AI Technology Research

k

Guidelines for Secure Development and Deployment of AI Systems

kaspersky
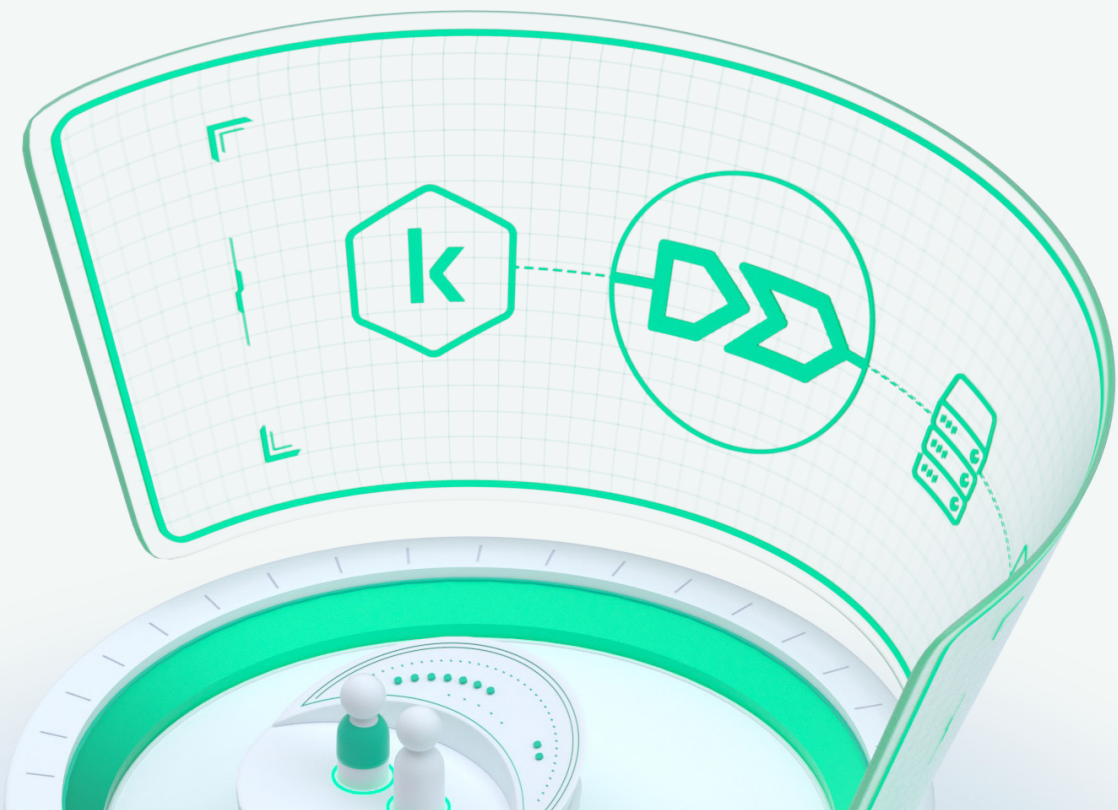
bring on
the future

Acknowledgements

Contents

Introduction

Purpose

New digital technologies come with **new cybersecurity risks and attack vectors.** Therefore, companies must ensure that the integration of AI is protected from these threats. The concept of security in the development of AI systems has been thrust to the forefront of various regulatory initiatives, such as the EU AI Act or Singapore's Model AI Governance Framework for Generative AI, to minimize the associated cyber risks. The EU is establishing strict AI regulations with the AI Act, which aims to ensure transparency, safety, and ethical standards. The U.S.A. is focused on developing industry standards and encouraging innovation rather than strict laws. China is actively formulating standards and regulations that support the development of AI technologies but also limiting their use for certain areas.

Despite this regulatory progress, important gaps remain between general frameworks and their practical implementation at a more technical level. In this paper, we explore the **basic cybersecurity requirements** that should be considered in the implementation of AI systems. These requirements should apply to a **broader range of companies relying on third-party AI components** to build their own solutions.

To implement AI safely, organizations need technical guidance on developing and deploying AI within their infrastructure. Implementing AI without proper guidance can pose significant risks. This document focuses on providing guidelines for developers and administrators of AI systems, MLOps, and AI DevOps, leveraging existing foundational models to create generalized AI solutions, with a particular emphasis on cloud-based AI systems. The paper addresses **key aspects of developing, deploying and operating AI systems**, including design, security best practices and integration, without focusing on foundational model development.

Artificial Intelligence (AI) has evolved into a critical technology for the global economy, becoming embedded in everyday life. AI enables organizations to automate routine tasks, enhance customer service and provide employees with quicker and easier access to information.

>50%

A recent Kaspersky study has revealed that more than 50% of companies have implemented solutions based on AI in their infrastructures*.

33%

are planning to adopt this technology within two years.

Overview of AI threat landscape

According to Kaspersky's research*, AI systems face unique and evolving security challenges that put the operation of the systems at risk. Malicious actors exploit vulnerabilities in training data, manipulate models to change their behavior, and compromise system integrity. This highlights the pressing need for comprehensive security in AI applications.

Issues with Model Development

Unlike traditional programming, where code behavior can be explicitly understood and tested, machine learning models, especially deep learning models with millions or billions of parameters, are inherently complex and often operate as a "black box." This complexity makes it difficult to fully predict and interpret the models' behavior. Therefore, the risk of undetected errors that can have serious consequences—such as the financial stability of a bank or even the life of a patient—increases significantly.

Another issue is the fact that AI models can sometimes base their decisions on **irrelevant or insignificant input data properties**, rather than on relevant features. For example, image recognition models might learn to classify animals such as cheetahs, leopards and jaguars by focusing solely on the patterns of their spots instead of using their overall anatomy**. In one case, a model misclassified a spotted sofa as a leopard because it associated the pattern of spots with the animal. Such misclassifications can lead to erroneous outputs in critical applications in such areas as healthcare, education, social welfare, transportation, the government sector, etc.

Inconsistencies between the data used for training and that encountered during deployment can lead to poor model performance. For instance, if a model is trained on data collected from one type of instrument but is applied to data from a different device, it may learn device-specific features rather than the underlying objects it is intended to recognize. This mismatch can lead to inaccurate predictions or classifications.

Both training models from scratch and fine-tuning foundational models can face such challenges. Although fine-tuning datasets are smaller and easier to manage, they may also introduce spurious correlations that cause the resulting model to misalign with its intended goal.

Threats to AI systems are growing as this technology is increasingly being deployed across organizations. Cyberattacks affect all stages of AI development, from datasets to algorithms and model outputs.

* AI under Attack, https://content.kaspersky-labs.com/se/media/en/business-security/enterprise/machine-learning-cybersecurity-whitepaper.pdf
** Suddenly, a leopard print sofa appears, https://web.archive.org/web/20200208171948/http://rocknrollnerd.github.io/ml/2015/05/27/leopard-sofa.html

Attacks on AI Models

Malicious actors can target AI models through various methods. Below are examples of how attackers exploit vulnerabilities in AI design, training, and interaction mechanisms:

Ways to attack	Description
Data poisoning: compromising model integrity	Data poisoning involves an attacker injecting malicious data into the training dataset to influence the model's behavior. By carefully crafting and adding poisoned samples*, attackers can cause the model to make faulty decisions or incorrect classifications on certain inputs. This type of attack can compromise the integrity of the model and undermine its reliability. This also applies to the fine-tuning of base models.
Adversarial attacks: invisible manipulation of AI	Adversarial attacks involve subtle modifications of input data that cause the AI model to misclassify it, while the changes remain unnoticed by humans**. Attackers add specially crafted noise to inputs, causing the model to produce incorrect outputs while the input appears unchanged to human observers.
AI data memorization: risk of unintentional exposure	Modern AI models can inadvertently memorize certain details from their training data, especially if the data contains unique or exceptional samples. Attackers can exploit this by using techniques to extract sensitive information that the model has inadvertently stored. This could lead to the exposure of personal user data or confidential business information.
Prompt injection: a threat to large language models	Prompt injection is a threat specific to LLMs such as ChatGPT. Developers program LLMs to perform tasks by providing initial prompts in natural language. Because users also interact with the model using natural language, the model cannot inherently distinguish between developer instructions and user inputs. Attackers can craft inputs that override or manipulate the model's behavior, causing it to perform unintended actions or disclose sensitive information. These malicious prompts can be entered directly by the user or embedded in the data that the model processes, such as documents or web pages.

These are only the most relevant attacks; a full description of all possible attacks on AI systems is beyond the scope of this document.

* Understanding Data Poisoning Attacks, https://bdtechtalks.com/2020/10/07/machine-learning-data-poisoning/
  Attacks Against Machine Learning: An Overview, (https://elie.net/blog/ai/attacks-against-machine-learning-an-overview/

** Explaining and Harnessing Adversarial Examples, https://arxiv.org/abs/1412.6572
  Adversarial Attacks and Defenses in Deep Learning, https://arxiv.org/abs/2201.06192
  How to Confuse Antimalware Neural Networks: Adversarial Attacks and Protection, https://securelist.com/how-to-confuse-antimalware-neural-networks-adversarial-attacks-and-protection/102949/

Traditional **Security Vulnerabilities**

AI models can be susceptible to traditional security weaknesses:

AI Vulnerabilities from Third-Party Resources

AI systems often rely on third-party models or datasets sourced from open repositories. These resources may contain unintentional errors or deliberate backdoors inserted by malicious actors. Incorporating compromised components such as these can introduce vulnerabilities into the AI system, affecting its security.

Supply Chain Risks in AI Development

The AI development pipeline can be vulnerable to supply-chain attacks, where attackers target the tools, libraries, or platforms used to build and deploy models*. This can lead to widespread security issues if not properly managed and audited.

Code Errors Exposing AI Vulnerabilities

Errors in the code of AI access interfaces can lead to vulnerabilities.

Risks of AI Component Theft

AI systems or their critical components, such as models or datasets, can be stolen without adequate protection as defined in this document.
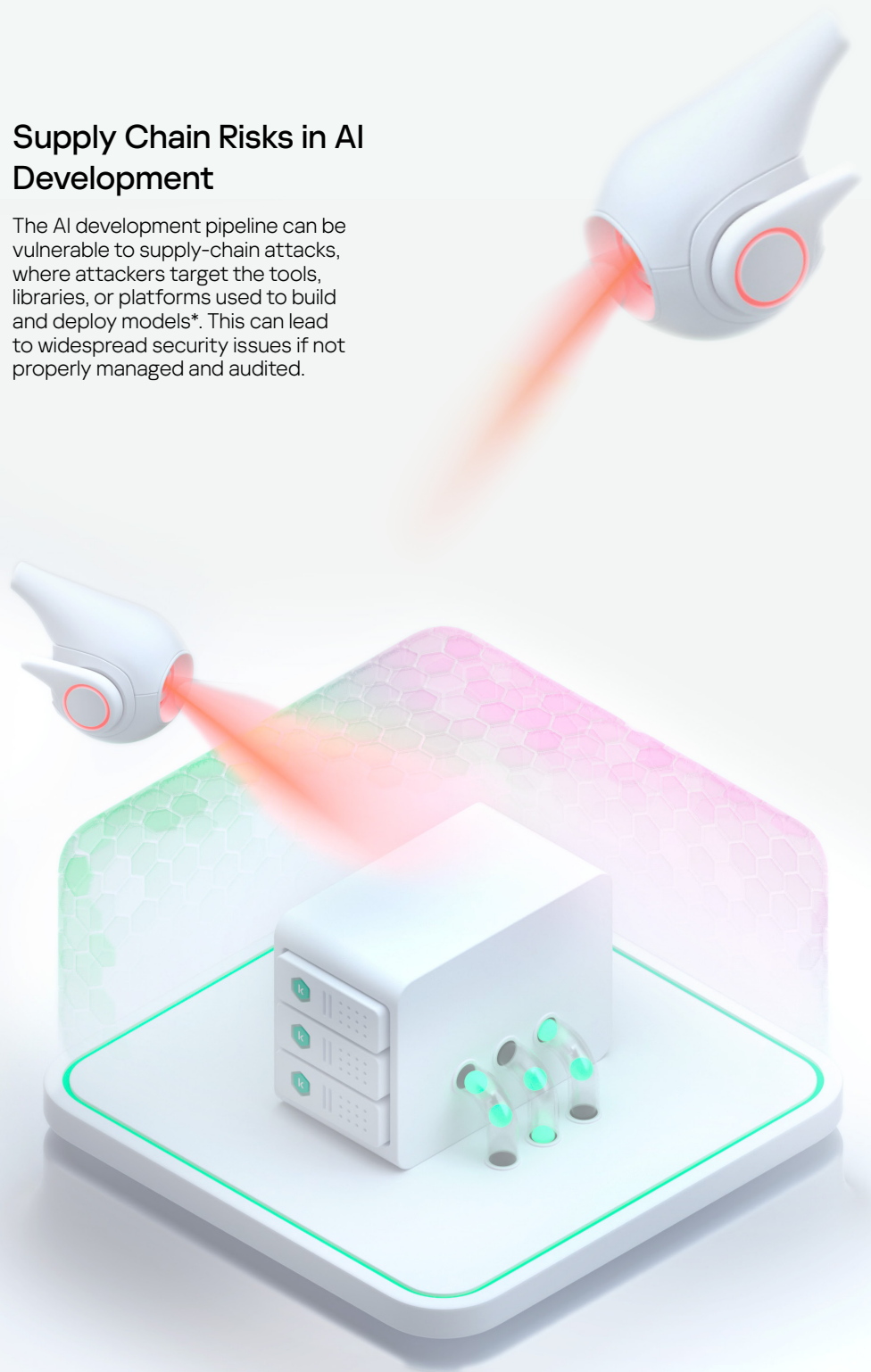
* Compromised PyTorch-nightly dependency chain between December 25th and December 30th, 2022, https://pytorch.org/blog/compromised-nightly-dependency

Guidelines

Cybersecurity awareness and training

The implementation of new technologies such as AI requires **leadership support, the establishment of internal policies and governance, and specialized training for employees on the risks and threats associated with AI.** The latter is critical due to the rapid evolution of this technology. Many of the AI resources available to developers are not mature enough, or do not fully embrace the security-by-default and security-by-design principles. As a result, an additional burden is placed on the developers of AI systems to address potential risks that need to be explained.

To this end, an organization should consider implementing the following measures in addition to their standard security practices:

1

The leadership of the organization needs to be aware of the security risks associated with using AI services and learn how to manage them.

2

The organization's security policies should be updated to address the specific requirements of AI services, ensuring that all employees and contractors are familiar with them.

3

The policy should outline the risks associated with developing and using AI services, as well as the current restrictions on their use in compliance with local legislation.

4

The policy should define roles and responsibilities pertaining to the use of AI services within the company.

7

Monitoring of legislation related to the safe use of AI services should be organized. Internal policies and training programs should be updated in a timely manner.

5

A corporate training course on the safe use of AI services within the company should be developed or purchased, ensuring that all employees and new hires complete it. The program should cover organizational policies, existing threats and incident examples, threat-protection measures, applicable laws, and other relevant topics, as well as include tabletop scenario-based exercises, if applicable. Employees should be tested when they complete the course. The course should be timely updated on a regular basis.

6

Existing information security courses should be updated to include new methods used by malicious actors that exploit AI services to attack companies, such as text generation. Examples of this include voice cloning, photo manipulation, or fake video generation.

Threat modelling / Risk assessment

This process is particularly important for AI systems, as this is an emerging technology with risks that are constantly evolving and adapting. Conducting a risk assessment can help stay ahead of these challenges. Additionally, threat modelling could help developers who are new to the field, better prepare for the challenges associated with developing AI systems. It will also allow them to identify and mitigate weaknesses proactively in the AI system before they are exploited.

To ensure an effective threat modelling process, an organization should adhere to the **following recommendations:**

Threat modelling helps identify, understand, and mitigate potential security risks in the early stages of AI system development.

Select a risk assessment methodology (e.g., STRIDE, DREAD, LINDDUN, PASTA, TRIKE) and develop procedures for conducting risk assessments and threat modelling for AI services. The assessment methodology should also include a framework for determining risk levels, strategies for managing risks within the organization (including acceptable risk thresholds), procedures for risk monitoring, and the assignment of responsible personnel to oversee the process.

A risk assessment should be performed for all existing and newly developed AI services.

A risk assessment should include the identification of potential threat actors or attackers, as well as identified threats and risks. Reference materials such as NIST-AI-600-1, MITRE ATLAS, OWASP Top 10 for LLM Applications, DASF, and CSA should be used to identify known threats and risks.

When managing risk, consider classifying threats into the following categories:

- threats arising from non-use of AI services,
- threats arising from non-compliance,
- threats arising from the misuse of AI services by users,
- threats to AI models and the datasets used for training,
- threats to services posed by AI models,
- threats to associated data, and
- threats to environmental, social and governance (ESG) performance.

Information about identified risks in AI services should be communicated to the organization's leadership.

Infrastructure security (cloud)

AI services are typically provided as cloud services, and often require specialized infrastructure, e.g., servers equipped with GPUs, FPGAs, ASICs, or TPUs. Given the sensitivity of AI systems, they should be protected according to **the most advanced cybersecurity frameworks**, such as the NIST Cybersecurity Framework or another with similar standards. AI services typically use open-source or free software like TensorFlow, PyTorch, or Keras, alongside libraries such as Pandas, NumPy, and SciPy. To secure this environment, the **following requirements** should be considered:

1

Identify all assets and maintain an inventory of information assets such as datasets for model training and testing, data for fine-tuning training, databases, data cards, models and risks, incoming and outgoing data to/from the service, model weights and hyperparameters, log data from LLM systems.

2

Control access at all levels, including network, operating systems, databases, software, data and models. Implement two-factor authentication (2FA) for administrative access.

3

Log all events and ensure log data is protected. Monitor for security incidents and potential breaches.

4

Implement safeguards against malware and other types of attacks. Apply security patches to infrastructure components on a regular basis.

5

Segment the network to protect sensitive areas. Use encryption for data in transit and at rest.

6

Ensure the integrity of critical data and verify the authenticity of libraries and models in use.

7

Provide server and communication channel redundancy. Perform regular backups and ensure their proper functioning.

8

Securely store keys in a KeyVault.

9

Apply the principles of least privilege and zero trust throughout the infrastructure.

Depending on the infrastructure supporting the AI services, additional requirements may include:

Use an API gateway to manage access to models and handle authentication via APIs.

Implement security measures specific to Kubernetes environments.

Adhere to best practices for securing cloud-based services.

Ensure the integrity of source code, training data, models, and automation scripts.

Isolate training data, models, and training environments to prevent data leakage or contamination.

Supply chain and data security

Supply chain attacks pose a significant threat to any organization's infrastructure, and AI architecture is no exception. There have been known instances where specialized libraries for neural network training have been targeted in such attacks. However, with AI, a specific concern arises regarding both the security of the service provider and the protection of machine learning models.

Access to advanced AI models, especially LLMs, often relies on cloud-based solutions. Nevertheless, the unavailability of certain models in certain regions, along with other constraints may prompt company employees and developers to opt for third-party services (proxies) that resell access to AI models via APIs for their daily tasks. This practice introduces significant risks — from opening up an additional vector for data leaks in the event of a security incident on the proxy service, to the unethical practice of misusing the obtained data for resale or to train its own versions of LLMs. **It is important to understand these risks,** carefully review the privacy policies of both the primary provider and the proxy, and conduct company-wide awareness training on the dangers of using third-party services for work tasks.

To mitigate these risks, an organization may choose to deploy a local LLM service. This approach ensures that confidential data processed by the LLM will remain within the company if certain specified conditions (e.g., network isolation, telemetry deactivation, etc.) are met. However, in addition to the risks associated with vulnerabilities in ML frameworks, this method entails threats related to backdoors in the models. In this context, it means that the data formats used to distribute machine learning models may have different security levels, and some formats potentially allow the embedding of arbitrary code that can be executed when the models are run. Research has shown that there are models available in public repositories, though limited in number, can execute specific code upon loading. The use of models from third-party repositories, instead of original ones, may be driven by the unavailability of downloadable models in certain regions or by licensing restrictions. The use of secure formats such as safetensors addresses this issue, but there is a need to raise awareness among developers and data analysts about the importance of selecting reliable sources for models and using secure formats to exchange model weights.

Ensure that AI models are obtained from reliable and legitimate sources. Avoid using third-party repositories.

Utilize secure formats such as safetensors to exchange model weights to avoid the risk of arbitrary code execution.

Implement measures to detect and respond to supply chain attacks on AI-related components.

Assess and review the privacy policies of third-party services and proxies used to access AI models to ensure they comply with security standards.

Deploy AI models locally under conditions that ensure data privacy, such as network isolation and the disabling of telemetry features.

Establish protocols for the secure deployment of local models to minimize the risks associated with potential backdoors in machine learning models.

Regularly update and patch machine learning frameworks to address known vulnerabilities.

Implement measures to ensure that sensitive data processed by AI models does not leave the organization's infrastructure.

When using third-party APIs, conduct security audits according to the top international standards, such as the OWASP API Security Top 10.

Testing and validation

Once the assessment has been carried out and the risks identified, it is crucial to understand how to guard against accidental or deliberate errors in the training and application of the model. To achieve this aim, an organization might consider implementing the following measures:

1

Assess the potential damage that could be caused by accidental or deliberate errors in the system. Evaluate the value of the data used to train the AI model and the data it processes.

2

Determine whether open-source frameworks, models, or datasets are being used to build the AI system.

3

Identify the potential user base: company employees, customers, or open public access.

4

Verify adherence to ML best practices in model construction. Ensure that the datasets are properly partitioned into training, test and validation sets based on how the model functions.

5

When validating AI models and their metrics (false positives and false negatives), check that the criteria for dataset splitting are appropriate for the nature of the data (e.g., chronological partitioning for temporal data, and avoidance of data leakage).

6

Assess which features the model uses to make decisions and whether they are consistent with the intuition of experts in the field. Use model interpretation methods, such as SHAP vectors, to understand the model's decision-making process.

7

Evaluate the real-world performance of the model to ensure that it is delivering the expected results. Monitoring of the model continuously, as the distribution of input data may change over time, potentially degrading the model's performance.

8

Adapt the test plan to check whether the model is susceptible to vulnerabilities unique to machine learning models (e.g., adversarial attacks and data poisoning).

Vulnerability reporting

AI is a relatively new area of technology and is evolving rapidly. Despite significant benefits, many AI systems are susceptible to vulnerabilities specific to these technologies.

One of the main concerns is that some AI systems may contain vulnerabilities that can be exploited to gain unauthorized access to their data. Another example of vulnerabilities in AI systems is bias, where models are trained on data that is unrepresentative or contains hidden biases. For instance, AI systems can be impacted by prejudice bias when stereotypes and faulty societal assumptions infiltrate the algorithm's dataset, or measurement bias provoked by incomplete data. As a result, these systems may make unfair or discriminatory decisions, negatively impacting users and undermining trust in AI technologies.

To address these issues, it is necessary to implement a mechanism that will allow users **to report identified vulnerabilities** and biases in AI systems. This reporting mechanism will enable organizations to receive feedback quickly and take action:

1

Establish a publicly available policy that defines vulnerabilities in AI systems and outlines how users can report them.

2

Provide secure methods for users to report vulnerabilities, such as encrypted web forms or dedicated email addresses.

3

Define procedures to promptly assess, prioritize, and remediate reported vulnerabilities.

4

Communicate with the person who reported the vulnerability regarding the status and resolution of the issue.

5

Keep users informed of known vulnerabilities and remediation efforts to build trust and demonstrate accountability.

6

Collaborate with security researchers through bug bounty programs. This will also help you stay abreast of emerging threats and AI security best practices.

Defense from ML-specific attacks

Given the current advancements in AI development, some of the AI components may be vulnerable to ML-specific attacks. These attacks can exploit vulnerabilities by deliberately feeding malformed data or hidden commands into the model, for instance. Organizations using free AI to build their systems should therefore be aware of these risks. Protecting against ML-specific attacks requires **the implementation of various security measures**, such as:

Incorporating adversarial examples* into the training dataset to help the model learn how to handle these inputs more effectively.

Applying distillation techniques that help make the model more resilient to adversarial inputs by simplifying its decision-making process.

Considering the use of monotonic models**, which can improve stability and reduce susceptibility to adversarial manipulation.

Introducing systems that can detect adversarial or anomalous inputs in user requests, allowing the model to detect and reject malicious attempts before processing the data.

To protect against data poisoning, AI system developers should **analyze the training samples** for anomalous objects and compare the performance of new models with previous versions to identify any sharp changes in the model's properties.

To protect against prompt injection attacks on the LLMs, AI system developers can implement a system that analyzes incoming user requests or other third-party data fed into the LLM input. Another approach is to analyze the responses to such requests and assess their compliance with the current task of the system.

* How to confuse antimalware neural networks. Adversarial attacks and protection, https://securelist.com/how-to-confuse-antimalware-neural-networks-adversarial-attacks-and-protection/102949/
** Monotonic models for real-time dynamic malware detection, https://arxiv.org/pdf/1804.03643

Regular security updates and maintenance

The field of AI, and in particular the application of LLMs, is still relatively young, and code quality is not always of a high standard. As a result, many frameworks and tools used to work with machine learning may contain a **significant number of vulnerabilities**. Fortunately, we are now in a phase where popular frameworks are being actively brought up to production-quality standards, with regular releases and security updates. Additionally, the emergence of bug bounty programs aimed at finding vulnerabilities in ML infrastructure are emerging, helping to address these issues quickly.

This underscores the importance of continuously monitoring the state of your infrastructure, from platforms used to track experiments to libraries designed to communicate with cloud services. Keeping the infrastructure up to date may take more time than necessary for projects within slower-evolving fields. Furthermore, using the latest versions of libraries can lead to compatibility issues, requiring greater investment in developing and maintaining the functionality of code that relies on them. These costs need to be factored in when planning AI initiatives.

A separate risk associated with using cloud-based AI models, such as LLMs, is the relatively **short lifecycle of each version of the model version**. The model selected for a project may be replaced by the platform provider with a new version within a short period of time. While the overall quality of the models is expected to improve, their behavior for specific tasks and their resilience to attacks may change. It may, for example, include prompt injection or attempts to obtain unauthorized output via jailbreaks. This requires advanced planning to ensure a smooth transition between models without compromising the quality of downstream tasks or the level of security:

1

Ensure that the infrastructure is kept up to date with the latest security patches and framework updates.

2

Actively participate in bug bounty programs and use vulnerability scanning tools to detect weaknesses in ML frameworks and AI infrastructure.

3

Regularly review and apply security updates for machine learning tools and libraries to reduce exposure to known vulnerabilities.

4

Plan for potential compatibility issues when using the latest versions of libraries and frameworks by allocating development and testing resources.

5

Implement a strategy for managing the lifecycle of cloud-based AI models, with plans for transitioning to new model versions as they are released by the provider.

Compliance with international standards

As we witness the rapid development of AI regulation, compliance with relevant laws and adherence to best practices are becoming increasingly important. First, AI training data can be collected from multiple sources in different jurisdictions, which makes it difficult to process and to use this information. Furthermore, models are often sourced from **open repositories**, adding a level of uncertainty to their conformity with a particular jurisdiction's regulatory requirements.

Thus, AI developers are faced with the difficult task of ensuring compliance with all legal requirements in the countries where the system will be used. The best strategy in this situation is **to follow the standards of leaders in AI regulation**, such as China, the European Union, or the United States. Many countries are already sharing their approaches and implementing similar requirements, allowing developers to prepare in advance for the global deployment of the system:

1

Establish guidelines for the ethical use and development of AI to ensure transparency and accountability in related processes.

2

Ensure that all data collected from disparate sources complies with data privacy laws in each jurisdiction, such as the General Data Protection Regulation (GDPR) in Europe or the California Consumer Privacy Act (CCPA) in the U.S.

3

When using AI models from open repositories, verify that they comply with intellectual property rights.

4

Follow leading regulatory frameworks, such as the European Union's AI Act or the U.S. AI Bill of Rights, as these are often used as benchmarks by other countries.

5

Stay abreast of new and evolving AI regulations worldwide.

6

Regularly audit AI models and systems for compliance with international standards to identify and mitigate potential legal and ethical risks.

Conclusion

Like most technological innovations, AI technologies present both great opportunities and significant threats. The cybersecurity risks associated with AI and its impact on society depend on the behavior and intentions of the developer. To implement AI safely, organizations need to follow technical guidance on how to develop and deploy AI in their infrastructure, as carrying out this process without proper recommendations can pose significant risks. It is vital for organizations to establish a culture of security and responsibility throughout the AI lifecycle and incorporate basic security controls, from risk assessment and system testing to securing supply chains and ongoing maintenance. Successful implementation of the presented requirements will help **mitigate the risks** related to the introduction of AI systems into a company's operations.

AI
Technology
Research

k

Learn more

#kaspersky
#bringonthefuture