	- Reduce the number of times a potential attacker can touch the system or model ✓ Limit the number of data inputs - Stop outputting information that is not requested ✓ Limit output, e.g., output only the top decision label ✓ Output only decision labels, not confidence scores - Reduce the amount of public information about the system/model - Limit the number of people who can input training data
Mitigation measures in the development process	- Use data and models from trusted providers for training
Damage mitigation	- Change the system design to allow for risk ✓ Add human judgment between the output and the next process in a system that automatically executes the next process from model output

If countermeasures cannot be taken by changing system specifications, threat analysis and countermeasures against the actual machine learning systems described in the next chapter should be performed.

I-6. Threat Analysis and Countermeasures for Actual Machine Learning Systems

This chapter describes threat analysis and countermeasures for actual machine learning systems. They are assumed to be performed in the “building AI” phase of the flow of AI utilization in the AI Utilization Guidelines of the Ministry of Internal Affairs and Communications [I-4].

The targets for implementation are attacks (Section I-2.2) for which sufficient countermeasures could not be taken in the threat analysis and countermeasures using the system specification information described in Chapter I-5. This countermeasure evaluates the feasibility of such attacks and implements countermeasures specific to machine learning elements.

I-6.1. Threat Analysis for Real Models

Threat analysis for real models is performed by analyzing whether attacks that pose threats specific to machine learning systems (Section I-2.2) are actually feasible for the target machine learning system.

First, specific attack methods that can be applied are extracted from published papers based on the machine learning algorithm of the target machine learning system and the type of handling data. Next, the extracted attack methods are implemented and the success rate of the attacks is evaluated. Several tools have been published to assist in implementing attacks, including the Adversarial Robustness Toolbox (ART) [I-12], CleverHans [I-13], and Counterfit [I-14].

I-6.2. Countermeasures Specific to Machine Learning Elements

On the basis of the results of the previous section, mitigation measures specific to machine learning elements are implemented for attacks that require countermeasures. Examples of mitigation measures for each type of attack are shown in Table I- 3. After implementing mitigation measures, the threat analysis described in Section I-6.1 is reperformed to evaluate and confirm the effectiveness of the countermeasures. The effectiveness of the countermeasures will be evaluated and verified.

Table I- 3. Examples of mitigation measures specific to machine learning elements

Attacks	Mitigation Examples
Evasion attack	- Adversarial Training Method to create models that are difficult to make adversarial examples for by training developer-created adversarial examples with correct labels in advance - Certified Robustness Techniques that determine the size of the perturbation are to be guaranteed in advance and ensure that no adversarial examples exist within that range. - Smoothing Techniques to make it harder for adversarial examples to exist by smoothing decision boundaries
Poisoning attack	- Robustness Methods to make it harder to apply poisoning attacks by devising training methods such as removing tainted datasets
Model extraction attack	- Differential Privacy Techniques that make it difficult to attack individual data by disrupting the algorithm that produces the output
Model inversion attack	- Differential Privacy Techniques that make it difficult to attack individual data by disrupting the algorithm that produces the output
Membership inference attacks	- Differential Privacy Techniques that make it difficult to attack individual data by disrupting the algorithm that produces the output

I-7. Detection and Response

This chapter describes how to detect and respond with attacks in the operation of machine learning systems. It is assumed that these technologies are drafted in “planning” and “implementation on systems” and implemented in “building AI” and “operation/utilization” in the flow of AI utilization in the Guidelines for AI Utilization [I-4] issued by the Ministry of Internal Affairs and Communications. These technologies are countermeasures against threats that could not be adequately addressed by the security measures described in Chapters I-5 and I-6.

In addition, technical papers on detection techniques for MLSAs are surveyed and categorized. These

papers are organized by “attack events to be detected” and “data to be monitored.” The results are summarized in “Overview of Detection Techniques for Machine Learning-Specific Attacks.”

I-7.1. Detection and Response in Machine Learning System Security

Detection and response to attacks in machine learning systems are performed in the same way as in general cyber-attack countermeasures [I-15]. In the “detection” phase, the sign of the precursor and the indicator of an attack are monitored. In the “response” phase, containment, eradication, and restoration are performed after an attack or damage is detected.

I-7.2. Detection

Detection includes “precursor detection” and “indicator detection.” “Precursor detection” is the detection of event signs that signal future attacks. “Indicator detection” is the detection of events that indicate attacks have already occurred or are currently occurring. In both cases, the attacks and events must be determined and the necessary logs must be recorded for detection.

MITRE ATT&CK [I-16], which analyzes and models attacker behavior, is a framework used for detection and response in general cyber-attack countermeasures. For countermeasures against attacks on machine learning systems, ATLAS [I-5], also published by MITRE, provides a list of tactics and methods for MLSAs. It summarizes the methods used in each stage of attacks, from reconnaissance and initial access to attack execution and information theft, and can be used as a reference for selecting the events to be detected.

Table I- 4 shows examples of possible attack events that could be detected on the machine learning system side among the five attacks listed in Section I-2.2.

Table I- 4. Examples of attack events

Attacks	Precursor Detection		Indicator Detection
Evasion attack	- Target system survey ✓ Number of classification labels ✓ Query limit - Creation of attack accounts	- Activities to create adversarial examples	- Input of adversarial examples
Poisoning attack		- Inclusion of contaminated data in the training data - Inclusion of tainted models	- Input containing triggers (backdoor attack)
Model extraction attack		-	- Input attack query
Model inversion attack		-	- Input attack query
Membership inference attack		-	- Input attack query

Common to all attacks, the target of precursor detection includes reconnaissance activities to investigate the target system and preparation for the attack. Investigation of the target system includes surveying the number of classification labels and query limits (size and number) to be used as a reference for attack activities. The preparation includes creating an account to conduct the attack. In terms of the precursor detection targets for each attack, evasion attacks include activities to create adversarial examples, and poisoning attacks include activities to introduce contaminated data or models into the system. Poisoning attacks are also expected to occur during the development process (building AI). Therefore, the detection of poisoning attacks requires precursor detection in the development process as well.

In indicator detection, the detection targets are activities that cause system malfunction and the theft of models or training data. Activities that can cause model or system malfunctions include entering adversarial samples for evasion attacks. Poisoning attacks include input triggers for backdoor attacks. For model and training data theft, model extraction attacks, model inversion attacks, and membership inference attacks involve the input of attack queries to thieve information.

As reference information for designing specific detection methods and logs, “Overview of Detection Techniques for Machine Learning-Specific Attacks” is summarized. In this section, the survey results of technical papers on the detection of MLSAs in terms of “attack events to be detected” and “data to

be monitored” are organized.

I-7.3. Response

This section describes the concepts in coping after an attack or damage has occurred, and examples of how to deal with the five attacks described in Section I-2.2. Similar to general cyber-attacks, it is difficult to prevent all attacks specific to machine learning systems. Therefore, functions that enable emergency countermeasures and logs that enable an investigation with reference to Section I-2.2 should be designed.

As with cyber-attack countermeasures, actions should be taken in the order of “emergency countermeasures,” “investigation,” and “permanent countermeasures” when the machine learning system is attacked or damaged. “Emergency countermeasures” are taken to mitigate or halt attacks that are occurring temporarily. The next step, “investigation,” investigates the current attack and checks for similar attacks in the past to recover the system and prevent recurring attacks. The final step takes “permanent countermeasures” to prevent recurring attacks based on the information obtained from the investigation, and the system is restored.

The process is described below.

I-7.3.1. Emergency Countermeasures

To prevent the spread of damage, emergency countermeasures including system restrictions or shutdown are taken (Table I- 5). Emergency countermeasures fall into three major categories. The first is “system restriction/shutdown,” which can respond to any type of attack. The second is “mitigation through pre-processing,” which deals with attacks whose objective is “model or system malfunction.” The third is “output spoofing” for attacks aimed at “model theft,” “training data theft,” or adversarial example generation activities in evasion attacks. However, “output spoofing” requires clarifying the relationship between system transparency and accountability.

Table I- 5. Examples of emergency countermeasures

Classifications	Examples
System restriction/shutdown	- System shutdown (common to all attacks) - Suspension of attacker accounts (common to all attacks) - Limit the number of inputs (adversarial examples generation for evasion attacks, model extraction attacks, model inversion attacks, membership inference attacks) - Rollback to past models (poisoning attacks) ✓ Rollback to past models that could correctly model (before pollution) - Restrict the output of labels that were the attack target (evasion attacks and poisoning attacks)
Mitigation through pre-processing	- Mitigation of perturbation by noise removal, etc. (evasion attacks) - Trigger removal (backdoor attack for poisoning attacks)
Output spoofing	- Spoofing of output labels (adversarial examples generation for evasion attacks, model extraction attacks, model inversion attacks, membership inference attacks) - Spoofing of confidence score (adversarial examples generation for evasion attacks, model extraction attacks, model inversion attacks, membership inference attacks)

I-7.3.2. Investigation

To recover the system and prevent recurring attacks, investigations of previous attacks should be conducted. Basically, three types of the investigations are conducted: the investigation of the purpose, methods, and damage of previous attacks; the investigation of similar previously conducted attacks; and the investigation of combined attacks.

Table I- 6. Examples of attack investigation

Attacks	Investigation of Attacks	Investigation of similar attacks	Combination survey
Evasion attack	- Identification of target labels for attacks - Identification of methods for creating adversarial examples	- Search for previous evasion attacks	- Feasibility investigation of evasion attacks through model extraction attacks - Feasibility Investigation of evasion attacks through poisoning attacks
Poisoning attack	- Identification of target labels for attacks - Trigger identification - Identification of contaminated data/contaminated model - Identification of contamination route - Identification of the method used to create contaminated data	- Search for past misclassification caused by poisoning attacks	- Feasibility investigation of model extraction attacks through poisoning attacks
Model extraction attack	- Identification of the amount of thieved information about the mode (model building from input/output data) - Identification of attack methods	- Search for past attempts of attacks	-
Model inversion attack	- Identification of leaked training data - Identification of attack methods	- Search for past attempts of attacks	- Feasibility investigation of model inversion attacks through model extraction attacks

Membership inference attack	- Identification of leaked information - Identification of attack methods	- Search for past attempts of attacks	- Feasibility investigation of membership inference attacks through model extraction attacks

I-7.3.3. Permanent Countermeasures

Permanent countermeasures involve applying defensive and mitigation measures against attacks and introducing measures to detect attacks. Information obtained from surveys will enable the effective application of defensive and mitigation measures.

- Evasion Attack

  Adversarial training with adversarial examples used in the attack
- Poisoning Attack

  Retraining after removing contaminated data and models

I-8. References

[I-1] Organization for Economic Co-operation and Development (OECD),
Principles on Artificial Intelligence.
https://legalinstruments.oecd.org/en/instruments/OECD-LEGAL-0449

[I-2] The Integrated Innovation Strategy Promotion Council, Social Principles of Human-Centric AI.
https://www.cas.go.jp/jp/seisaku/jinkouchinou/pdf/humancentricai.pdf

[I-3] Minister of Economy, Trade and Industry (METI), AI Governance in Japan Ver. 1.1.
https://www.meti.go.jp/shingikai/mono_info_service/ai_shakai_jisso/pdf/20210709_8.pdf

[I-4] Ministry of Internal Affairs and Communications (MIC), The Conference toward AI Network Society, AI Utilization Guidelines.
https://www.soumu.go.jp/main_content/000658284.pdf

[I-5] MITRE, ATLAS. https://atlas.mitre.org

[I-6] European Network and Information Security Agency (ENISA),
Artificial Intelligence Cybersecurity Challenges.
https://www.enisa.europa.eu/publications/artificial-intelligence-cybersecurity-challenges

[I-7] Microsoft Corporation, Threat Modeling AI/ML Systems and Dependencies.
https://docs.microsoft.com/en-us/security/engineering/threat-modeling-aiml

[I-8] Information Commissioner’s Office (ICO),
AI and data protection risk mitigation and management toolkit.
https://ico.org.uk/about-the-ico/ico-and-stakeholder-consultations/ai-and-data-protection-risk-mitigation-and-management-toolkit/

[I-9] National Institute of Standards and Technology (NIST), Draft NIST IR8269:
A Taxonomy and Terminology of Adversarial Machine Learning.
https://csrc.nist.gov/publications/detail/nistir/8269/draft

[I-10] National Institute of Advanced Industrial Science and Technology (AIST), Machine Learning Quality Management Guideline, 2nd English Edition.
https://www.digiarc.aist.go.jp/en/publication/aiqm/guideline-rev2.html

[I-11] Ministry of Internal Affairs and Communications (MIC) x Mitsui Bussan Secure Directions, Inc., AI Security Matrix.
https://www.mbsd.jp/aisec_portal/index.html

[I-12] International Business Machines Corporation, Adversarial Robustness Toolbox.
https://github.com/Trusted-AI/adversarial-robustness-toolbox

[I-13] CleverHans Lab, CleverHans.
https://github.com/cleverhans-lab/cleverhans

[I-14] Microsoft Corporation, Counterfit.
https://github.com/Azure/counterfit/

[I-15] National Institute of Standards and Technology (NIST), NIST SP800-61 Rev. 2: Computer Security Incident Handling Guide.
https://csrc.nist.gov/publications/detail/sp/800-61/rev-2/final

[I-16] MITRE, MITRE ATT&CK.
https://attack.mitre.org

Members of the Editing Committee of Machine Learning System Security Guidelines

<Current Members>
Yoshikazu Hanatani (Toshiba Corporation)
Masazumi Hayashi (Teikyo Heisei University)
Maki Inui (Fujitsu Limited)
Fumiyoshi Kasahara (Fujitsu Limited)
Kei Kureishi (Toshiba Corporation)
Takao Okubo (Institute of Information Security)
Kentaro Tsuji (Fujitsu Limited)
Jun Yajima (Fujitsu Limited)
Nobukazu Yoshioka (Waseda University)

<Former Members>
Daiki Ichihara (NTT DATA Corporation)
Tomoko Kaneko (National Institute of Informatics)
Ikuya Morikawa (Fujitsu Limited)
Takanori Oikawa (Fujitsu Limited)
Kenji Taguchi (National Institute of Informatics)