0. AI Security Overview

About the AI Exchange

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/about/*

Summary

Welcome to the go-to single resource for AI security & privacy - over 200 pages of practical advice and references on protecting AI, and data-centric systems from threats - where AI consists of Analytical AI, Discriminative AI, Generative AI and heuristic systems. This content serves as key bookmark for practitioners, and is contributed actively and substantially to international standards such as ISO/IEC and the AI Act through official standard partnerships. Through broad collaboration with key institutes and SDOs the *Exchange* represents the consensus on AI security and privacy.

Details

The OWASP AI Exchange has open sourced the global discussion on the security and privacy of AI and data-centric systems. It is an open collaborative OWASP project to advance the development of AI security & privacy standards, by providing a comprehensive framework of AI threats, controls, and related best practices. Through a unique official liaison partnership, this content is feeding into standards for the EU AI Act (50 pages contributed), ISO/IEC 27090 (AI security, 70 pages contributed), ISO/IEC 27091 (AI privacy), and OpenCRE - which we are currently preparing to provide the AI Exchange content through the security chatbot OpenCRE-Chat.

Data-centric systems can be divided into AI systems and 'big data' systems that don't have an AI model (e.g. data warehousing, BI, reporting, big data) to which many of the threats and controls in the AI Exchange are relevant: data poisoning, data supply chain management, data pipeline security, etc.

Security here means preventing unauthorized access, use, disclosure, disruption, modification, or destruction. Modification includes manipulating the behaviour of an AI model in unwanted ways.

Our **mission** is to be the go-to resource for security & privacy practitioners for AI and data-centric systems, to foster alignment, and drive collaboration among initiatives. By doing so, we provide a safe, open, and independent place to find and share insights for everyone. Follow AI Exchange at LinkedIn.

How it works

The AI Exchange is displayed here at owaspai.org and edited using a GitHub repository (see the links *Edit on Github*). It is is an **open-source living publication** for the worldwide exchange of AI security & privacy expertise. It is structured as one coherent resource consisting of several sections under 'content', each represented by a page on this website.

This material is evolving constantly through open source continuous delivery. The authors group consists of over 70 carefully selected experts (researchers, practitioners, vendors, data scientists, etc.) and other people in the community are welcome to provide input too. See the contribute page.

History

The AI Exchange was founded in 2022 by Rob van der Veer - bridge builder for security standards, Chief AI Officer at Software Improvement Group, with 33 years of experience in AI & security, lead author of ISO/IEC 5338 on AI lifecycle, founding father of OpenCRE, and currently working in ISO/IEC 27090, ISO/IEC 27091 and the EU AI act in CEN/CENELEC, where he was elected co-editor by the EU member states.

The project started out as the 'AI security and privacy guide' in October 22 and was rebranded a year later as 'AI Exchange' to highlight the element of global collaboration. In March 2025 the AI Exchange was awarded the status of 'OWASP Flagship project' because of its critical importance, together with the 'GenAI Security Project'.

Relevant OWASP AI initiatives

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/aiatowasp/*

In short, the two flagship OWASP AI projects:

- The **OWASP AI Exchange** is a comprehensive core framework of threats, controls and related best practices for all AI, actively aligned with international standards and feeding into them. It covers all types of AI, and next to security it discusses privacy as well.
- The **OWASP GenAI Security Project** is a growing collection of documents on the security of Generative AI, covering a wide range of topics including the LLM top 10.

if you're looking for information on AI at OWASP:

- If you want to **ensure security or privacy of your AI or data-centric system** (GenAI or not), or want to know where AI security standardisation is going, you can use the AI Exchange, and from there you will be referred to relevant further material (including GenAI security project material) where necessary.
- If you want to get a **quick overview** of key security concerns for Large Language Models, check out the LLM top 10 of the GenAI project. Please know that it is not complete, intentionally - for example it does not include the security of prompts.
- For **any specific topic** around Generative AI security, check the GenAI security project or the AI Exchange references.

Some more details on the projects:

- The OWASP AI Exchange(this work) is the go-to single resource for AI security & privacy - over 200 pages of practical advice and references on protecting AI, and data-centric systems from threats - where AI consists of Analytical AI, Discriminative AI, Generative AI

and heuristic systems. This content serves as key bookmark for practitioners, and is contributed actively and substantially to international standards such as ISO/IEC and the AI Act through official standard partnerships.

- The OWASP GenAI Security Project is an umbrella project of various initiatives that publish documents on Generative AI security, including the LLM AI Security & Governance Checklist and the LLM top 10 - featuring the most severe security risks of Large Language Models.
- OpenCRE.org has been established under the OWASP Integration standards project(from the *Project wayfinder*) and holds a catalog of common requirements across various security standards inside and outside of OWASP. OpenCRE will link AI security controls soon.

When comparing the AI Exchange with the GenAI Security Project, the Exchange:

- feeds straight into international standards
- is about all AI and data centric systems instead of just Generative AI
- is delivered as a single resource instead of a collection of documents
- is updated continuously instead of published at specific times
- is focusing on a framework of threats, controls, and related practices, so more technical-oriented, whereas the GenAI project covers a broader range of aspects
- also covers AI privacy
- is offered completely free of copyright and attribution

Summary - How to address AI Security?

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/summary/*

While AI offers tremendous opportunities, it also brings new risks including security threats. It is therefore imperative to approach AI applications with a clear understanding of potential threats and the controls against them. In a nutshell, the main steps to address AI security are:

- Implement **AI governance**.
- **Extend your security practices** with the AI security assets, threats and controls from this document.

- If you develop AI systems (even if you don't train your own models):
  - Involve your data and AI engineering into your traditional **(secure) software development practices**.
  - Apply appropriate process **controls** and technical controls through understanding of the threats as discussed in this document.
- Make sure your AI **suppliers** applied the appropriate controls.
- **Limit the impact** of AI by minimizing data and privileges, and by adding oversight, e.g. guardrails, human oversight.

Note that an AI system can for example be a Large Language Model, a linear regression function, a rule-based system,or a lookup table based on statistics. Throughout this document it is made clear when which threats and controls play a role.

How to use this Document

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/document/*

The AI Exchange is a single coherent resource on how to protect AI systems, presented on this website, divided over several pages.

Ways to start

- If you want to **protect your AI system**, start with risk analysis which will guide you through a number of questions, resulting in the attacks that apply. And when you click on those attacks you'll find the controls to select and implement.
- If you want to get an overview of the **attacks** from different angles, check the AI threat model or the AI security matrix. In case you know the attack you need to protect against, find it in the overview of your choice and click to get more information and how to protect against it.
- To understand how **controls** link to the attacks, check the controls overview or the periodic table.
- If you want to **test** the security of AI systems with tools, gogo the testing page.

- To learn about **privacy** of AI systems, check the privacy section.
- Looking for more information, or training material: see the references.

The structure

You can see the high-level structure on the main page. On larger screens you can see the structure of pages on the left sidebar and the structure within the current page on the right. On smaller screens you can view these structures through the menu.

In short the structure is:
0. AI security overview - this page, contais an overview of AI security and discussions of various topics.

1. General controls, such as AI governance
2. Threats through use, such as evasion attacks
3. Development-time threats, such as data poisoning
4. Runtime security threats, such as insecure output
5. AI security testing
6. AI privacy
7. References

This page will continue about:

- Threats high-over
- Various overviews of threats and controls: the matrix, the periodic table, and the navigator
- Risk analysis to select relevant threats and controls
- Discussion (how about ...) of various topics: heuristic systems, responsible AI, generative AI, the NCSC/CISA guidelines,and copyright

---

Threats overview

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/threatsoverview/*

Threat model

We distinguish three types of threats:

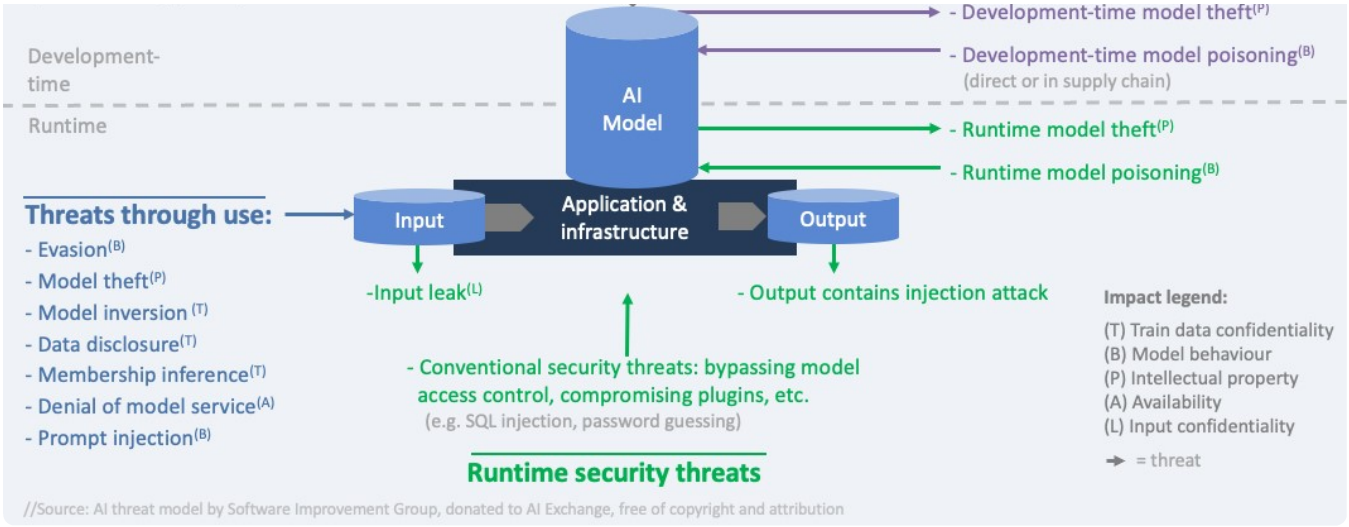
1. during development-time (when data is obtained and prepared, and the model is trained/obtained),
2. through using the model (providing input and reading the output), and
3. by attacking the system during runtime (in production).

In AI we distinguish 6 types of impacts, for three types of attacker goals (disclose, deceive and disrupt):

1. disclose: hurt confidentiality of train/test data
2. disclose: hurt confidentiality of model Intellectual property (the *model parameters* or the process and data that led to them)
3. disclose: hurt confidentiality of input data
4. deceive: hurt integrity of model behaviour (the model is manipulated to behave in an unwanted way to deceive)
5. disrupt: hurt availability of the model (the model either doesn't work or behaves in an unwanted way - not to deceive but to disrupt)
6. disrupt/disclose: confidentiality, integrity, and availability of non AI-specific assets

The threats that create these impacts use different attack surfaces. For example: the confidentiality of train data can be compromised by hacking into the database during development-time, but it can also leak by a *membership inference attack* that can find out whether a certain individual was in the train data, simply by feeding that person's data into the model and looking at the details of the model output.

The diagram shows the threats as arrows. Each threat has a specific impact, indicated by letters referring to the Impact legend. The control overview section contains this diagram with groups of controls added.



How about Agentic AI?

Think of Agentic AI as voice assistants that can control your heating, send emails, and even invite more assistants into the conversation. That's powerful—but you'd probably want it to check with you first before sending a thousand emails.

There are four key aspects to understand:

1. Action: Agents don't just chat—they invoke functions such as sending an email.
2. Autonomous: Agents can trigger each other, enabling autonomous responses (e.g. a script receives an email, triggering a GenAI follow-up).
3. Complex: Agentic behaviour is emergent.
4. Multi-system: You often work with a mix of systems and interfaces.

What does this mean for security?

- Hallucinations and prompt injections can change commands—or even escalate privileges. Don't give GenAI direct access control. Build that into your architecture.
- The attack surface is wide, and the potential impact should not be underestimated.
- Because of that, the known controls become even more important—such as traceability, protecting memory integrity, prompt injection defenses, rule-based guardrails, least model privilege, and human oversight. See the controls overview section.

For more details on the agentic AI threats, see the Agentic AI threats and mitigations, from the GenAI security project. For a more general discussion of Agentic AI, see this article from Chip Huyen.

The testing section goes into agentic AI red teaming.

AI Security Matrix

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/aisecuritymatrix/*

The AI security matrix below (click to enlarge) shows all threats and risks, ordered by type and impact.

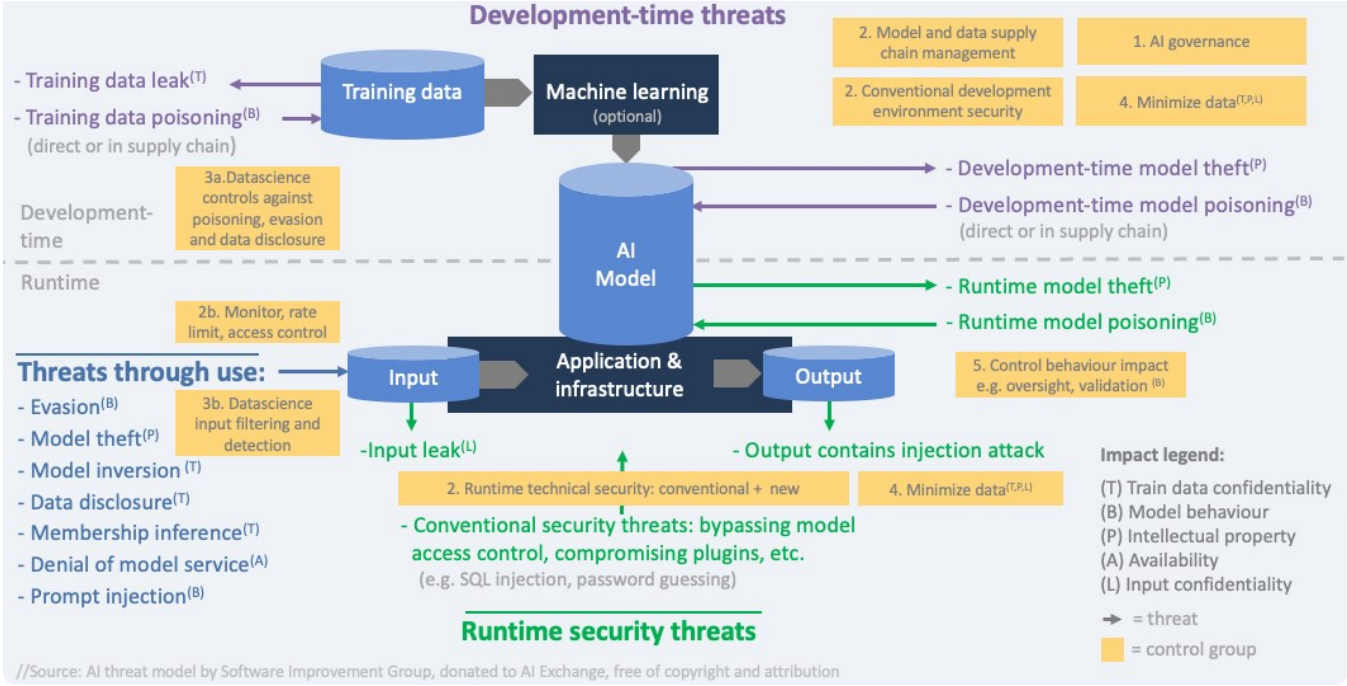
AI-specific?	Lifecycle	Attack surface	Threat/Risk category	Asset	Impacted	Unwanted result
AI	Operation	Model use (provide input/ read output)	Direct prompt injection	Model behaviour	Integrity	Manipulated unwanted model behaviour causes wrong decisions leading to business financial loss, misbehaviour going undetected, reputational damage, legal and compliance issues, operational disruption, customer dissatisfaction and churn, reduced empoloyee morale, incorrect strategic decisions, liabiity issues, personal damage and safety issues
			Indirect prompt injection			
			Evasion (e.g. adversarial examples)			
		Break into deployed model	Model poisoning in runtime (reprogramming)			
	Development	Engineering environment	Model poisoning development time			
			Data poisoning of train/finetune data			
		Supply chain	Model poisoning in supply chain (transfer learning attack)			
			Data poisoning in supply chain			
	Operation	Model use	Data disclosure in model output	Training data	Confidentiality	Leaking sensitive data can cause costs from fines and legal fees and remediation effort, loss of business through customer churn, reputation damage, loss of competitive advantage in case of trade secrets, operational disruption, impacted business relationships, and employee morale issues
			Model inversion / Membership inference			
	Development	Engineering environment	Training data leak			
	Operation	Model use	Model theft through use (input-output harvesting)	Model intellectual property	Confidentiality	If attackers can copy a model, the investment in the model is devalued caused by loss of competitive advantage, plus a copy can help craft (evasion) attacks
		Break into deployed model	Runtime model theft (not through use)			
	Development	Engineering environment	Model theft development-time			
	Operation	Model use	Denial of model service (model resource depletion)	Model behaviour	Availability	The model is not available, leading to business continuity issues, or safety problems
	Operation	All IT	Model input leak	Model input data	Confidentiality	Sensitive data in model input leaks. E.g. an LLM prompt with a sensitive question, enhanced with retrieved company secrets
	Operation	All IT	Model output contains injection attack	Any asset	C, I, A	Injection attack (from model output) causes harm
Generic	Operation	All IT	Generic runtime security attack	Any asset	C, I, A	Generic runtime security attack causes harm (includes social engineering/phishing)
	Development	All IT	Generic supply chain attack	Any asset	C, I, A	Generic supply chain security attack causes harm (e.g. vulnerability in a component)

Controls overview

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/controlsoverview/*

Threat model with controls - general

The below diagram puts the controls in the AI Exchange into groups and places these groups in the right lifecycle with the corresponding threats.



The groups of controls form a summary of how to address AI security (controls are in capitals):

1. **AI Governance**: implement governance processes for AI risk, and include AI into your processes for information security and software lifecycle:

   > *(AIPROGRAM, SECPROGRAM, DEVPROGRAM, SECDEVPROGRAM, CHECKCOMPLIANCE, SECEDUCATE)*

2. Apply conventional **technical IT security controls** risk-based, since an AI system is an IT system:
   - 2a Apply **standard** conventional IT security controls (e.g. 15408, ASVS, OpenCRE, ISO 27001 Annex A, NIST SP800-53) to the complete AI system and don't forget the new AI-specific assets :
     - Development-time: model & data storage, model & data supply chain, data science documentation:
       > *(DEVSECURITY, SEGREGATEDATA, SUPPLYCHAINMANAGE, DISCRETE)*
     - Runtime: model storage, model use, plug-ins, and model input/output:
       > *(RUNTIMEMODELINTEGRITY, RUNTIMEMODELIOINTEGRITY, RUNTIMEMODELCONFIDENTIALITY, MODELINPUTCONFIDENTIALITY, ENCODEMODELOUTPUT, LIMITRESOURCES)*
   - 2b **Adapt** conventional IT security controls to make them more suitable for AI (e.g. which usage patterns to monitor for):

> (*MONITORUSE*, *MODELACCESSCONTROL*, *RATELIMIT*)

- 2c Adopt **new** IT security controls:

> (*CONFCOMPUTE*, *MODELOBFUSCATION*, *PROMPTINPUTVALIDATION*, *INPUTSEGREGATION*)

3. Data scientists apply **data science security controls** risk-based :
   - 3a Development-time controls when developing the model:
     > (*FEDERATEDLEARNING*, *CONTINUOUSVALIDATION*, *UNWANTEDBIASTESTING*, *EVASIONROBUSTMODEL*, *POISONROBUSTMODEL*, *TRAINADVERSARIAL*, *TRAINDATADISTORTION*, *ADVERSARIALROBUSTDISTILLATION*, *MODELENSEMBLE*, *MORETRAINDATA*, *SMALLMODEL*, *DATAQUALITYCONTROL*)
   - 3b Runtime controls to filter and detect attacks:
     > (*DETECTODDINPUT*, *DETECTADVERSARIALINPUT*, *DOSINPUTVALIDATION*, *INPUTDISTORTION*, *FILTERSENSITIVEMODELOUTPUT*, *OBSCURECONFIDENCE*)
4. **Minimize data:** Limit the amount of data in rest and in transit, and the time it is stored, development-time and runtime:

   > (*DATAMINIMIZE*, *ALLOWEDDATA*, *SHORTRETAIN*, *OBFUSCATETRAININGDATA*)

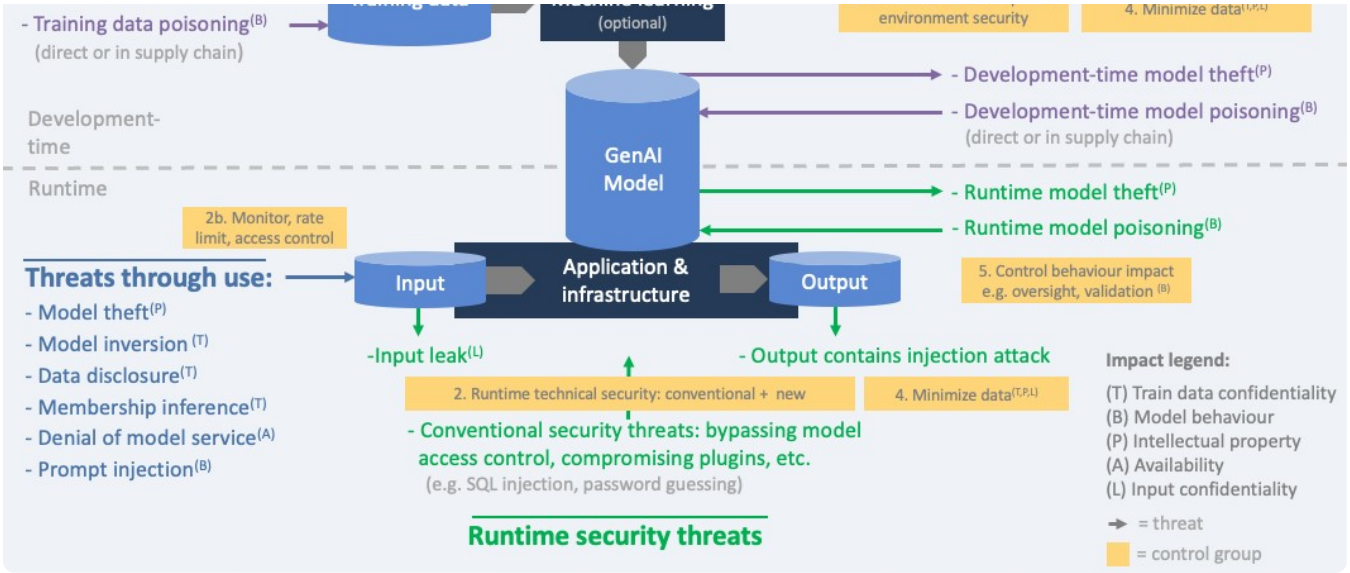
5. **Control behaviour impact** as the model can behave in unwanted ways - by mistake or by manipulation:

   > (*OVERSIGHT*, *LEASTMODELPRIVILEGE*, *AITRANSPARENCY*, *EXPLAINABILITY*, *CONTINUOUSVALIDATION*, *UNWANTEDBIASTESTING*)

All threats and controls are discussed in the further content of the AI Exchange.

Threat model with controls - GenAI trained/fine tuned

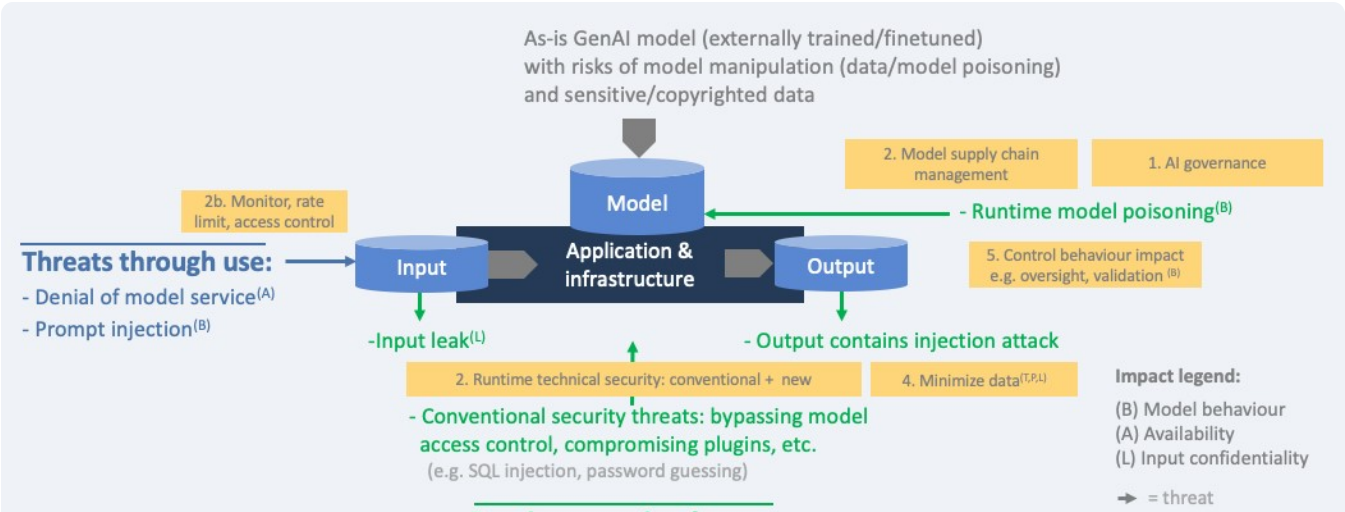
Below diagram restricts the threats and controls to Generative AI only, for situations in which **training or fine tuning** is done by the organization (note: this is not very common given the high cost and required expertise).



Threat model with controls - GenAI as-is

Below diagram restricts the threats and controls to Generative AI only where the model is used **as-is** by the organization. The provider (e.g. OpenAI) has done the training/fine tuning. Therefore, some threats are the responsibility of the model provider (sensitive/copyrighted data, manipulation at the provider). Nevertheless, the organization that uses the model should take these risks into account and gain assurance about them from the provider.

In many situation, the as-is model will be hosted externally and therefore security depends on how the supplier is handling the data, including the security configuration. How is the API protected? What is virtual private cloud? The entire external model, or just the API? Key management? Data retention? Logging? Does the model reach out to third party sources by sending out sensitive input data?



Periodic table of AI security

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/periodictable/*

The table below, created by the OWASP AI Exchange, shows the various threats to AI and the controls you can use against them – all organized by asset, impact and attack surface, with deeplinks to comprehensive coverage here at the AI Exchange website.
Note that general governance controls apply to all threats.

Asset & Impact	Attack surface with lifecycle	Threat/Risk category	Controls
Model behaviour Integrity	Runtime - Model use (provide input/ read output)	Direct prompt injection	Limit unwanted behavior, Input validation, further controls implemented in the model itself
		Indirect prompt injection	Limit unwanted behavior, Input validation, Input segregation
		Evasion (e.g. adversarial examples)	Limit unwanted behavior, Monitor, rate limit, model access control plus: Detect odd input, detect adversarial input, evasion robust model, train adversarial, input distortion, adversarial robust distillation

Asset & Impact	Attack surface with lifecycle	Threat/Risk category	Controls
	Runtime - Break into deployed model	Model poisoning runtime (reprogramming)	Limit unwanted behavior, Runtime model integrity, runtime model input/ output integrity
	Development - Engineering environment	Development-environment model poisoning	Limit unwanted behavior, Development environment security, data segregation, federated learning, supply chain management plus: model ensemble
		Data poisoning of train/finetune data	Limit unwanted behavior, Development environment security, data segregation, federated learning, supply chain management plus: model ensemble plus: More training data, data quality control, train data distortion, poison robust model, train adversarial
	Development - Supply chain	Supply-chain model poisoning	Limit unwanted behavior, Supplier: Development environment security, data segregation,

Asset & Impact	Attack surface with lifecycle	Threat/Risk category	Controls
			federated learning Producer: supply chain management plus: model ensemble
Training data Confidentiality	Runtime - Model use	Data disclosure in model output	Sensitive data limitation (data minimize, short retain, obfuscate training data) plus: Monitor, rate limit, model access control plus: Filter sensitive model output
		Model inversion / Membership inference	Sensitive data limitation (data minimize, short retain, obfuscate training data) plus: Monitor, rate limit, model access control plus: Obscure confidence, Small model
	Development - Engineering environment	Training data leaks	Sensitive data limitation (data minimize, short retain, obfuscate training data) plus: Development environment security,

Asset & Impact	Attack surface with lifecycle	Threat/Risk category	Controls
			data segregation, federated learning
Model confidentiality	Runtime - Model use	Model theft through use (input-output harvesting)	Monitor, rate limit, model access control
	Runtime - Break into deployed model	Direct model theft runtime	Runtime model confidentiality, Model obfuscation
	Development - Engineering environment	Model theft development-time	Development environment security, data segregation, federated learning
Model behaviour Availability	Model use	Denial of model service (model resource depletion)	Monitor, rate limit, model access control plus: Dos input validation, limit resources
Model input data Confidentialiy	Runtime - All IT	Model input leak	Model input confidentiality
Any asset, CIA	Runtime-All IT	Model output contains injection	Encode model output
Any asset, CIA	Runtime - All IT	Conventional runtime security attack on conventional asset	Conventional runtime security controls
Any asset, CIA	Runtime - All IT	Conventional attack on conventional supply chain	Conventional supply chain management controls

Structure of threats and controls in the deep dive section

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/navigator/*

The next big section in this document is an extensive deep dive in all the AI security threats and their controls.
The navigator diagram below shows the structure of the deep dive section, with threats, controls and how they relate, including risks and the types of controls.

Click on the image to get a PDF with clickable links.

How to select relevant threats and controls? risk analysis

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/riskanalysis/*

There are many threats and controls described in this document. Your situation and how you use AI determines which threats are relevant to you, to what extent, and what controls are who's responsibility. This selection process can be performed through risk analysis (or risk assessment) in light of the use case and architecture.

Risk management introduction
Organizations classify their risks into several key areas: Strategic, Operational, Financial, Compliance, Reputation, Technology, Environmental, Social, and Governance (ESG). A threat becomes a risk when it exploits one or more vulnerabilities. AI threats, as discussed in this resource, can have significant impact across multiple risk domains. For example, adversarial attacks on AI systems can lead to disruptions in operations, distort financial models, and result in compliance issues. See the AI security matrix for an overview of potential impact.

General risk management for AI systems is typically driven by AI governance - see AIPROGRAM and includes both risks BY relevant AI systems and risks TO those systems. Security risk assessment is typically driven by the security management system - see SECPROGRAM as this system is tasked to include AI assets, AI threats, and AI systems into consideration - provided that these have been added to the corresponding repositories.

Organizations often adopt a Risk Management framework, commonly based on ISO 31000 or

similar standards such as ISO 23894. These frameworks guide the process of managing risks through four key steps as outlined below:

1. **Identifying Risks**: Recognizing potential risks (Threats) that could impact the organization. See “Threat through use” section to identify potential risks (Threats).
2. **Evaluating Risks by Estimating Likelihood and Impact**: To determine the severity of a risk, it is necessary to assess the probability of the risk occurring and evaluating the potential consequences should the risk materialize. Combining likelihood and impact to gauge the risk’s overall severity. This is typically presented in the form of a heatmap. See below for further details.
3. **Deciding What to Do (Risk Treatment)**: Choosing an appropriate strategy to address the risk. These strategies include: Risk Mitigation, Transfer, Avoidance, or Acceptance. See below for further details.
4. **Risk Communication and Monitoring**: Regularly sharing risk information with stakeholders to ensure awareness and support for risk management activities. Ensuring effective Risk Treatments are applied. This requires a Risk Register, a comprehensive list of risks and their attributes (e.g. severity, treatment plan, ownership, status, etc). See below for further details.

Let’s go through the risk management steps one by one.

1. Identifying Risks

Selecting potential risks (Threats) that could impact the organization requires technical and business assessment of the applicable threats. A method to do this is discussed below, for every type of risk impact:

Unwanted model behaviour

Regarding model behaviour, we focus on manipulation by attackers, as the scope of this document is security. Other sources of unwanted behaviour are general inaccuracy (e.g. hallucinations) and/or unwanted bias regarding certain groups (discrimination).

This will always be an applicable threat, independent of your situation, although the risk level may sometimes be accepted - see below.

Which means that you always need to have in place:

- General governance controls (e.g. having an inventory of AI use and some control over it)
- Controls to limit effects of unwanted model behaviour (e.g. human oversight)

Is the model GenAI (e.g. a Large Language Model)?

- Prevent prompt injection (mostly done by the model supplier) in case untrusted input goes directly into the model, and there are risks that the model output creates harm, for example by offending, by providing dangerous information, or misinformation, or output that triggers harmful functions (Agentic AI). Mostly this is the case if model input is from end users and output also goes straight to end users, or can trigger functions.
- Prevent indirect prompt injection, in case untrusted data goes somehow into the prompt e.g. you retrieve somebody's resume and include it in a prompt.

Sometimes model training and running the model is deferred to a supplier. For generative AI, training is mostly performed by an external supplier given the cost of typically millions of dollars. Finetuning of generative AI is also not often performed by organizations given the cost of compute and the complexity involved. Some GenAI models can be obtained and run at your own premises. The reasons to do this can be lower cost (if is is an open source model), and the fact that sensitive input information does not have to be sent externally. A reason to use an externally hosted GenAI model can be the quality of the model.

Who trains/finetunes the model?

- The supplier: you need to prevent obtaining a poisoned model by proper supply chain management (selecting a proper supplier and making sure you use the actual model), including assuring that: the supplier prevents development-time model poisoning including data poisoning and obtaining poisoned data. If the remaining risk for data poisoning cannot be accepted, performing post-training countermeasures can be an option - see POISONROBUSTMODEL.
- You: you need to prevent development-time model poisoning which includes model poisoning, data poisoning and obtaining poisoned data or a poisoned pre-trained model in case you finetune

If you use RAG (Retrieval Augmented Generation using GenAI), then your retrieval repository

plays a role in determining the model behaviour. This means:

- You need to prevent data poisoning of your retrieval repository, which includes preventing that it contains externally obtained poisoned data.

Who runs the model?

- The supplier: make sure the supplier prevents runtime model poisoning just like any supplier who you expect to protect the running application from manipulation
- You: You need to prevent runtime model poisoning

Is the model predictive AI or Generative AI used in a judgement task (e.g. does this text look like spam)?

- Prevent an evasion attack in which a user tries to fool the model into a wrong decision using data (not instructions). Here, the level of risk is an important aspect to evaluate - see below. The risk of an evasion attack may be acceptable.

In order to assess the level of risk for unwanted model behaviour through manipulation, consider what the motivation of an attacker could be. What could an attacker gain by for example sabotaging your model? Just a claim to fame? Could it be a disgruntled employee? Maybe a competitor? What could an attacker gain by a less conspicuous model behaviour attack, like an evasion attack or data poisoning with a trigger? Is there a scenario where an attacker benefits from fooling the model? An example where evasion IS interesting and possible: adding certain words in a spam email so that it is not recognized as such. An example where evasion is not interesting is when a patient gets a skin disease diagnosis based on a picture of the skin. The patient has no interest in a wrong decision, and also the patient typically has no control - well maybe by painting the skin. There are situations in which this CAN be of interest for the patient, for example to be eligible for compensation in case the (faked) skin disease was caused by certain restaurant food. This demonstrates that it all depends on the context whether a theoretical threat is a real threat or not. Depending on the probability and impact of the threats, and on the relevant policies, some threats may be accepted as risk. When not accepted, the level of risk is input to the strength of the controls. For example: if data poisoning can lead to substantial benefit for a group of attackers, then the training data needs to be get a high level of protection.

Leaking training data

Do you train/finetune the model yourself?

- Yes: and is the training data sensitive? Then you need to prevent:
  - unwanted disclosure in model output
  - model inversion (but not for GenAI)
  - training data leaking from your engineering environment.
  - membership inference - but only if the **fact** that something or somebody was part of the training set is sensitive information. For example when the training set consists of criminals and their history to predict criminal careers: membership of that set gives away the person is a convicted or alleged criminal.

If you use RAG: apply the above to your repository data, as if it was part of the training set: as the repository data feeds into the model and can therefore be part of the output as well.

If you don't train/finetune the model, then the supplier of the model is responsible for unwanted content in the training data. This can be poisoned data (see above), data that is confidential, or data that is copyrighted. It is important to check licenses, warranties and contracts for these matters, or accept the risk based on your circumstances.

Model theft

Do you train/finetune the model yourself?

- Yes, and is the model regarded intellectual property? Then you need to prevent:
  - Model theft through use
  - Model theft development-time
  - Source code/configuration leak
  - Runtime model theft

Leaking input data

Is your input data sensitive?

- Prevent leaking input data. Especially if the model is run by a supplier, proper care needs to be taken that this data is transferred or stored in a protected way and as little as possible. Study the security level that the supplier provides and the options you have to for example disable logging or monitoring at the supplier side. Note, that if you use RAG, that the data you retrieve and insert into the prompt is also input data. This typically contains company secrets or personal data.

Misc.

Is your model a Large Language Model?

- Prevent insecure output handling, for example when you display the output of the model on a website and the output contains malicious Javascript.

Make sure to prevent model inavailability by malicious users (e.g. large inputs, many requests). If your model is run by a supplier, then certain countermeasures may already be in place.

Since AI systems are software systems, they require appropriate conventional application security and operational security, apart from the AI-specific threats and controls mentioned in this section.

2. Evaluating Risks by Estimating Likelihood and Impact

To determine the severity of a risk, it is necessary to assess the probability of the risk occurring and evaluating the potential consequences should the risk materialize.

Estimating the Likelihood:
Estimating the likelihood and impact of an AI risk requires a thorough understanding of both the technical and contextual aspects of the AI system in scope. The likelihood of a risk occurring in an AI system is influenced by several factors, including the complexity of the AI algorithms, the data quality and sources, the conventional security measures in place, and the potential for adversarial attacks. For instance, an AI system that processes public data is more susceptible to data poisoning and inference attacks, thereby increasing the likelihood of such risks. A financial institution's AI system, which assesses loan applications using public credit scores, is exposed to data poisoning attacks. These attacks could manipulate creditworthiness

assessments, leading to incorrect loan decisions.

Evaluating the Impact: Evaluating the impact of risks in AI systems involves understanding the potential consequences of threats materializing. This includes both the direct consequences, such as compromised data integrity or system downtime, and the indirect consequences, such as reputational damage or regulatory penalties. The impact is often magnified in AI systems due to their scale and the critical nature of the tasks they perform. For instance, a successful attack on an AI system used in healthcare diagnostics could lead to misdiagnosis, affecting patient health and leading to significant legal, trust, and reputational repercussions for the involved entities.

Prioritizing risks The combination of likelihood and impact assessments forms the basis for prioritizing risks and informs the development of Risk Treatment decisions. Commonly organizations use a risk heat map to visually categorize risks by impact and likelihood. This approach facilitates risk communication and decision-making. It allows the management to focus on risks with highest severity (high likelihood and high impact).

3. Risk Treatment

Risk treatment is about deciding what to do with the risks. It involves selecting and implementing measures to mitigate, transfer, avoid, or accept cybersecurity risks associated with AI systems. This process is critical due to the unique vulnerabilities and threats related to AI systems such as data poisoning, model theft, and adversarial attacks. Effective risk treatment is essential to robust, reliable, and trustworthy AI.

Risk Treatment options are:

1. **Mitigation**: Implementing controls to reduce the likelihood or impact of a risk. This is often the most common approach for managing AI cybersecurity risks. See the many controls in this resource and the 'Select controls' subsection below.
   - Example: Enhancing data validation processes to prevent data poisoning attacks, where malicious data is fed into the Model to corrupt its learning process and negatively impact its performance.
2. **Transfer**: Shifting the risk to a third party, typically through transfer learning, federated learning, insurance or outsourcing certain functions. - Example: Using third-party cloud

services with robust security measures for AI model training, hosting, and data storage, transferring the risk of data breaches and infrastructure attacks.

3. **Avoidance**: Changing plans or strategies to eliminate the risk altogether. This may involve not using AI in areas where the risk is deemed too high. - Example: Deciding against deploying an AI system for processing highly sensitive personal data where the risk of data breaches cannot be adequately mitigated.
4. **Acceptance**: Acknowledging the risk and deciding to bear the potential loss without taking specific actions to mitigate it. This option is chosen when the cost of treating the risk outweighs the potential impact. - Example: Accepting the minimal risk of model inversion attacks (where an attacker attempts to reconstruct publicly available input data from model outputs) in non-sensitive applications where the impact is considered low.

4. Risk Communication & Monitoring

Regularly sharing risk information with stakeholders to ensure awareness and support for risk management activities.

A central tool in this process is the Risk Register, which serves as a comprehensive repository of all identified risks, their attributes (such as severity, treatment plan, ownership, and status), and the controls implemented to mitigate them. Most large organizations already have such a Risk Register. It is important to align AI risks and chosen vocabularies from Enterprise Risk Management to facilitate effective communication of risks throughout the organization.

5. Arrange responsibility

For each selected threat, determine who is responsible to address it. By default, the organization that builds and deploys the AI system is responsible, but building and deploying may be done by different organizations, and some parts of the building and deployment may be deferred to other organizations, e.g. hosting the model, or providing a cloud environment for the application to run. Some aspects are shared responsibilities.

If components of your AI system are hosted, then you share responsibility regarding all controls for the relevant threats with the hosting provider. This needs to be arranged with the provider, using for example a responsibility matrix. Components can be the model, model extensions, your application, or your infrastructure. See Threat model of using a model as-is.

If an external party is not open about how certain risks are mitigated, consider requesting this information and when this remains unclear you are faced with either 1) accept the risk, 2) or provide your own mitigations, or 3)avoid the risk, by not engaging with the third party.

6. Verify external responsibilities

For the threats that are the responsibility of other organisations: attain assurance whether these organisations take care of it. This would involve the controls that are linked to these threats.

Example: Regular audits and assessments of third-party security measures.

7. Select controls

Then, for the threats that are relevant to you and for which you are responsible: consider the various controls listed with that threat (or the parent section of that threat) and the general controls (they always apply). When considering a control, look at its purpose and determine if you think it is important enough to implement it and to what extent. This depends on the cost of implementation compared to how the purpose mitigates the threat, and the level of risk of the threat. These elements also play a role of course in the order you select controls: highest risks first, then starting with the lower cost controls (low hanging fruit).

Controls typically have quality aspects to them, that need to be fine tuned to the situation and the level of risk. For example: the amount of noise to add to input data, or setting thresholds for anomaly detection. The effectiveness of controls can be tested in a simulation environment to evaluate the performance impact and security improvements to find the optimal balance. Fine tuning controls needs to continuously take place, based on feedback from testing in simulation in in production.

8. Residual risk acceptance

In the end you need to be able to accept the risks that remain regarding each threat, given the controls that you implemented.

9. Further management of the selected controls

(see SECPROGRAM), which includes continuous monitoring, documentation, reporting, and incident response.

10. Continuous risk assessment

Implement continuous monitoring to detect and respond to new threats. Update the risk management strategies based on evolving threats and feedback from incident response activities.
Example: Regularly reviewing and updating risk treatment plans to adapt to new vulnerabilities.

How about ...

How about AI outside of machine learning?

A helpful way to look at AI is to see it as consisting of machine learning (the current dominant type of AI) models and *heuristic models*. A model can be a machine learning model which has learned how to compute based on data, or it can be a heuristic model engineered based on human knowledge, e.g. a rule-based system. Heuristic models still need data for testing, and sometimes to perform analysis for further building and validating the human knowledge.
This document focuses on machine learning. Nevertheless, here is a quick summary of the machine learning threats from this document that also apply to heuristic systems:

- Model evasion is also possible for heuristic models, -trying to find a loophole in the rules
- Model theft through use - it is possible to train a machine learning model based on input/output combinations from a heuristic model
- Overreliance in use - heuristic systems can also be relied on too much. The applied knowledge can be false
- Data poisoning and model poisoning is possible by manipulating data that is used to improve knowledge and by manipulating the rules development-time or runtime
- Leaks of data used for analysis or testing can still be an issue

- Knowledge base, source code and configuration can be regarded as sensitive data when it is intellectual property, so it needs protection
- Leak sensitive input data, for example when a heuristic system needs to diagnose a patient

How about responsible or trustworthy AI?

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/responsibleai/*

There are many aspects of AI when it comes to positive outcome while mitigating risks. This is often referred to as responsible AI or trustworthy AI, where the former emphasises ethics, society, and governance, while the latter emphasises the more technical and operational aspects.

If your main responsibility is security, then the best strategy is to first focus on AI security and after that learn more about the other AI aspects - if only to help your colleagues with the corresponding responsibility to stay alert. After all, security professionals are typically good at identifying things that can go wrong. Furthermore, some aspects can be a consequence of compromised AI and are therefore helpful to understand, such as *safety*.

Let's clarify the aspects of AI and see how they relate to security:

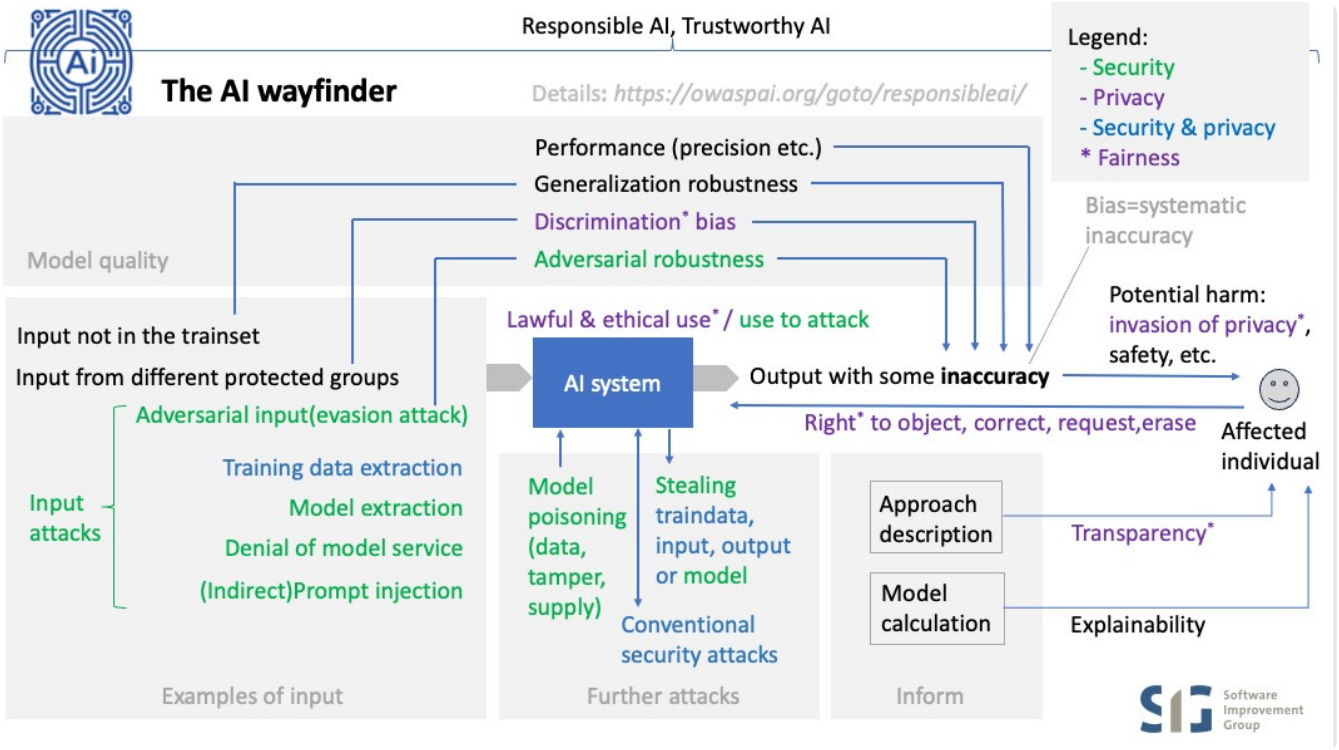
- **Accuracy** is about the AI model being sufficiently correct to perform its 'business function'. Being incorrect can lead to harm, including (physical) safety problems (e.g. car trunk opens during driving) or other wrong decisions that are harmful (e.g. wrongfully declined loan). The link with security is that some attacks cause unwanted model behaviour which is by definition an accuracy problem. Nevertheless, the security scope is restricted to mitigating the risks of those attacks - NOT solve the entire problem of creating an accurate model (selecting representative data for the trainset etc.).
- **Safety** refers to the condition of being protected from / unlikely to cause harm. Therefore safety of an AI system is about the level of accuracy when there is a risk of harm (typically implying physical harm but not restricted to that) , plus the things that are in place to mitigate those risks (apart from accuracy), which includes security to safeguard accuracy, plus a number of safety measures that are important for the business function of the model. These need to be taken care of and not just for security reasons because the model can make unsafe decisions for other reasons (e.g. bad training data), so they are a shared

concern between safety and security:

- oversight to restrict unsafe behaviour, and connected to that: assigning least privileges to the model,
- continuous validation to safeguard accuracy,
- transparency: see below,
- explainability: see below.

- **Transparency**: sharing information about the approach, to warn users and depending systems of accuracy risks, plus in many cases users have the right to know details about a model being used and how it has been created. Therefore it is a shared concern between security, privacy and safety.
- **Explainability**: sharing information to help users validate accuracy by explaining in more detail how a specific result came to be. Apart from validating accuracy this can also support users to get transparency and to understand what needs to change to get a different outcome. Therefore it is a shared concern between security, privacy, safety and business function. A special case is when explainability is required by law separate from privacy, which adds 'compliance' to the list of aspects that share this concern.
- **Robustness** is about the ability of maintaining accuracy under expected or unexpected variations in input. The security scope is about when those variations are malicious (*adversarial robustness*) which often requires different countermeasures than those required against normal variations (_generalization robustness). Just like with accuracy, security is not involved per se in creating a robust model for normal variations. The exception to this is when generalization robustness adversarial malicious robustness , in which case this is a shared concern between safety and security. This depends on a case by case basis.
- **Free of discrimination**: without unwanted bias of protected attributes, meaning: no systematic inaccuracy where the model 'mistreats' certain groups (e.g. gender, ethnicity). Discrimination is undesired for legal and ethical reasons. The relation with security is that having detection of unwanted bias can help to identify unwanted model behaviour caused by an attack. For example, a data poisoning attack has inserted malicious data samples in the training set, which at first goes unnoticed, but then is discovered by an unexplained detection of bias in the model. Sometimes the term 'fairness' is used to refer to discrimination issues, but mostly fairness in privacy is a broader term referring to fair treatment of individuals, including transparency, ethical use, and privacy rights.
- **Empathy**. The relation of that with security is that the feasible level of security should always be taken into account when validating a certain application of AI. If a sufficient level of security cannot be provided to individuals or organizations, then empathy means invalidating the idea, or takin other precautions.

- **Accountability**. The relation of accountability with security is that security measures should be demonstrable, including the process that have led to those measures. In addition, traceability as a security property is important, just like in any IT system, in order to detect, reconstruct and respond to security incidents and provide accountability.
- **AI security**. The security aspect of AI is the central topic of the AI Exchange. In short, it can be broken down into:
  - Input attacks, that are performed by providing input to the model
  - Model poisoning, aimed to alter the model's behavior
  - Stealing AI assets, such as train data, model input, output, or the model itself, either development time or runtime (see below)
  - Further runtime conventional security attacks



How about Generative AI (e.g. LLM)?

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/genai/*

Yes, GenAI is leading the current AI revolution and it's the fastest moving subfield of AI security. Nevertheless it is important to realize that other types of algorithms (let's call it

predictive AI) will remain to be applied to many important use cases such as credit scoring, fraud detection, medical diagnosis, product recommendation, image recognition, predictive maintenance, process control, etc. Relevant content has been marked with 'GenAI' in this document.

Important note: from a security threat perspective, GenAI is not that different from other forms of AI (*predictive AI*). GenAI threats and controls largely overlap and are very similar to AI in general. Nevertheless, some risks are (much) higher. Some are lower. Only a few risks are GenAI-specific. Some of the control categories differ substantially between GenAI and predictive AI - mostly the data science controls (e.g. adding noise to the training set). In many cases, GenAI solutions will use a model as-is and not involve any training by the organization whatsoever, shifting some of the security responsibilities from the organization to the supplier. Nevertheless, if you use a ready-made model, you need still to be aware of those threats.

What is mainly new to the threat landscape because of LLMs?

- First of all, LLMs pose new threats to security because they may be used to create code with vulnerabilities, or they may be used by attackers to create malware, or they may cause harm otherwiser through hallucinations, but these are out of scope of the AI Exchange, as it focuses on security threats TO AI systems.
- Regarding input:
  - Prompt injection is a completely new threat: attackers manipulating the behaviour of the model with crafted and sometimes hidden instructions.
  - Also new is organizations sending huge amounts of data in prompts, with company secrets and personal data.
- Regarding output: New is the fact that output can contain injection attacks, or can contain sensitive or copyrighted data (see Copyright).
- Overreliance is an issue. We let LLMs control and create things and may have too much trust in how correct they are, and also underestimate the risk of them being manipulated. The result is that attacks can have much impact.
- Regarding training: Since the training sets are so large and based on public data, it is easier to perform data poisoning. Poisoned foundation models are also a big supply chain issues.

GenAI security particularities are:

Nr.	GenAI security particularities	OWASP for LLM TOP 10
1	GenAI models are controlled by natural language in prompts, creating the risk of Prompt injection. Direct prompt injection is where the user tries to fool the model to behave in unwanted ways (e.g. offensive language), whereas with indirect prompt injection it is a third party that injects content into the prompt for this purpose (e.g. manipulating a decision).	(OWASP for LLM 01:Prompt injection)
2	GenAI models have typically been trained on very large datasets, which makes it more likely to output sensitive data or licensed data, for which there is no control of access privileges built into the model. All data will be accessible to the model users. Some mechanisms may be in place in terms of system prompts or output filtering, but those are typically not watertight.	(OWASP for LLM 02: Sensitive Information Disclosure)
3	Data and model poisoning is an AI-broad problem, and with GenAI the risk is generally higher since training data can be supplied from different sources that may be challenging to control, such as the internet. Attackers could for example hijack domains and place manipulated information.	(OWASP for LLM 04: Data and Model Poisoning)
4	GenAI models can be inaccurate and hallucinate. This is an AI-broad risk factor, and Large Language Models (GenAI) can make matters worse by coming across very confident and knowledgeable. In essence this is about the risk of underestimating the probability that the model is wrong or the model has been manipulated. This means that it is connected to each and every security control. The strongest link is with controls that limit the impact of unwanted model behavior, in particular Least model privilege.	(OWASP for LLM 06: Excessive agency) and (OWASP for LLM 09: Misinformation)
5	Leaking input data: GenAI models mostly live in the cloud - often managed by an external party, which may increase the risk of leaking training data and leaking prompts. This issue is not limited to GenAI, but GenAI has 2 particular	Not covered in LLM top 10

Nr.	GenAI security particularities	OWASP for LLM TOP 10
	risks here: 1) model use involves user interaction through prompts, adding user data and corresponding privacy/ sensitivity issues, and 2) GenAI model input (prompts) can contain rich context information with sensitive data (e.g. company secrets). The latter issue occurs with *in context learning* or *Retrieval Augmented Generation(RAG)* (adding background information to a prompt): for example data from all reports ever written at a consultancy firm. First of all, this information will travel with the prompt to the cloud, and second: the system will likely not respect the original access rights to the information.	
6	Pre-trained models may have been manipulated. The concept of pretraining is not limited to GenAI, but the approach is quite common in GenAI, which increases the risk of supply-chain model poisoning.	(OWASP for LLM 03 - Supply chain vulnerabilities)
7	Model inversion and membership inference are typically low to zero risks for GenAI	Not covered in LLM top 10, apart from LLM06 which uses a different approach - see above
8	GenAI output may contain elements that perform an injection attack such as cross-site-scripting.	(OWASP for LLM 05: Improper Output Handling)
9	Denial of service can be an issue for any AI model, but GenAI models typically cost more to run, so overloading them can create unwanted cost.	(OWASP for LLM 10: Unbounded consumption)

GenAI References:

- OWASP LLM top 10
- Demystifying the LLM top 10
- Impacts and risks of GenAI

- LLMsecurity.net

How about the NCSC/CISA guidelines?

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/jointguidelines/*

Mapping of the UK NCSC /CISA Joint Guidelines for secure AI system development to the controls here at the AI Exchange.
To see those controls linked to threats, refer to the Periodic table of AI security.

Note that the UK Government drove an initiative through their DSIT repartment to build on these joint guidelines and produce the DSIT Code of Practice for the Cyber Secyrity of AI, which reorganizes things according to 13 principles, does a few tweaks, and adds a bit more governance. The principle mapping is added below, and adds mostly post-market aspects:

- Principle 10: Communication and processes assoiated with end-users and affected entities
- Principle 13: Ensure proper data and model disposal

1. Secure design

- Raise staff awareness of threats and risks (DSIT principle 1):
  #SECURITY EDUCATE
- Model the threats to your system (DSIT principle 3):
  See Risk analysis under #SECURITY PROGRAM
- Design your system for security as well as functionality and performance (DSIT principle 2):
  #AI PROGRAM, #SECURITY PROGRAM, #DEVELOPMENT PROGRAM, #SECURE DEVELOPMENT PROGRAM, #CHECK COMPLIANCE, #LEAST MODEL PRIVILEGE, #DISCRETE, #OBSCURE CONFIDENCE, #OVERSIGHT, #RATE LIMIT, #DOS INPUT VALIDATION, #LIMIT RESOURCES, #MODEL ACCESS CONTROL, #AI TRANSPARENCY
- Consider security benefits and trade-offs when selecting your AI model
  All development-time data science controls (currently 13), #EXPLAINABILITY

2. Secure Development

- Secure your supply chain (DSIT principle 7):

#SUPPLY CHAIN MANAGE

- Identify, track and protect your assets (DSIT principle 5):
  #DEVELOPMENT SECURITY, #SEGREGATE DATA, #CONFIDENTIAL COMPUTE, #MODEL INPUT CONFIDENTIALITY, #RUNTIME MODEL CONFIDENTIALITY, #DATA MINIMIZE, #ALLOWED DATA, #SHORT RETAIN, #OBFUSCATE TRAINING DATA and part of #SECURITY PROGRAM
- Document your data, models and prompts (DSIT principle 8):
  Part of #DEVELOPMENT PROGRAM
- Manage your technical debt:
  Part of #DEVELOPMENT PROGRAM

3. Secure deployment

- Secure your infrastructure (DSIT principle 6):
  Part of #SECURITY PROGRAM and see 'Identify, track and protect your assets'
- Protect your model continuously:
  #INPUT DISTORTION, #FILTER SENSITIVE MODEL OUTPUT, #RUNTIME MODEL IO INTEGRITY, #MODEL INPUT CONFIDENTIALITY, #PROMPT INPUT VALIDATION, #INPUT SEGREGATION
- Develop incident management procedures:
  Part of #SECURITY PROGRAM
- Release AI responsibly:
  Part of #DEVELOPMENT PROGRAM
- Make it easy for users to do the right things (DSIT principe 4, called Enable human responsibility for AI systems):
  Part of #SECURITY PROGRAM, and also involving #EXPLAINABILITY, documenting prohibited use cases, and #HUMAN OVERSIGHT)

4. Secure operation and maintenance

- Monitor your system's behaviour (DSIT principle 12 and similar to DSIT principle 9 - appropriate testing and validation):
  #CONTINUOUS VALIDATION, #UNWANTED BIAS TESTING
- Monitor your system's inputs:
  #MONITOR USE, #DETECT ODD INPUT, #DETECT ADVERSARIAL INPUT
- Follow a secure by design approach to updates (DSIT Principle 11: Maintain regular security

updates, patches and mitigations):
Part of #SECURE DEVELOPMENT PROGRAM

- Collect and share lessons learned:
  Part of #SECURITY PROGRAM and #SECURE DEVELOPMENT PROGRAM

How about copyright?

> *Category: discussion*
> *Permalink: https://owaspai.org/goto/copyright/*

Introduction

AI and copyright are two (of many) areas of law and policy, (both public and private), that raise complex and often unresolved questions. AI output or generated content is not yet protected by US copyright laws. Many other jurisdictions have yet to announce any formal status as to intellectual property protections for such materials. On the other hand, the human contributor who provides the input content, text, training data, etc. may own a copyright for such materials. Finally, the usage of certain copyrighted materials in AI training may be considered fair use.

AI & Copyright Security

In AI, companies face a myriad of security threats that could have far-reaching implications for intellectual property rights, particularly copyrights. As AI systems, including large data training models, become more sophisticated, they inadvertently raise the specter of copyright infringement. This is due in part to the need for development and training of AI models that process vast amounts of data, which may contain copyright works. In these instances, if copyright works were inserted into the training data without the permission of the owner, and without consent of the AI model operator or provider, such a breach could pose significant financial and reputational risk of infringement of such copyright and corrupt the entire data set itself.

The legal challenges surrounding AI are multifaceted. On one hand, there is the question of whether the use of copyrighted works to train AI models constitutes infringement, potentially exposing developers to legal claims. On the other hand, the majority of the industry grapples

with the ownership of AI-generated works and the use of unlicensed content in training data. This legal ambiguity affects all stakeholders—developers, content creators, and copyright owners alike.

Lawsuits Related to AI & Copyright

Recent lawsuits (writing is April 2024) highlight the urgency of these issues. For instance, a class action suit filed against Stability AI, Midjourney, and DeviantArt alleges infringement on the rights of millions of artists by training their tools on web-scraped images2.
Similarly, Getty Images' lawsuit against Stability AI for using images from its catalog without permission to train an art-generating AI underscores the potential for copyright disputes to escalate. Imagine the same scenario where a supplier provides vast quantities of training data for your systems, that has been compromised by protected work, data sets, or blocks of materials not licensed or authorized for such use.

Copyright of AI-generated source code

Source code constitutes a significant intellectual property (IP) asset of a software development company, as it embodies the innovation and creativity of its developers. Therefore, source code is subject to IP protection, through copyrights, patents, and trade secrets. In most cases, human generated source code carries copyright status as soon as it is produced.

However, the emergence of AI systems capable of generating source code without human input poses new challenges for the IP regime. For instance, who is the author of the AI-generated source code? Who can claim the IP rights over it? How can AI-generated source code be licensed and exploited by third parties?

These questions are not easily resolved, as the current IP legal and regulatory framework does not adequately address the IP status of AI- generated works. Furthermore, the AI-generated source code may not be entirely novel, as it may be derived from existing code or data sources. Therefore, it is essential to conduct a thorough analysis of the origin and the process of the AI-generated source code, to determine its IP implications and ensure the safeguarding of the company's IP assets. Legal professionals specializing in the field of IP and technology should be consulted during the process.

As an example, a recent case still in adjudication shows the complexities of source code copyrights and licensing filed against GitHub, OpenAI, and Microsoft by creators of certain code they claim the three entities violated. More information is available here: : GitHub Copilot copyright case narrowed but not neutered • The Register

Copyright damages indemnification

Note that AI vendors have started to take responsibility for copyright issues of their models, under certain circumstances. Microsoft offers users the so-called Copilot Copyright Commitment, which indemnifies users from legal damages regarding copyright of code that Copilot has produced - provided a number of things including that the client has used content filters and other safety systems in Copilot and uses specific services. Google Cloud offers its Generative AI indemnification.
Read more at The Verge on Microsoft indemnification and Direction Microsoft on the requirements of the indemnification.

Do generative AI models really copy existing work?

Do generative AI models really lookup existing work that may be copyrighted? In essence: no. A Generative AI model does not have sufficient capacity to store all the examples of code or pictures that were in its training set. Instead, during training it extracts patterns about how things work in the data that it sees, and then later, based on those patterns, it generates new content. Parts of this content may show remnants of existing work, but that is more of a coincidence. In essence, a model doesn't recall exact blocks of code, but uses its 'understanding' of coding to create new code. Just like with human beings, this understanding may result in reproducing parts of something you have seen before, but not per se because this was from exact memory. Having said that, this remains a difficult discussion that we also see in the music industry: did a musician come up with a chord sequence because she learned from many songs that this type of sequence works and then coincidentally created something that already existed, or did she copy it exactly from that existing song?

Mitigating Risk

Organizations have several key strategies to mitigate the risk of copyright infringement in their

AI systems. Implementing them early can be much more cost effective than fixing at later stages of AI system operations. While each comes with certain financial and operating costs, the “hard savings” may result in a positive outcome. These may include:

1. Taking measures to mitigate the output of certain training data. The OWASP AI Exchange covers this through the corresponding threat: data disclosure through model output.
2. Comprehensive IP Audits: a thorough audit may be used to identify all intellectual property related to the AI system as a whole. This does not necessarily apply only to data sets but overall source code, systems, applications, interfaces and other tech stacks.
3. Clear Legal Framework and Policy: development and enforcement of legal policies and procedures for AI use, which ensure they align with current IP laws including copyright.
4. Ethics in Data Sourcing: source data ethically, ensuring all date used for training the AI models is either created in-house, or obtained with all necessary permissions, or is sourced from public domains which provide sufficient license for the organization’s intended use.
5. Define AI-Generated Content Ownership: clearly defined ownership of the content generated by AI systems, which should include under what conditions it be used, shared, disseminated.
6. Confidentiality and Trade Secret Protocols: strict protocols will help protect confidentiality of the materials while preserving and maintaining trade secret status.
7. Training for Employees: training employees on the significance and importance of the organization’s AI IP policies along with implications on what IP infringement may be will help be more risk averse.
8. Compliance Monitoring Systems: an updated and properly utilized monitoring system will help check against potential infringements by the AI system.
9. Response Planning for IP Infringement: an active plan will help respond quickly and effectively to any potential infringement claims.
10. Additional mitigating factors to consider include seeking licenses and/or warranties from AI suppliers regarding the organization’s intended use, as well as all future uses by the AI system. With the help of legal counsel the organization should also consider other contractually binding obligations on suppliers to cover any potential claims of infringement.

Helpful resources regarding AI and copyright:

- Artificial Intelligence (AI) and Copyright | Copyright Alliance
- AI industry faces threat of copyright law in 2024 | Digital Watch Observatory

- Using generative AI and protecting against copyright issues | World Economic Forum -weforum.org
- Legal Challenges Against Generative AI: Key Takeaways | Bipartisan Policy Center
- Generative AI Has an Intellectual Property Problem - hbr.org
- Recent Trends in Generative Artificial Intelligence Litigation in the United States | HUB | K&L Gates - klgates.com
- Generative AI could face its biggest legal tests in 2024 | Popular Science - popsci.com
- Is AI Model Training Compliant With Data Privacy Laws? - termly.io
- The current legal cases against generative AI are just the beginning | TechCrunch
- (Un)fair Use? Copyrighted Works as AI Training Data — AI: The Washington Report | Mintz
- Potential Supreme Court clash looms over copyright issues in generative AI training data | VentureBeat
- AI-Related Lawsuits: How The Stable Diffusion Case Could Set a Legal Precedent | Fieldfisher