



General Intelligence and  
Security Service  
*Ministry of the Interior and  
Kingdom Relations*

# AI systems: develop them securely





# Why should you pay extra attention to secure AI systems?



*An automatic scanner for the transit of goods that inadvertently passes weapons. An AI-based malware detection program that received incorrect training data and now no longer works. Attackers who manage to extract sensitive data from your AI system. All without you, as the owner, realising it in time. These are just a few examples of the potentially disastrous consequences of manipulated AI systems.*

Artificial intelligence (AI) gives computerised machines the ability to solve problems on their own<sup>1</sup>. More and more computer systems are using AI or incorporating ML models. Whether they're models for image recognition, speech technology or cybersecurity – AI can help you execute processes faster, smarter and better. Developments in AI are moving fast – so fast that it's important to develop your AI systems securely. Security is not something you can do afterwards; you have to think about it right from the start ('security by design'). Otherwise, you run the risk that your AI system will no longer work as it should, with all the consequences that entails.

In this publication, the National Communications Security Agency (NCSA) of the General Intelligence and Security Service (AIVD) shares ways AI systems can be attacked and how you can defend against it happening. We explain five different attacks that specifically target AI systems, also called *adversarial AI*. And we give five principles that help you to safely develop an AI system.

## What is the NCSA?

The National Communications Security Agency (NCSA), as part of the Unit Resilience, aims to keep the Netherlands digitally secure from nation-state threats and other Advanced Persistent Threats (APTs). We are unique in that we combine our expert security knowledge with the distinct intelligence position we have as part of the AIVD. We work closely with our security partners MIVD, NCTV, NCSC and the Central Government CIO Office. Together we help the central government, vital sectors and industry to protect special and sensitive information such as state secrets.

---

<sup>1</sup> Machine learning (ML) is a subdomain of AI and consists of algorithms that enable computers to learn from data. Training is the automated adaptation of a model by finding patterns in data. These algorithms can find complex patterns in data that are almost impossible to recognise for the human mind.



# This is how you keep control of your AI systems

AI systems have a different, additional attack surface compared to “traditional” digital systems. Thus, attackers can try to fool your AI models, sabotage the operation of the system, or figure out how your algorithms work without you realising it yourself. Therefore, it is not sufficient to implement general cybersecurity measures on your AI systems.

Using five types of attacks, we show how AI systems can be vulnerable. Additionally, we provide five principles for securely developing an AI model or system. The principles provide context and structure to help make informed decisions about system design, development processes, and help assess specific threats to an AI system. This publication intends to bring AI experts and cybersecurity experts together and enable them to collaborate within an organisation.

## Pay constant attention to quality and safety

There is no simple roadmap or checklist for developing AI systems securely. That would imply that you can follow this checklist and be ‘done’, without having to continue working on security. Therefore, the principles outlined here are drafted more as a mindset when you put your organisation’s AI experts and cybersecurity experts together. You can compare them to *secure coding principles*: principles to follow when writing “regular” code securely.

Security is important when designing your AI system, but also when your system is deployed. Developments in AI (and thus *adversarial AI*) are rapid, attack types continue to evolve in the future. Use the principles to consciously create or use secure AI models. We advise organisations to stay up-to-date of further developments in this field.



Currently, the NCSA classifies five different categories of attacks that specifically target AI systems:

1. Poisoning attacks
2. Input (evasion) attacks
3. Backdoor attacks
4. Model reverse engineering & inversion attacks
5. Inference attacks

Our five principles are:

- ✓ Ensure the quality of your datasets
- ✓ Consider validation of your data
- ✓ Take supply chain security into account
- ✓ Make your model robust against attacks
- ✓ Make sure your model is auditable





# Five attacks on AI systems

## 1. Poisoning attacks

With a poisoning attack, an attacker attempts to make adjustments to your data, algorithm or model so that the AI system is “poisoned” and therefore no longer works as desired. For example, spam filters that incorrectly classify malicious website links as benign. These types of attacks decrease the reliability of your AI system’s output.

Poisoning attacks occur during the training phase of a model. The attack can be performed, for example, by adding falsified data (data injection), manipulating existing data (data manipulation), or by disrupting the labelling process. Thus, the model learns to draw wrong conclusions. The algorithm or the model itself can also be affected – before, during or after the training phase. In order to do this, the attacker needs write permissions to (parts of) your data.

## 2. Input (evasion) attacks

An input attack, also called an *evasion* attack, is designed to manipulate the input to an AI system in such a way that the system performs incorrectly or not at all. Because the changes are often minimal and the attack is undetectable by the human eye, it means that in some cases, detection of this type of attack is very difficult. Consider traffic signs with intentionally added sticky notes that take on a completely different meaning, causing a self-driving car to perform unwanted actions.

Input attacks do not change anything about the AI system itself, but provide certain types of input that cause the system to make mistakes. Thus, the attack takes place when the AI product is already implemented and prevents your system from functioning properly when handling certain input.

## 3. Backdoor attacks

By building a *backdoor* into an AI model, an external party can add an additional branch in the decision tree, which can be used to determine the model’s ultimate decision for specific input. For example, an attacker does not want an automatic license plate recognition model to recognise the cars of a criminal organisation. He manages to gain access to the system where this model is developed and implements a backdoor that prevents license plates with a specific characteristic from being recognised. By applying this characteristic to the license plates, they then pass the scan every time.

It is possible that models you are using, but that you have not developed yourself, contain backdoors. Consider downloading an already trained model. You could train this one on your own data, but how the original model was trained is not clear. With that, a backdoor may already be present in the model. In some cases, a backdoor remains even if you retrain the model. It is very difficult, if not impossible, to detect a well-implemented backdoor in a model.

## 4. Model reverse engineering & inversion attacks

When *reverse engineering* an AI model, an attacker tries to figure out how your model works. For *inversion* attacks, the goal is to rebuild the dataset that was used to train your model. These data can contain sensitive information, which are potentially interesting for an attacker. The attacks can serve different purposes by, for example, stealing your intellectual property or investigating weaknesses in your model.

Reverse engineering of an AI model can be performed by sending many queries to the model, in order to map out its operation piece by piece. An attacker is then able to run the model in its own environment, and then search for further vulnerabilities using input attacks. The attacker can also investigate how your model reacts to a certain attack, to prepare a counter-reaction to that response. Model reverse engineering and inversion attacks are almost undetectable.

## 5. Inference attacks

*Inference* attacks are aimed at retrieving (potentially secret) training data. Models are often trained with large amounts of data that in many cases include personal data or intellectual property. Inference attacks investigate whether a piece of information originated in the training data based on the model’s output.

For example, say you want to know whether a person’s photo was used to train a facial detection model. With an inference attack, it’s possible to confirm this. Another type of inference attack could be that an attacker has partial knowledge of a person’s record (name, address, phone number) and wants to know the missing attributes (such as their bank account or social security number). If this person’s record was used to train a certain AI model, inference attacks could help the attacker find the missing information.



# Five principles for defending your AI systems

With these five AI-specific attacks in mind, and the high cyber threat in general, it is important to know how to protect your AI systems from them. NCSA has defined five principles for this, based on our own knowledge and experience, and information from our collaboration partners (such as TNO and NCSC-UK<sup>2</sup>). These principles are not one-on-one linked to the five attacks, and are complementary to general *best practices* for developing software and controlling your networks and systems.

For example, read our brochure:

- [Cyber-attacks by state actors - seven moments to stop an attack.](#)

The idea of our principles is that they help you think about how to safely develop and use AI models in your organisation. There is no universally applicable list of measures that will always keep you safe. Even when securing AI systems, risk management should be leading: what are the consequences if your model fails or is stolen? Based on this, determine what your level of risk acceptance is and where you need to take strict measures. Always consider our principles in the context of your organisation and the characteristics of your systems.

Our five principles are:

- ✓ Ensure the quality of your datasets
- ✓ Consider validation of your data
- ✓ Take supply chain security into account
- ✓ Make your model robust against attacks
- ✓ Make sure your model is auditable

## Ensure the quality of your datasets

Generally speaking, the quality of your data is very important when developing an AI model or system. With data quality we mean, among other things, how structured is your data? Is it known where the data comes from and can you check the quality, do you know it has not been tampered with? And also: can you detect elements in your data sets that have a negative impact on the performance of your model?

In addition, there are ways to strengthen your data that make it more difficult for an attacker to manipulate anything about your training data or input. Examples include *data transformations*, *gradient shaping* and *NULL labeling*. By paying close attention to your data quality, you improve the performance of your model and can counter poisoning and input attacks, among other things.

## Consider validation of your data

When you use data from external sources, you don't always know how that dataset was created. Therefore, it is important to validate the data properly. How was the dataset created? How do I make sure that I am not too dependent on this single source?

With externally obtained data, you often have no influence on the data. For example, a data provider may decide to add other labels, making your model less accurate. Therefore, it is important to continuously monitor and proactively control this where possible.

## Take supply chain security into account

As soon as a ready-made model is downloaded or set up for you by others, the chance of an exploitable vulnerability is imaginable. Counteracting a backdoor is very difficult if you cannot understand the model yourself. If you can build or assess the model yourself, introducing a backdoor becomes much more difficult for an attacker. That takes a lot of knowledge, skill and time.


Sometimes it is not possible to build a model yourself because you simply do not have the right kind or amount of data. Or because you lack access to the necessary computing power. In that case, make sure you have safeguards in place to trust the supplier. This is also called *supply chain security*: ensuring you have control over your suppliers and the quality of the products and services they provide.

For example, you can randomly check the externally sourced data or models for possible errors. You can also use techniques to ensure that incorrect and/or malicious data have as little impact as possible on your model: see also the principle on data quality. Furthermore, make as much use as possible of products and services from trusted suppliers.

## Make your model robust against attacks

The robustness of your AI model is the extent to which the model can function properly when facing anomalous inputs, changes in the data, or attempted abuse. All the principles in this handout will help you make your model more robust to attacks. This particular principle adds: make sure you train your model against possible attacks.

<sup>2</sup> Principles for the security of machine learning, NCSC UK. [www.ncsc.gov.uk/collection/machine-learning](https://www.ncsc.gov.uk/collection/machine-learning)



With *adversarial training*, for example, you can make sure that your model recognises modified, malicious data, and becomes resistant. This ensures that your model becomes stronger against adversarial inputs and is not affected by them. It is also possible to develop detection mechanisms that can distinguish manipulated input from desired input. In addition, you can protect your model and training data from attacks, such as by implementing *rate limiting* that limits how many times an action can be performed within a given time frame.

Finally, we recommend running *red teaming* exercises on your AI model. This involves having a 'red team' (with different extents of knowledge of your model) try to discover and abuse vulnerabilities in the model. For example by using the five methods of attack described above. Red teaming is widely used on traditional networks and is becoming more prevalent for AI systems as well. Red teaming can help you gain insight into the security weaknesses of your model.

## Make sure your model is auditable

An AI model provides predictions but it is often unclear how your model came to this prediction. If the model is already "working," it's hard to explain why exactly. Does your image recognition model really recognise horses in images, or is it being fooled by inconspicuous watermarks? What makes your model detect one specific type of malware, but not others?


If you already take the explainability of your model into account when building and training it, you will find that it becomes less like a black box. This field of work is called *Explainable AI*. For simple models, you can often easily figure out why certain choices are made. For more sophisticated models, this can also be done to some extent. If you yourself understand how your model works, you can also develop controls and test cases.

## Any questions?

Do you have any questions about developing your AI-systems securely?

Call us at: +31 79 320 50 50 and ask for NLNCSA or send an e-mail to [nbv-ai@minbzk.nl](mailto:nbv-ai@minbzk.nl).

We'd love to help you make your organisation more resilient.



---

**The NCSA's principles help you to look at your AI system with security in mind - and to take the necessary precautions.**

---



General Intelligence and Security Service  
P.O. Box 20010 | 2500 EA The Hague  
T +3179 320 50 50

February 2023