National Institute of Advanced Industrial Science and Technology
AIST

A Taxonomy of Security Threats, Vulnerabilities, and Controls of AI Systems

Yusuke Kawamoto

AIST (National Institute of Advanced Industrial Science and Technology), Japan

PRESTO, JST, Japan

2nd Ethical AI Workshop in Comète

About this talk

about my own opinion (not AIST's)

- Introduces a part of an activity on the *ML Quality Management Guideline* published by AIST (National Institute of Advanced Industrial Science and Technology).

Principles	Social principles of Human-centric AI (Japan)	Ethics Guidelines for Trustworthy AI (EU)	Recommendation on the Ethics of AI (UNESCO)
Laws		AI Act (EU)	
Governance	Governance Guidelines for Implementation of AI Principles (METI, Japan)		
Frameworks for quality management	**ML Quality Management Guideline (AIST, Japan)**	Guideline for Trustworthy AI (Fraunhofer Institute, Germany)	AI Risk Management Framework (NIST, USA)
Techniques for quality management			

Trying to fill the gap between abstract principles & individual ML technologies.

ISO/IEC TR 5469 (Functional Safety and AI systems) is based on this guideline.

Background: Key principles/requirements for ethical AI

Principles	Social principles of Human-centric AI (Japan)	Ethics Guidelines for Trustworthy AI (EU)

7 social principles:
(1) Human-centricity
(2) Education/literacy
(3) Privacy protection
(4) Ensuring security
(5) Fair competition
(6) Fairness, accountability, transparency
(7) Innovation

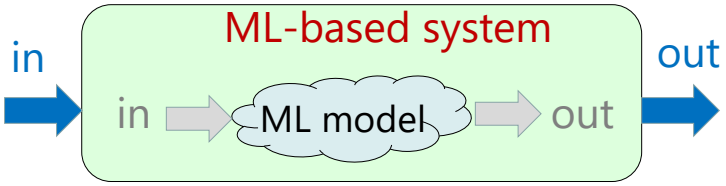
https://www.cas.go.jp/jp/seisaku/jinkouchinou/pdf/humancentricai.pdf

7 requirements:
(1) Human agency and oversight
(2) Technical robustness and safety
(3) Privacy and data gover[illegible]nce
(4) Transparency
(5) Diversity, non-di[illegible]
(6) Environmental and societal well-being
(7) Accountability

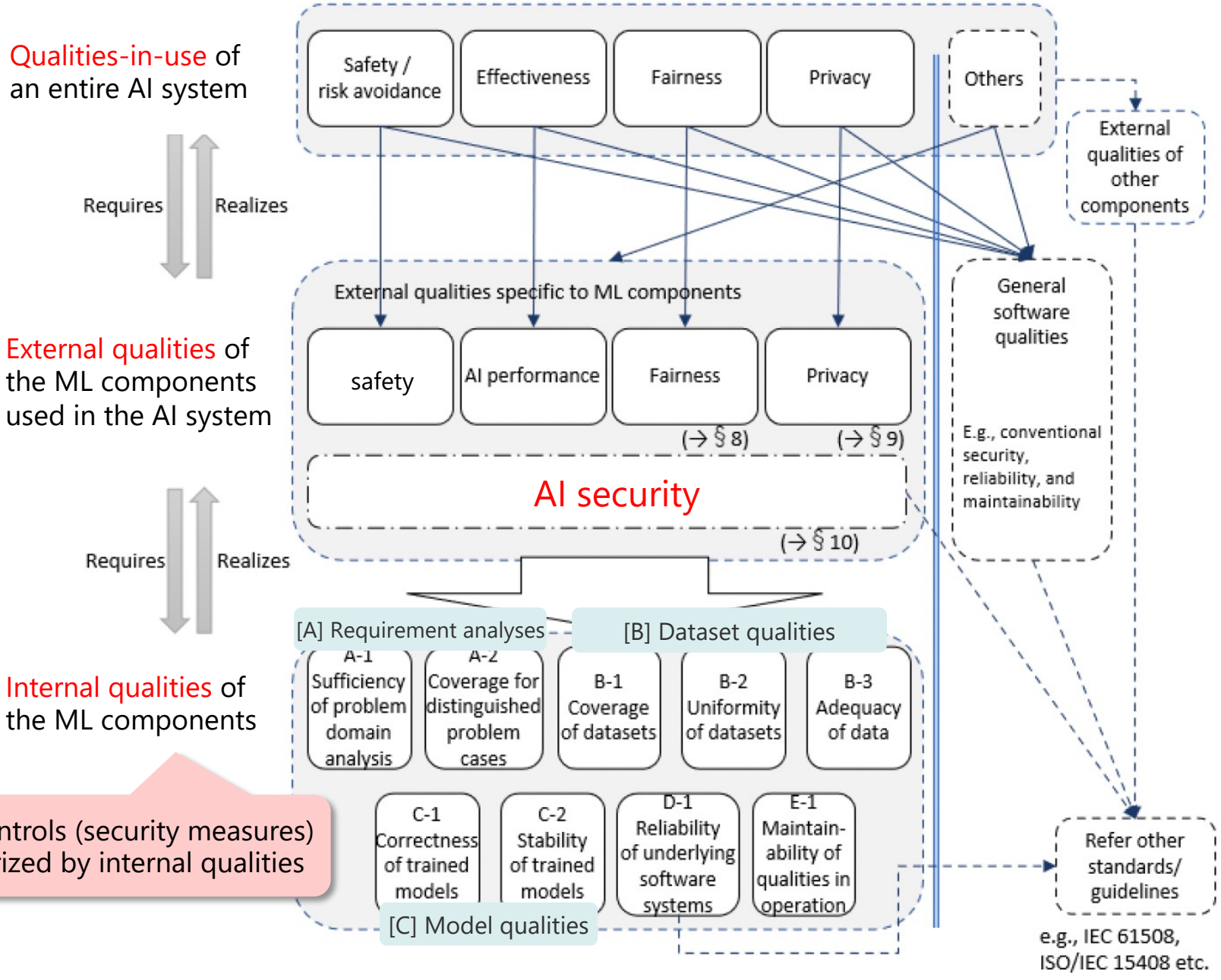
"Including ***resilience to attack and security***"

https://ec.europa.eu/digital-single-market/en/news/ethics-guidelines-trustworthy-ai

- Our guideline shows a taxonomy of the security of ML (machine learning)-based systems.



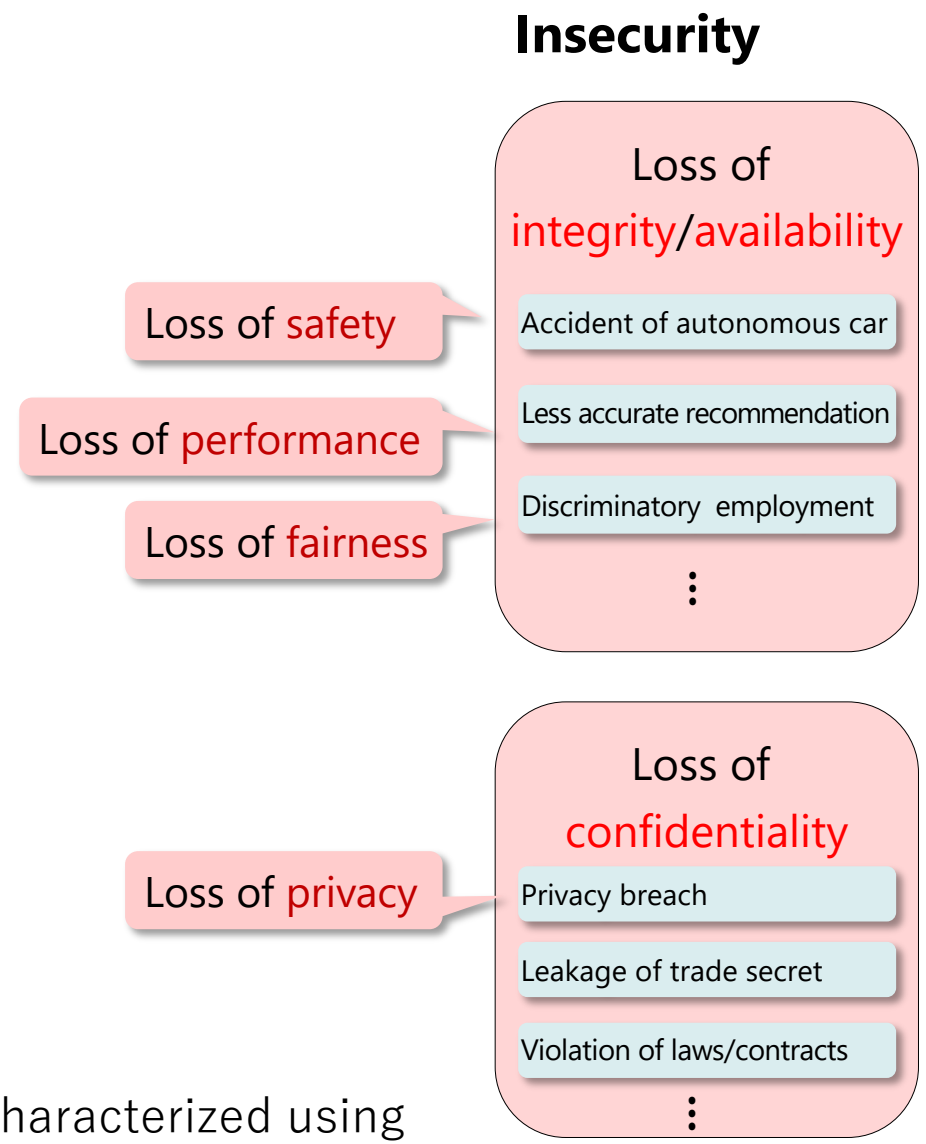
Background: Achieving the qualities in our guideline



Outline

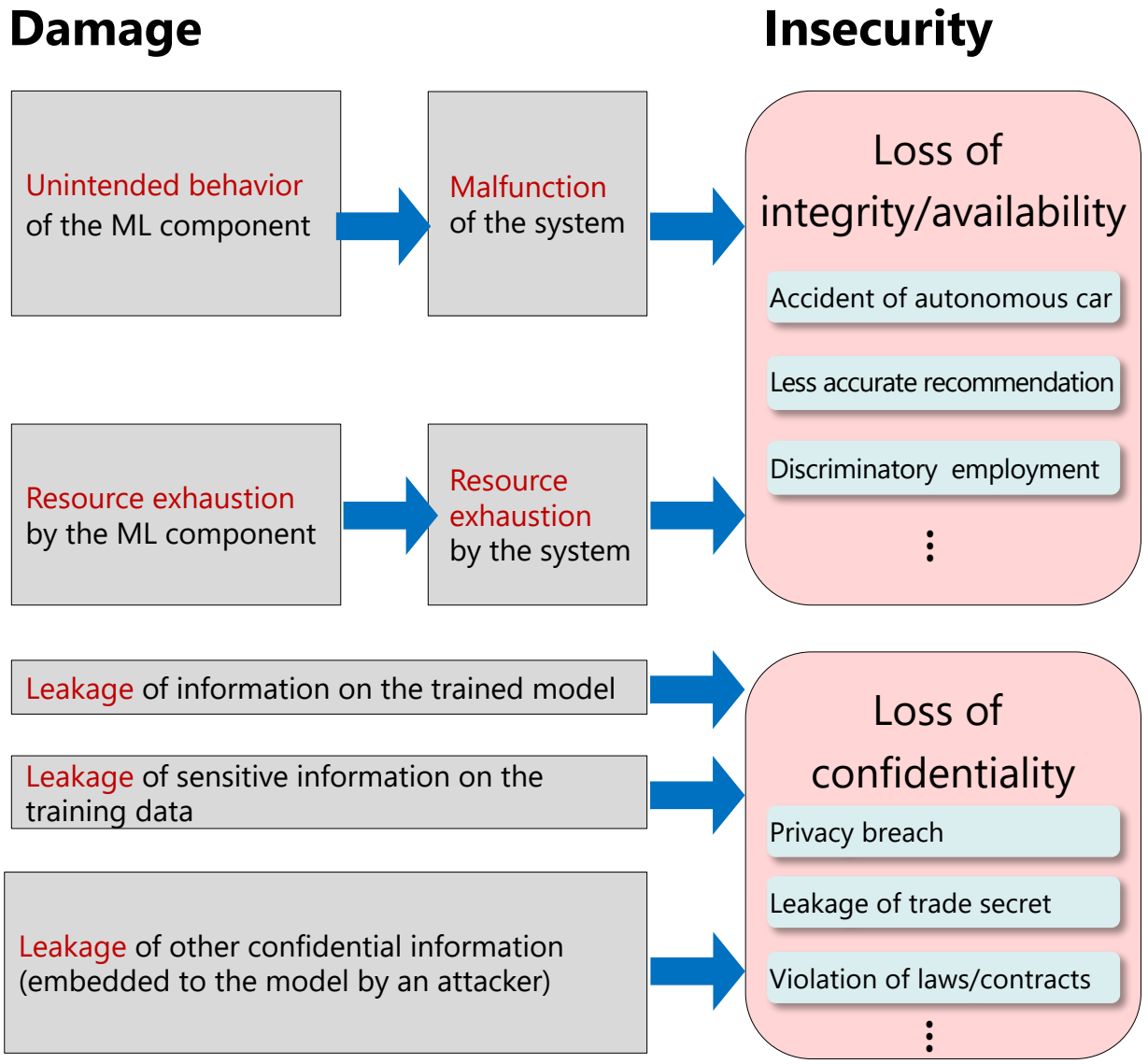
- Overview of ML Quality Management Guideline
- Damages by ML-specific threats
- Characteristics of the security of AI
- Assets & stakeholders
- ML-specific threats, Vulnerabilities & security controls

Overview: Insecurity results in the loss of external qualities



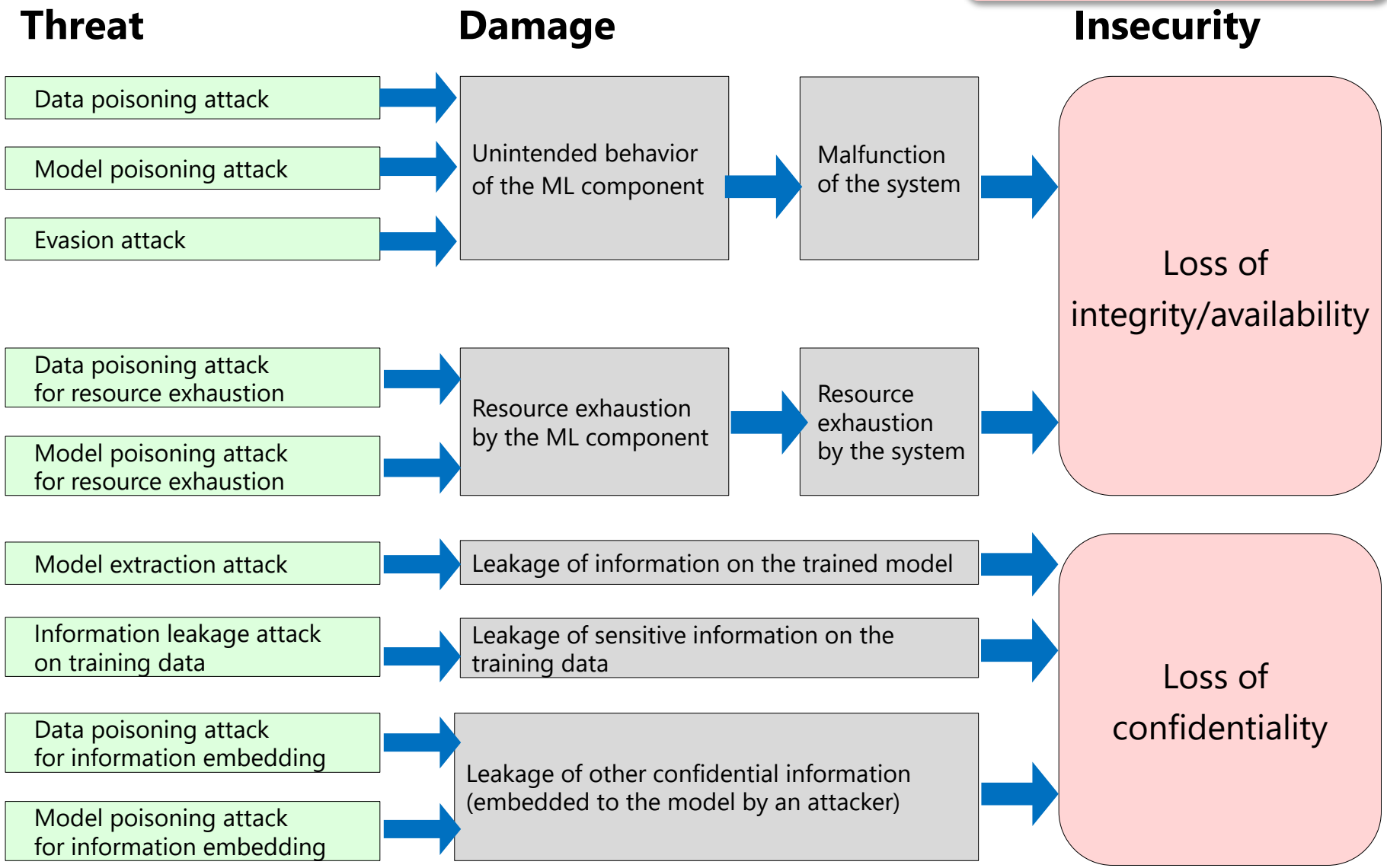
AI security is characterized using the loss of other external qualities.

Overview: Damages

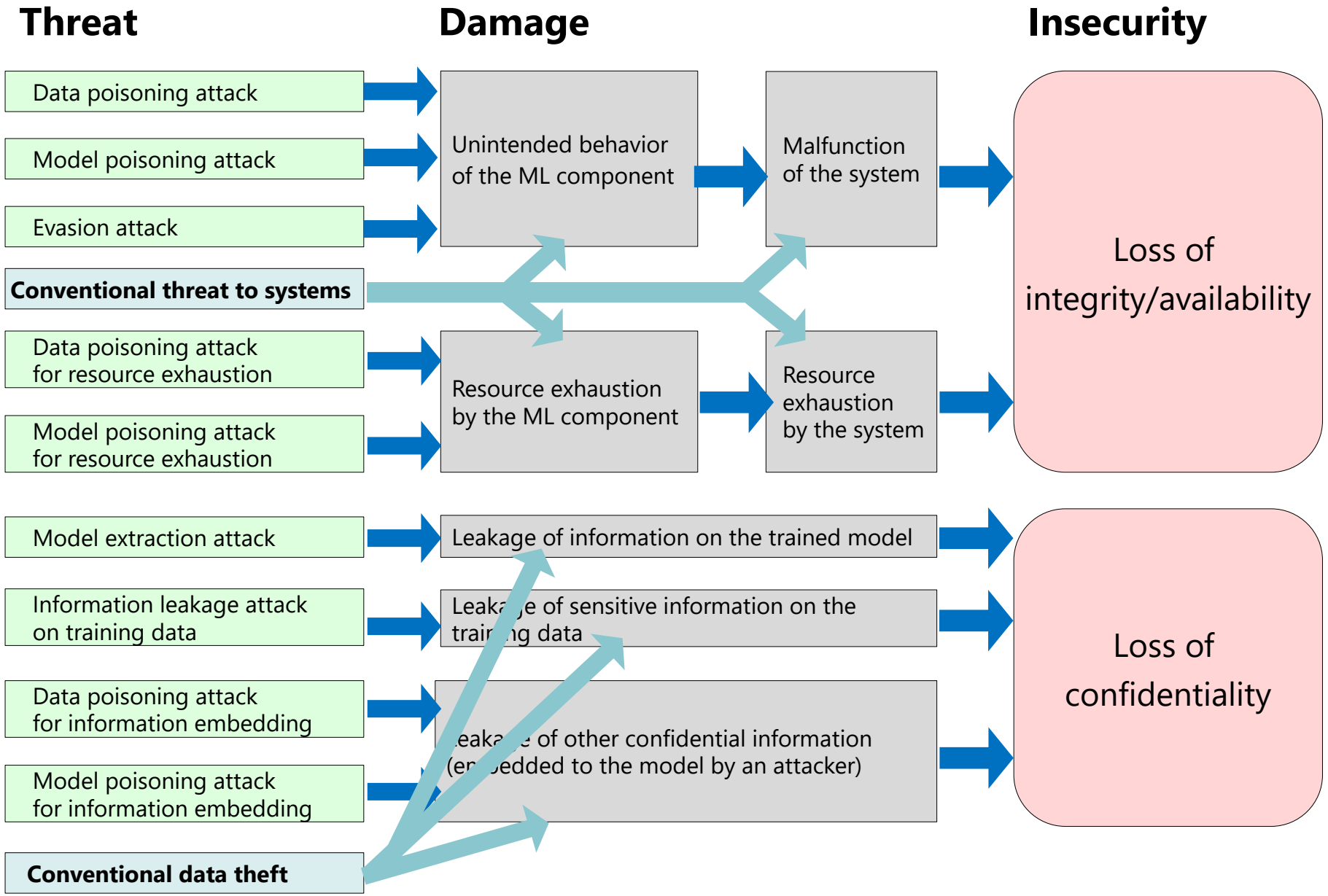


Overview: Damages by ML-specific threats

Threats that cause damages via trained models



Overview: Damages by ML-specific threats + conventional threats

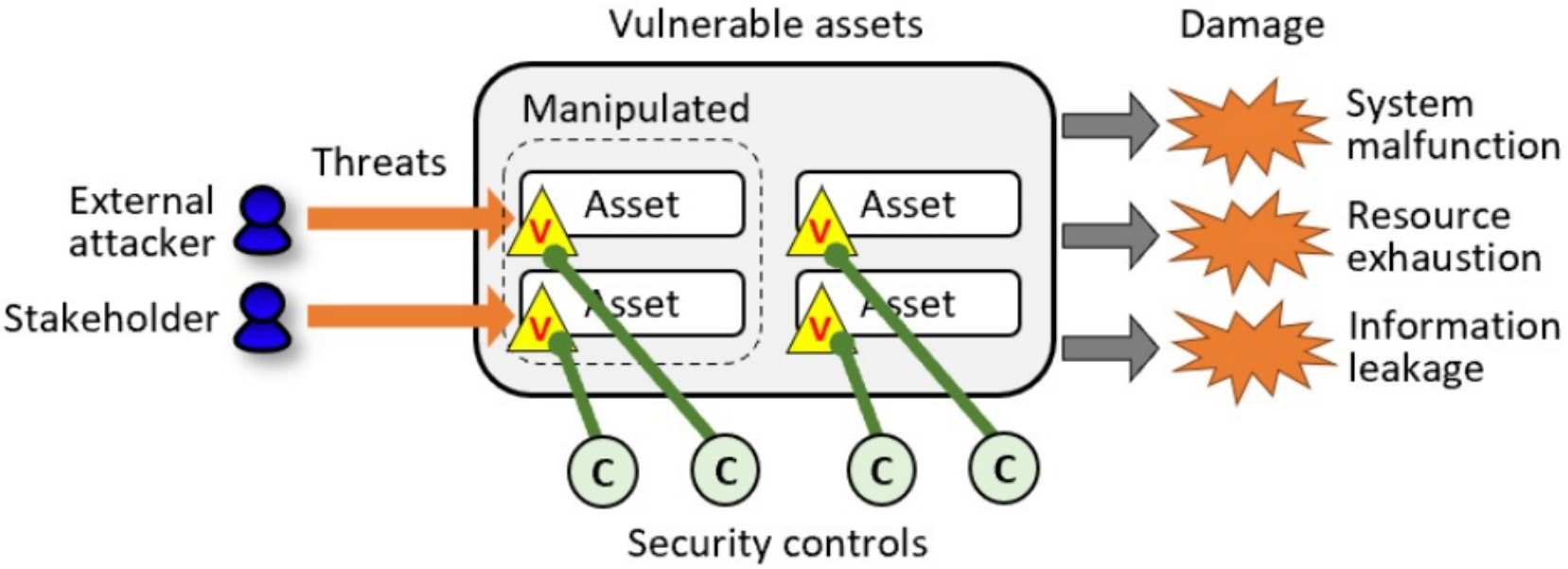


Outline

- Overview of ML Quality Management Guideline
- Damages by ML-specific threats
- Characteristics of the security of AI
- Assets & stakeholders
- ML-specific threats, Vulnerabilities & security controls

Background: Assets – stakeholders - threats – vulnerabilities – controls

Our framework is consistent with ISO 27000 series (a standard of information security).



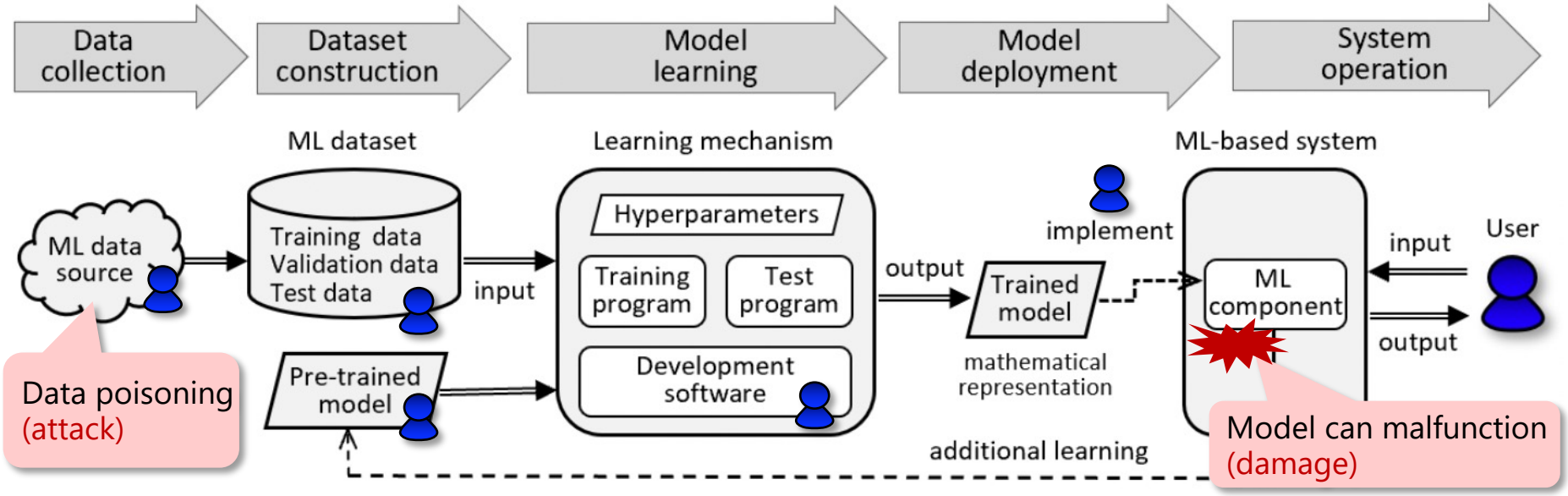
- An asset is anything valuable (e.g., data sources, datasets, trained models, and programs).
- A threat is a potential cause of the compromise of assets.
- A vulnerability is a weakness of the asset that can be exploited by threats.
- A security control is a measure against threats and vulnerabilities.

We enumerate these concepts for AI systems.

Overview: Characteristics of the security of AI/ML

1. Need to evaluate the assets in the entire lifecycle of an AI system.
   - Attacks & damages can take place in different phases.

2. Need to enumerate stakeholders and their situations to find out possible attackers.
   - Many stakeholders along the supply chain of data and pre-trained models.

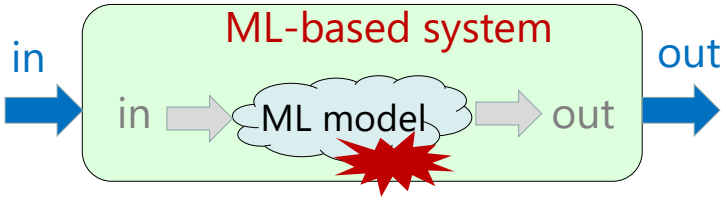


Overview: Characteristics of the security of AI/ML

1. Need to evaluate the assets in the entire lifecycle of an AI system.
   - Attacks & damages can take place in different phases.

2. Need to enumerate stakeholders and their situations to find out possible attackers.
   - Many stakeholders along the supply chain of data and pre-trained models.

3. Need to apply security controls at the system level.
   - Cannot remove a model's vulnerabilities completely.



4. Need to design multiple layers of security controls to hidden threats
   - Cannot detect threats completely

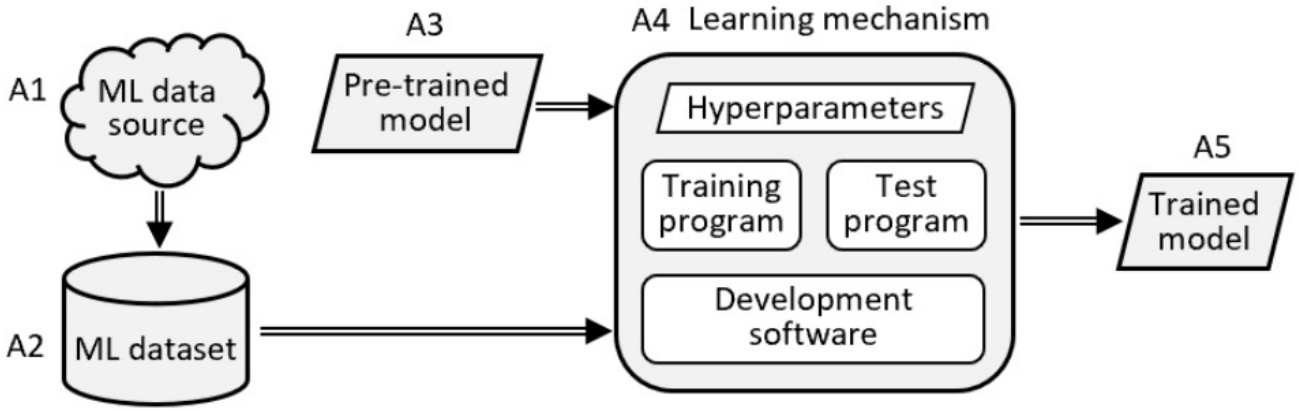
5. Need to follow the latest research about threats & vulnerabilities
   - Still many missing pieces of security controls that have not studied sufficiently

Outline

- Overview of ML Quality Management Guideline
- Damages by ML-specific threats
- Characteristics of the security of AI
- Assets & stakeholders
- ML-specific threats, Vulnerabilities & security controls

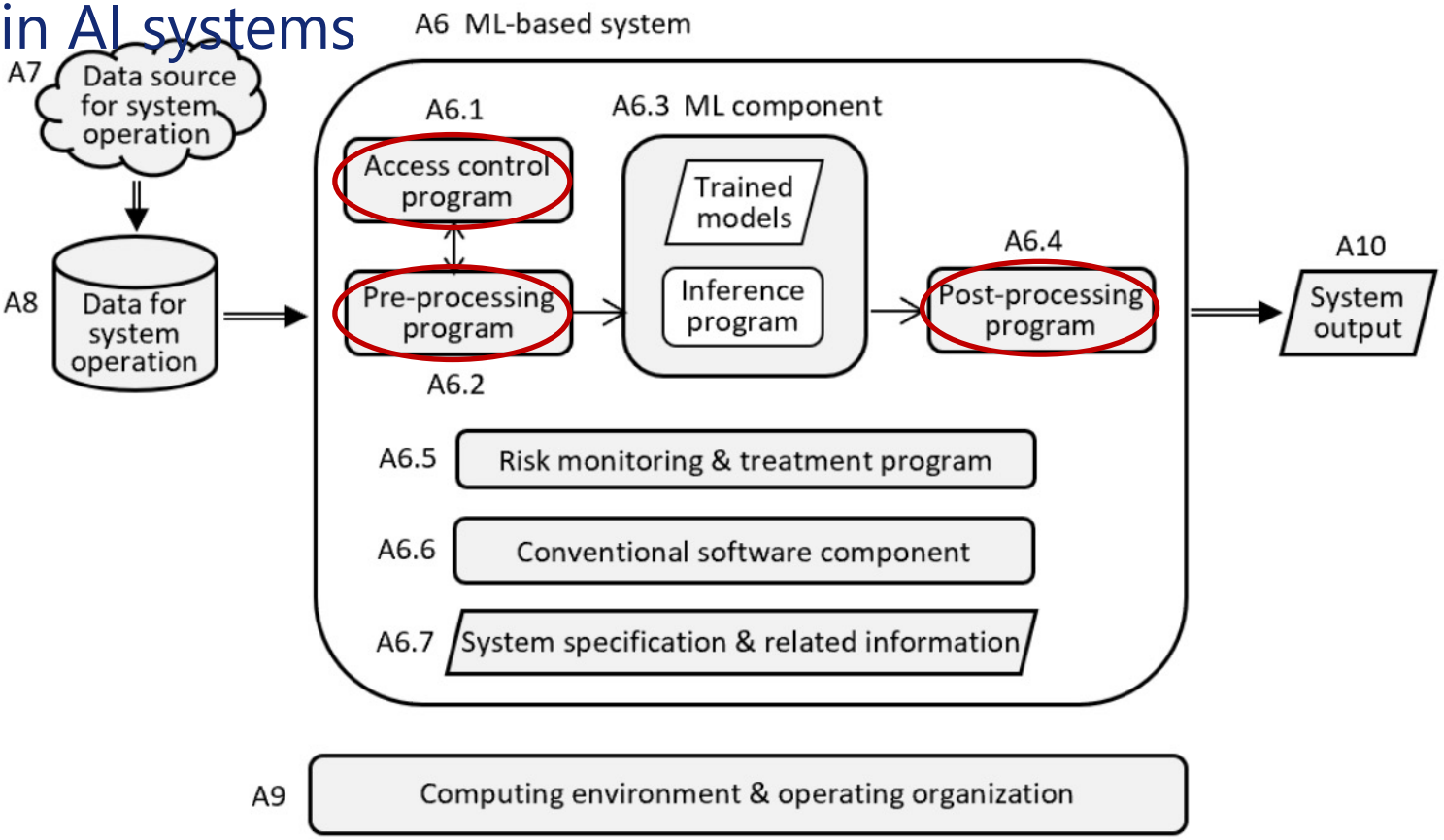
Systems with supervised learning

Assets in AI systems



Asset	Description
A1 ML data source	A population, a process, or an environment from which raw data instances are collected to construct an ML dataset.
A2 ML dataset	A collection of data instances that is obtained by pre-processing raw data and is used to train and test a model.
A3 Pre-trained model	A trained model that third parties have developed in advance and provided for other developers.
A4 Learning mechanism	A software component for developing a model from an ML dataset (and possibly a pre-trained model) by using machine learning technologies; typically consisting of hyperparameters, training programs, test programs, and development software.
A5 Trained model (or model)	A mathematical representation produced by a learning mechanism using an ML dataset.
A6 ML-based system	An information system that executes ML components and uses their output.
A11 ML data source for additional learning	A data source used to construct a dataset for additional learning after the system's operation.
A12 ML dataset for additional learning	A dataset used for additional learning after the system's operation.

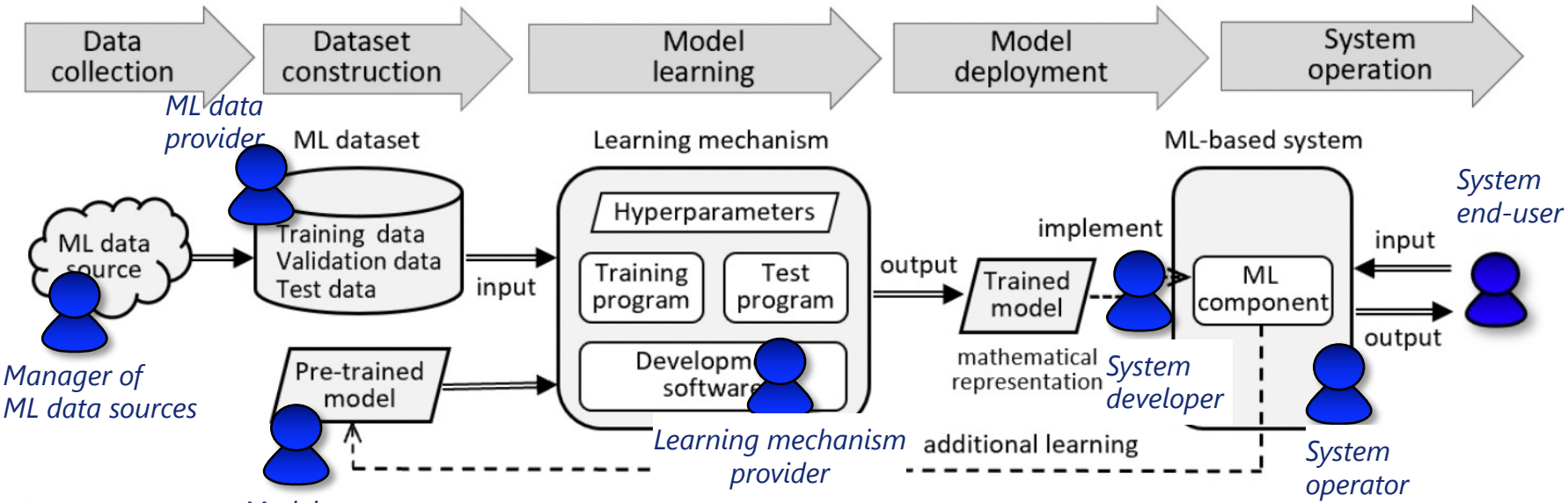
Assets in AI systems



A6.1	Access control program	A program that controls the input of data for system operation.
A6.2	Pre-processing program	A program that processes raw data to produce input to ML components. (This may access the ML components' internal information or may be combined with the ML components.)
A6.3	ML component	A software component that implements a trained model and possibly its interpretation functionality.
A6.4	Post-processing program	A program that processes the ML component's output and its interpretation.
A6.5	Monitoring/risk treatment program	A program that treats risks by monitoring the system's behavior.
A6.6	Other conventional software components	Other software components that do not consist of ML components.
A6.7	System specification & related information	Information on the ML datasets, the trained models, the other system specifications, and their related information, such as datasets or models resembling the ones used in the system development or operation.

Stakeholders in AI systems

For each asset, we may have a stakeholder that provides or manages the asset.



Stakeholder	Description
P1 Manager of ML data sources	Actor who manages ML data sources
P2 ML Data provider	Actor who provides ML datasets
P3 Model provider	Actor who provides pre-trained models
P4 Learning mechanism provider	Actor who provides learning mechanisms
P5 System developer	Actor who develops ML-based systems
P6 Manager of data sources for system operation	Actor who manages data sources for system operation
P7 Data provider for system operation	Actor who provides data for system operation
P8 System operator	Actor who operates ML-based systems to use them or to provide their services for system end-users
P9 System end-user	Actor who provides ML-based systems with input data to use their services
P10 Manager of data sources for additional learning	Actor who manages data sources for the additional learning of models used in the ML-based systems
P11 Data provider for additional learning	Actor who provides data for the additional learning of models used in the ML-based systems
P12 Model user	Actor who is provided pre-trained models and uses the models to develop new ones

Situations of stakeholders in AI systems

For each asset, we make a list of stakeholders that provide the asset.
Then we enumerate a list of possible attackers that can manipulate each asset.

Asset		Stakeholders that provide the asset			
		System development phase		System operation phase	
A1	ML data source	P1	Manager of ML data sources		
			(Nobody)		
A2	ML dataset	P2	ML data provider		
		P5	System developer		
A3	Pre-trained model	P3	Model provider		
			(Nobody)		
A4	Learning mechanism	P4	Learning mechanism provider		
		P5	System developer		
A5	Trained model	P5	System developer		
A6	System	P5	System developer	P8	System operator

Situations of stakeholders in AI systems

For each asset, we make a list of stakeholders that provide the asset.
Then we enumerate a list of possible attackers that can manipulate each asset.

A7	Data source for system operation			P6	Manager of data sources for system operation
					(Nobody)
A8	Data for system operation			P7	Data provider for system operation
				P9	System end-user
				P8	System operator
A9	Computing environment & operating organization			P8	System operator
A10	System's output data				(Nobody)
A11	ML data source for additional learning	P10	Manager of ML data sources for additional learning		
			(Nobody)		
A12	ML dataset for additional learning	P11	Data provider for additional learning		
		P5	System developer		
			(Nobody)		

Attackers & attack surface in AI systems

We made a table of possible attackers & attack surface.

The set of assets manipulated by attackers

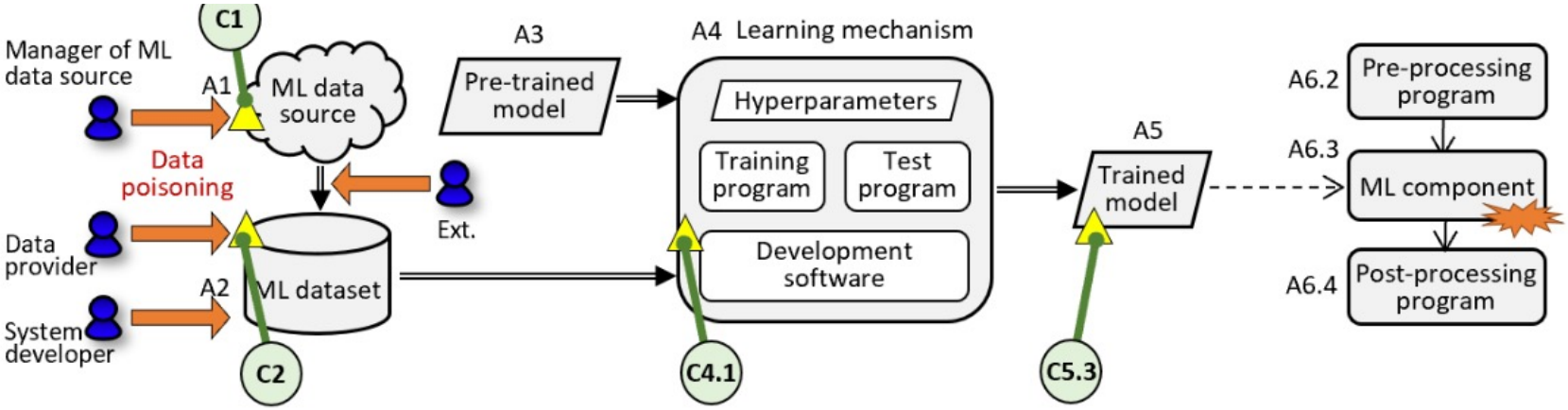
Attack situation	ML-specific Threat		Assets on the attack surface		Attackers Ext.	Dev.	Op.	Others	
Development	T1.1	Data poisoning	A1, A11	ML data source (for additional learning)	✓			P1, P10	Manager of ML data sources (for additional learning)
			A2, A12	ML dataset (for additional learning)	✓	✓		P2, P11	ML data provider (for additional learning)
	T1.2	Model poisoning	A3	Pre-trained model	✓	✓		P3	Model provider
			A4	Learning mechanism	✓	✓		P4	Learning mechanism provider
			A5	Trained model	✓	✓			
Operation	T2	Malicious input of data for system operation	A6	System	✓	✓	✓	P9	System end-user
			A7	Data source for system operation	✓			P6	Manager of data sources for system operation
			A8	Data for system operation	✓		✓	P7	Data provider for system operation
								P9	System end-user
Model provision	T1.2	Model poisoning	A3	Pre-trained model	✓	✓			
	T3	Malicious data input to ML component	A3	Pre-trained model	✓			P12	Model user

Outline

- Overview of ML Quality Management Guideline
- Damages by ML-specific threats
- Characteristics of the security of AI
- Assets & stakeholders
- ML-specific threats, vulnerabilities & security controls

[T1.1] Data poisoning attacks

An attack that manipulates a data source or a dataset to cause:
(i) the trained model's unintended behavior,
(ii) the exhaustion of resources by the trained model, or
(iii) the leakage of sensitive information from the trained model.



C1.1	Evaluate the trustworthiness of ML data sources
C1.2	Apply security controls to prevent/mitigate the poisoning of ML data sources
C1.3	Use techniques to detect the poisoning of ML data sources
C2.1	Evaluate the trustworthiness of ML datasets
C2.2	Apply security controls to prevent/mitigate the poisoning of ML datasets
C2.3	Use techniques to detect the poisoning of ML datasets
C2.4a	Use techniques to synthesize/ pre-process the ML datasets to make them resilient to data poisoning
C4.1	Use learning mechanisms being more resilient to data poisoning
C5.3	Use techniques to remove/ reduce poisoning effects from trained models

[T1.2] Model poisoning attacks

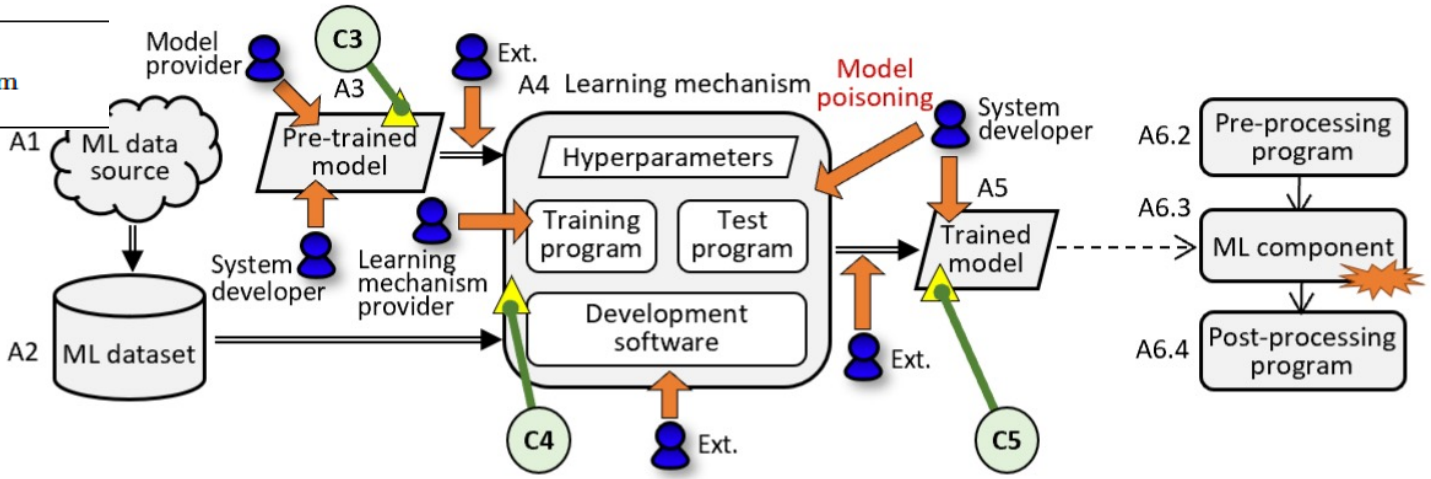
C3.1	Evaluate the trustworthiness of pre-trained models
C3.2	Apply security controls to prevent/mitigate the manipulation of pre-trained models
C3.3	Use techniques to detect poisoning effects from pre-trained models
C3.4	Use techniques to remove/ reduce poisoning effects from pre-trained models

An attack that manipulates a pre-trained model, a learning mechanism, or a trained model to cause:

(i) the trained model's unintended behavior,

(ii) the exhaustion of resources by the trained model, or

(iii) the leakage of sensitive information from the trained model.



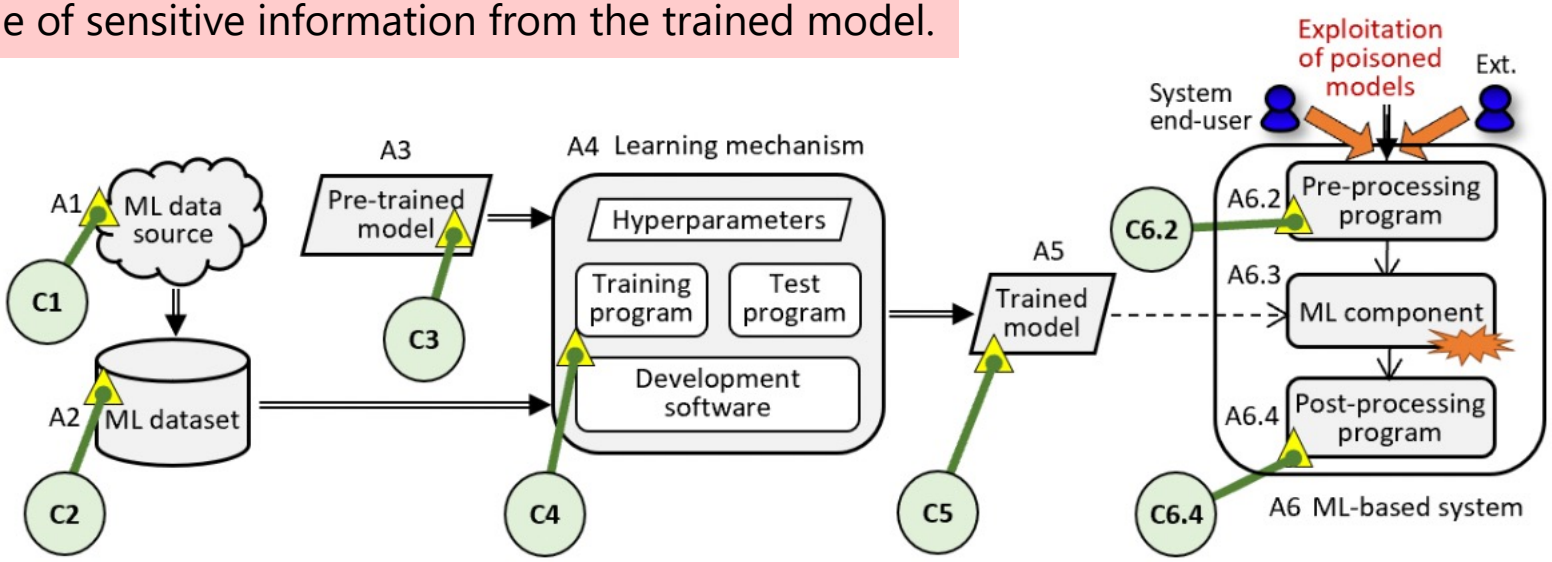
C4.2	Evaluate the trustworthiness of learning mechanisms
C4.3	Apply security controls to prevent/mitigate the manipulation of learning mechanisms
C4.4	Use learning mechanisms that can remove/reduce poisoning effects from pre-trained models

C5.1	Apply security controls to suppress/prevent the poisoning of trained models
C5.2	Use techniques to detect poisoning effects in trained models
C5.3	Use techniques to remove/ reduce poisoning effects from trained models

[T2.1] Exploitation of poisoned models

An attack that inputs malicious data during operation to exploit poisoning to cause:
(i) the trained model's unintended behavior,
(ii) the exhaustion of resources by the trained model, or
(iii) the leakage of sensitive information from the trained model.

C6.2	Use techniques to detect/ pre-process/restrict malicious input to ML components during operation



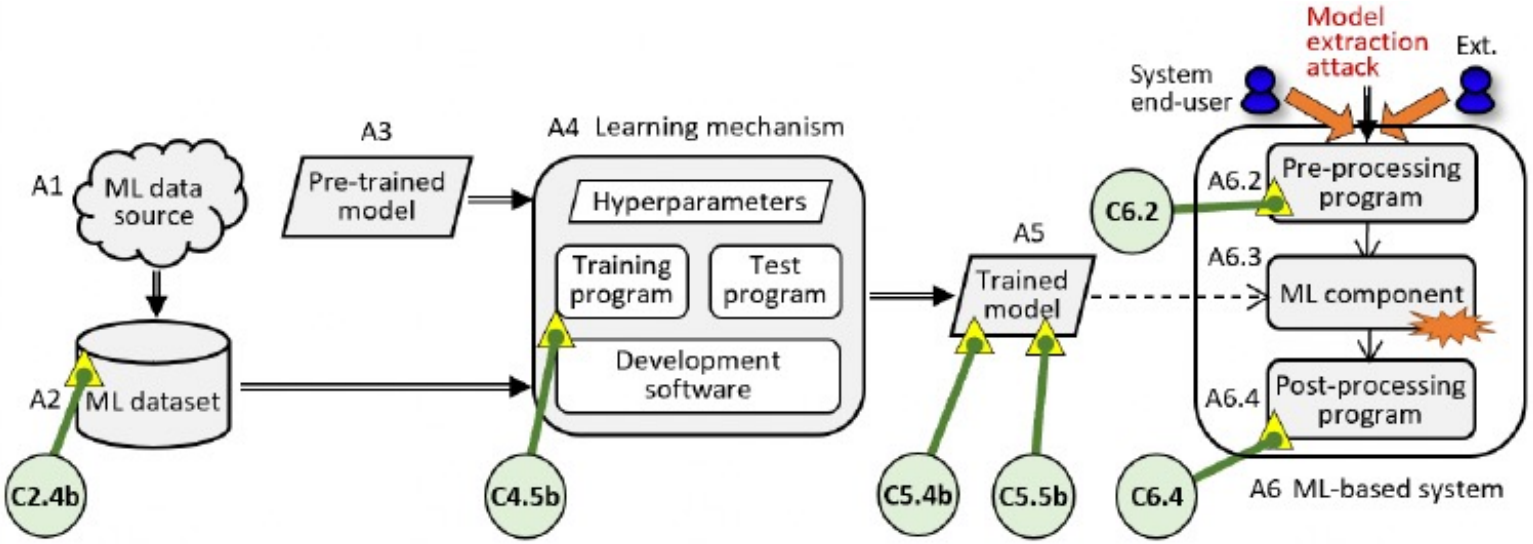
C1 to C5	Apply security controls to assets A1 to A5 against poisoning attacks (See Tables 11 and 12)

C6.4 ▲	Restrict the disclosure of the output and internal information of ML components during operation

[T2.2] Model extraction attacks

An attack that inputs malicious data to a trained model during operation to cause the leakage of information on the trained model.

C6.2	Use techniques to detect/ pre-process/restrict malicious input to ML components during operation



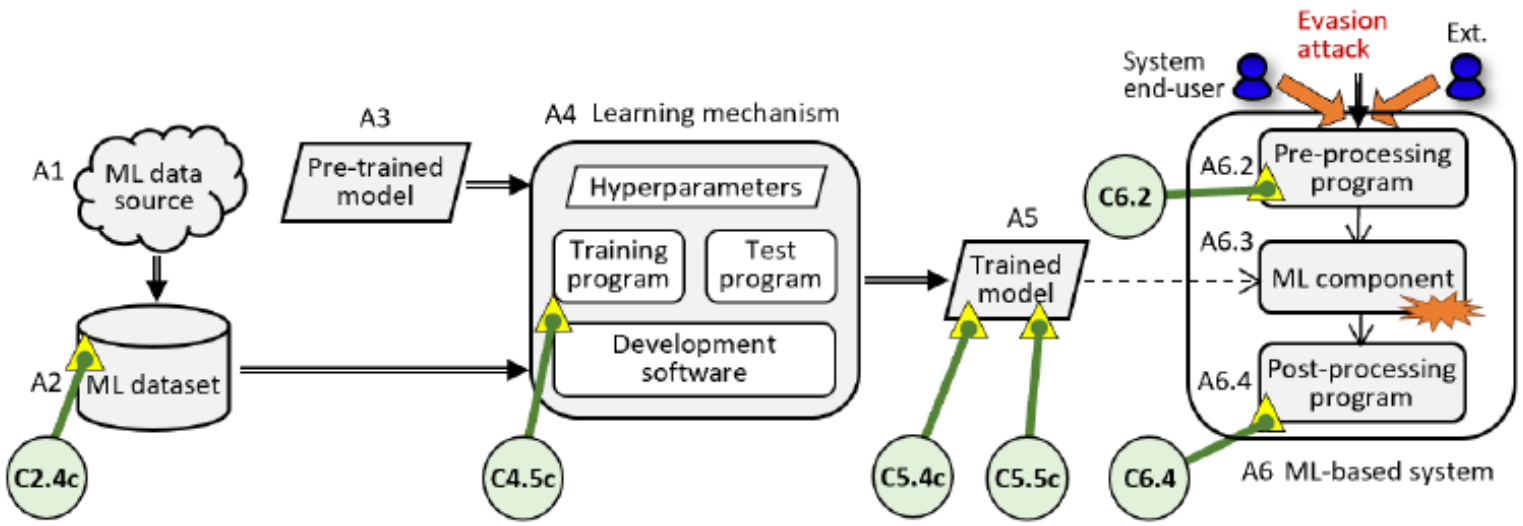
C2.4b ▲	Use techniques to synthesize/ pre-process the ML datasets to mitigate model extraction
C4.5b ▲	Use learning mechanisms that produce trained models resilient to their extraction

C5.4b	Evaluate the risk of extraction of trained models
C5.5b ▲	Use techniques to improve trained models to mitigate the leakage of information on trained models

C6.4	Restrict the disclosure of the output and internal information of ML components during operation

[T2.3] Evasion attacks

An attack that inputs adversarial examples to a trained model to cause a malfunction of the trained model.



C2.4c	Use techniques to synthesize/ pre-process the ML datasets to produce robust models against adversarial examples
C4.5c	Use learning mechanisms that produce robust models against adversarial examples

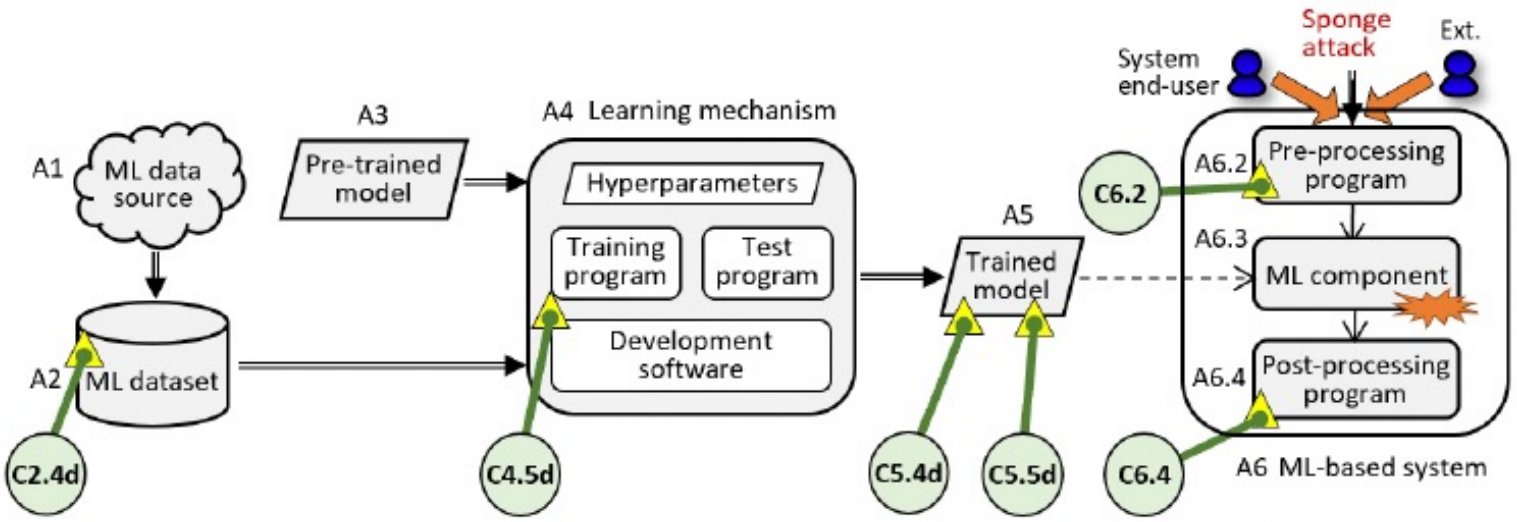
C5.4c	Evaluate the robustness of trained models against adversarial examples
C5.5c	Use techniques to improve the robustness of trained models against adversarial examples

C6.2	Use techniques to detect/ pre-process/restrict malicious input to ML components during operation
C6.4	Restrict the disclosure of the output and internal information of ML components during operation

[T2.4] Sponge attacks

An attack that inputs malicious data to a trained model
to cause the exhaustion of resources during system operation.

C6.2 ▲	Use techniques to detect/ pre-process/restrict malicious input to ML components during operation



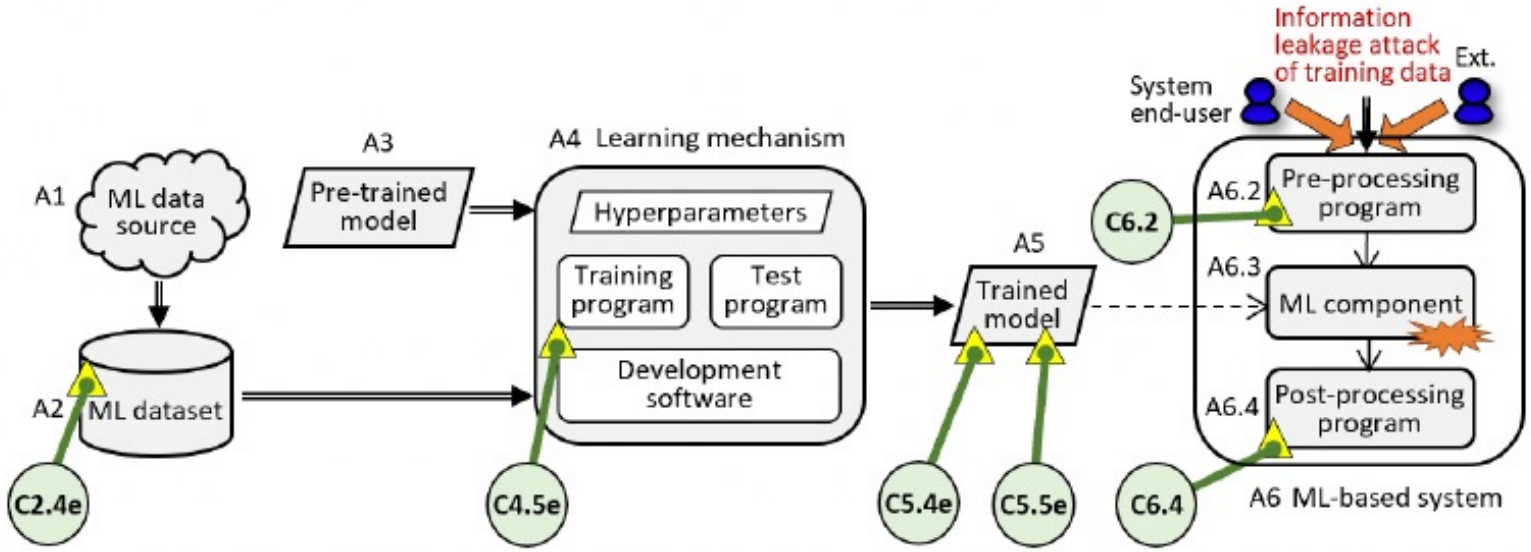
C2.4d ▲	Use techniques to synthesize/ pre-process the ML datasets to produce robust models against sponge examples
C4.5d ▲	Use learning mechanisms that produce robust models against sponge examples

C5.4d ▲	Evaluate the robustness of trained models against sponge examples
C5.5d ▲	Use techniques to improve the robustness of trained models against sponge examples

C6.4 ▲	Restrict the disclosure of the output and internal information of ML components during operation

[T2.5] Information leakage attacks of training data

An attack that inputs malicious data to a trained model to cause the leakage of sensitive information in a training dataset used to train the model.



C6.2 ▲	Use techniques to detect/ pre-process/restrict malicious input to ML components during operation

C2.4e	Use techniques to synthesize/ pre-process the ML datasets to mitigate the leakage of sensitive information in the training datasets
C4.5e	Use learning mechanisms that can prevent/mitigate the leakage of sensitive information in a training dataset

C5.4e	Evaluate the risk of information leakage from trained models
C5.5e	Use techniques to improve trained models to mitigate the leakage of information in the training dataset

C6.4	Restrict the disclosure of the output and internal information of ML components during operation

Taxonomy of threats in AI systems

ML-specific threats in the development phase

Threat	Sub-threat	Damage	Description
T1.1	Data poisoning attack		Manipulation of an ML data source or an ML dataset
	Malfunction	I, A	– to cause a malfunction of the trained model
	Targeted		* for specific inputs
	Backdoor		* for inputs that contain specific information
	Non-targeted		* for unspecified inputs
	Functionality change	I, A	– to obtain a trained model with an unintended functionality
	Resource exhaustion	A	– to cause the exhaustion of resources during system operation
	Information embedding	C	– to embed sensitive information into ML datasets to disclose it during system operation
T1.2	Model poisoning attack		Manipulation of a pre-trained model, a learning mechanism, or a trained model
	Malfunction	I, A	– to cause a malfunction of a trained model
	Targeted		* for specific inputs
	Backdoor		* for inputs that contain specific information
	Non-targeted		* for unspecified inputs
	Functionality change	I, A	– to obtain a trained model with an unintended functionality
	Resource exhaustion	A	– to cause the exhaustion of resources during system operation
	Information embedding	C	– to embed sensitive information into model parameters or hyperparameters to disclose them during system operation

Taxonomy of threats in AI systems

ML-specific threats in the operation phase

Threat	Sub-threat	Damage	Description
T2.1 Exploitation of a poisoned model			Input of malicious data to a trained model to exploit poisoning to cause
	For system malfunction	I, A	– the model's unintended behavior
	For resource exhaustion	A	– the exhaustion of resources by the model
	For information leakage	C	– the leakage of sensitive information from the model
T2.2 Model extraction attack			Input of malicious data to a trained model to cause the leakage of
	For model attributes	C	– attributes of the trained model
	For model functionalities	C	– functionalities of the trained model
T2.3 Evasion attack			Input of adversarial examples to a trained model to cause a malfunction of the trained model
	Targeted	I, A	– for specific inputs during operation
	Indiscriminate	I, A	– for unspecified inputs during operation
T2.4 Sponge attack		A	Input of sponge examples to a trained model to cause resource exhaustion during operation
T2.5 Information leakage attack of training data			Input of malicious data to a trained model to cause the leakage of sensitive information in a training dataset used to train the model
	Membership inference	C	
	Attribute inference	C	
	Data reconstruction	C	
	Property inference	C	

Security controls for each asset

Our guideline shows tables of security controls for each asset and each threat.

Assets to be controlled	Threat	Control	
A1, A11 ML data sources	Data poisoning attack	C1.1	Evaluate the trustworthiness of ML data sources
		C1.2	Apply security controls to prevent/mitigate the poisoning of ML data sources
		C1.3	Use techniques to detect the poisoning of ML data sources
A2, A12 ML datasets	Data poisoning attack	C2.1	Evaluate the trustworthiness of ML datasets
		C2.2	Apply security controls to prevent/mitigate the poisoning of ML datasets
		C2.3	Use techniques to detect the poisoning of ML datasets
		C2.4a	Use techniques to synthesize/pre-process the ML datasets to make them resilient to data poisoning
	Model extraction attack	C2.4b	Use techniques to synthesize/pre-process the ML datasets to mitigate model extraction
	Evasion attack	C2.4c	Use techniques to synthesize/pre-process the ML datasets to produce robust models against adversarial examples
	Sponge attack	C2.4d	Use techniques to synthesize/pre-process the ML datasets to produce robust models against sponge examples
	Information leakage attack of training data	C2.4e	Use techniques to synthesize/pre-process the ML datasets to mitigate the leakage of sensitive information in the training datasets

See our paper for the other assets.

Summary

- Introduced an outline of ML Quality Management Guideline (AIST).
- Analysis of assets & stakeholders is useful to list possible attackers along the supply chain of data & pre-trained models.
- Defined a taxonomy of ML-specific threats.
- Listed each asset's vulnerabilities & security controls for each threat.
- Showed the tables for system developers to check the security controls for each asset & each ML-specific threat.

For discussion (?)

- In the real development of products, we need to achieve various goals (external qualities) under many kinds of possible threats.
- Trade-offs between different quality goals may be optimized by combining different security controls over different assets?
- Our guideline says we need to investigate the trade-offs among various quality goals in the development, but can we say more about this technically?

We would appreciate it if you could give us any feedbacks about the details later.

References

[1] Yusuke Kawamoto, Kazumasa Miyake, Koichi Konishi, and Yutaka Oiwa.
Threats, Vulnerabilities, and Controls of Machine Learning Based Systems: A Survey and Taxonomy.
https://arxiv.org/pdf/2301.07474.pdf

[2] National Institute of Advanced Industrial Science and Technology (AIST).
Machine Learning Quality Management Guideline, 3rd English Edition.
https://www.digiarc.aist.go.jp/en/publication/aiqm/guideline-rev3.html

Security controls for each asset

Our guideline shows tables of security controls for each asset and each threat.

Assets to be controlled	Threat	Control	
A3 Pre-trained model	Model poisoning attack	C3.1	Evaluate the trustworthiness of pre-trained models
		C3.2	Apply security controls to prevent/mitigate the manipulation of pre-trained models
		C3.3	Use techniques to detect poisoning effects from pre-trained models
		C3.4	Use techniques to remove/reduce poisoning effects from pre-trained models
A4 Learning mechanism	Data poisoning attack	C4.1	Use learning mechanisms being more resilient to data poisoning
	Model poisoning attack	C4.2	Evaluate the trustworthiness of learning mechanisms
		C4.3	Apply security controls to prevent/mitigate the manipulation of learning mechanisms
		C4.4	Use learning mechanisms that can remove/reduce poisoning effects from pre-trained models
	Model extraction attack	C4.5b	Use learning mechanisms that produce trained models resilient to their extraction
	Evasion attack	C4.5c	Use learning mechanisms that produce robust models against adversarial examples
	Sponge attack	C4.5d	Use learning mechanisms that produce robust models against sponge examples
	Information leakage attack of training data	C4.5e	Use learning mechanisms that can prevent/mitigate the leakage of sensitive information in a training dataset
A5 Trained model	Model poisoning attack	C5.1	Apply security controls to suppress/prevent the poisoning of trained models
		C5.2	Use techniques to detect poisoning effects in trained models
	Data/model poisoning attack	C5.3	Use techniques to remove/reduce poisoning effects from trained models
	Model extraction attack	C5.4b	Evaluate the risk of extraction of trained models
	Evasion attack	C5.4c	Evaluate the robustness of trained models against adversarial examples
	Sponge attack	C5.4d	Evaluate the robustness of trained models against sponge examples
	Information leakage attack of training data	C5.4e	Evaluate the risk of information leakage from trained models

Security controls for each asset

Our guideline shows tables of security controls for each asset and each threat.

Assets to be controlled		Threat	Control	
A6.1	Access control program	Malicious input of data for system operation	C6.1	Enforce access control for ML components during operation
A6.2	Pre-processing program	Malicious input of data for system operation	C6.2	Use techniques to detect/pre-process/restrict malicious input to ML components during operation
A6.3	ML component	Exploitation of poisoned models	C1 to C5	Apply the security controls of assets A1 to A5 against poisoning attacks (See Tables 20 and 21)
		Model extraction attack	C5.5b	Use techniques to improve trained models to mitigate the leakage of information on trained models
		Evasion attack	C5.5c	Use techniques to improve the robustness of trained models against adversarial examples
		Sponge attack	C5.5d	Use techniques to improve the robustness of trained models against sponge examples
		Information leakage attack of training data	C5.5e	Use techniques to improve trained models to mitigate the leakage of information in the training dataset
A6.4	Post-processing program	Malicious input of data for system operation	C6.4	Restrict the disclosure of the output and internal information of ML components during operation
A6.5	Monitoring/risk treatment program	All types of ML-specific attacks	C6.5	Apply security controls to monitor the system's behavior and treat the risks caused by ML components
A6.1 to A6.5	Programs directly supporting the model operation	Conventional threats to systems	C6.6	Apply security controls for the vulnerability of conventional software
A6.6	Other conventional software components	Conventional threats to systems	C6.6	Apply security controls for the vulnerability of conventional software
A6.7	System specification, etc.	All types of ML-specific attacks	C6.7	Restrict the disclosure of the ML datasets, the trained models, the other system specifications, and their related information

Security controls for each asset

Our guideline shows tables of security controls for each asset and each threat.

Assets to be controlled	Threat	Control	
A7 Source of data for system operation	Malicious input of data for system operation	C7.1	Evaluate the trustworthiness of sources of data for system operation
		C7.2	Apply security controls to prevent/mitigate the manipulation of sources of data for system operation
		C7.3	Use techniques to detect the manipulation of sources of data for system operation
A8 Data for system operation	Malicious input of data for system operation	C8.1	Evaluate the trustworthiness of data for system operation
		C8.2	Apply security controls to prevent/mitigate the manipulation of data for system operation
		C8.3	Use techniques to detect the manipulation of data for system operation
A9 Computing environment & operating organization during system operation	All types of threats	C9.1	Apply security controls for the vulnerability of the computing environment and the operating organization during system operation
		C9.2	Update security controls continuously by the system operator to cope with the changes in the system and the environment
		C9.3	Enable the system operator to monitor attacks and damage manually