Deep Science Publishing
https://doi.org/10.70593/978-81-988918-1-5_5

DeepScience
Open Access Books

Chapter 5: Building secure artificial intelligence systems: Defending against vulnerabilities in intelligent technologies

5.1. Introduction

Given the increasing capability and applicability of AI systems in sensitive domains within society, we, cyber and information security specialists with a long-standing interest in critical computer systems, must extend our mission to include those systems dedicated to Artificial Intelligence. We must ensure, to the degree feasible, that AI systems function dependably and securely when deployed. After years of pushing back decades of optimism that had located AI systems beyond our field of study, a realistic attitude toward the considerable benefits and, equally, the considerable dangers that AI systems can engender has emerged. While the goal of designing such systems so that they reflect or generate intelligent behavior in a quantifiable way has regained attention, our focus here is on their security. AI systems are vulnerable to a set of attacks that differ on key dimensions from the traditional attacks against conventional computer systems. We refer to this set of attacks as the "AI Security Vulnerability Landscape." Some of the vulnerabilities of non-AI systems are also present in AI systems, but heightened or modified. In this chapter, we summarize the kinds of vulnerabilities that we feel are most salient. We also consider some new ideas, surprisingly longstanding in some contexts, such as verification of generated behavior. Our particular focus is defensive activities (Huang et al., 2011; Goodfellow et al., 2014; Biggio & Roli, 2018).

To keep our focus limited, we restrict our attention predominantly to Machine Learning, the most visible AI activity. Most of the vulnerabilities that we would summarize for AI systems more generally are also the most relevant for Learning Systems. However, the types of intelligent systems that present other forms of weakness are somewhat broader than the kind of supervised or unsupervised learning through repetition, with a focus on generating probability distributions over symbol strings, that presently dominates in practice. For example, the increasingly popular area of Ontology-based Systems for Knowledge Representation and Generation raises different issues than those affecting Learning Systems. Other logical activities, such as planning via deriving deductions, not already covered, also require distinct emphasis (Moosavi-Dezfooli et al., 2016; Papernot et al., 2016).

5.2. Understanding AI Vulnerabilities

Threats and vulnerabilities are the backbone of security. Understanding the vulnerabilities of AI systems is fundamental to building secure AI systems. This section analyzes the vulnerabilities of AI systems and how these vulnerabilities are different from other existing systems.

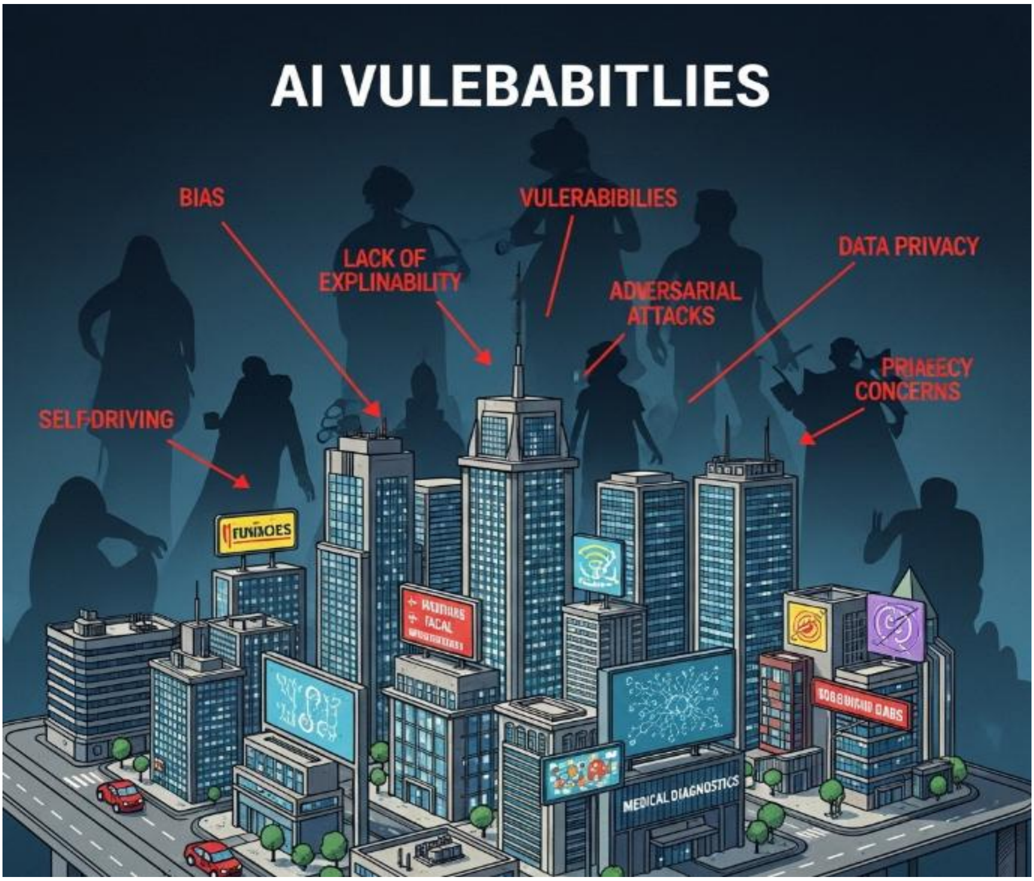

Fig 1 : AI Vulnerabilities

Threats are assumptions on the capabilities and intentions of the enemy. When we say an AI system is secure, we are making a statement regarding the vulnerabilities of AI systems. A vulnerability of AI systems are weaknesses in a system that can be exploited with a malicious intention to harm. Although the common notion is that AI vulnerabilities pertain only to adversarial examples and failure of such models during generalization, it is much more than that.

Intelligent technologies capture models which in some sense capture the essence of the problem, and any violation of that model either due to program bugs, model capacity issues, bad data or lack of domain knowledge can become sources for vulnerabilities for your system. Security keywords like confidentiality, integrity, availability, timeliness, non-repudiation, feasibility, maintenance concern legal and ethical aspects are all relevant to the vulnerability and security of AI systems. In many ways, it is similar to constructing normal systems based on first principles but in the case with intelligent systems, it is more about what other choices you have of the project itself, the computing environment, the sensing and actuating aspects, the prior domain knowledge and the model in order to cause the model to become vulnerable and thus give a system that is not secure.

5.2.1. Types of Vulnerabilities

To the best of our understanding, the term “vulnerability” has been defined in the following ways. A vulnerability is some feature (or property) of a system that allows an adversary to bypass its security mechanisms, resulting in the violation of any of the security properties: integrity, confidentiality, and availability. Vulnerabilities also are the points of weaknesses within artifice; hence by exploring them, one can reduce the trust in the artifice itself. A vulnerability can also be defined as a weakness, flaw, or error in a program or component that could be exploited to breach the system security policy which may result in a violation of any of the security properties.

This definition differs from that of an exploit or a risk. An exploit is a piece of code or a sequence of commands that takes advantage of a vulnerability and triggers it; hence, it is not a weakness within the system. A risk is the probability of a threat-source exploiting a vulnerability, as well as the resulting impact of a threat, which may result in a violation of system security policy. A vulnerability, a risk, and an exploit are different things, but they are linked together. Any risk associated with an exploit will depend on the existence of an underlying vulnerability. Without an identified and classified vulnerability, it is difficult to assess the risk levels assigned to an exploit or a threat. Only through the identification of vulnerabilities can the business decide how to mitigate the threat associated with an exploit or a risk. It is essential to determine policy decisions for system configurations according to the identified vulnerabilities. Many governmental and private organizations review the security vulnerabilities of computer systems and disclose them with classified information.

5.2.2. Impact of Vulnerabilities on AI Systems

This section considers the impact of security vulnerabilities on AI systems. To put this in perspective, we first briefly examine the adversarial impact considered in defeatism. This is then followed by a more practical consideration of the impact of security vulnerabilities in making AI systems impossible to deploy. Both perspectives capture different types of impact on vulnerability presence and exploitation in AI systems. A typical object in computer security is the theft, destruction, modification, or denial-of-service against the services of a vulnerable system. However, many computer security vulnerabilities are just that – vulnerabilities in a computer program that is but one component of a larger, interacting software ecosystem. In this latter situation, while the existence of the vulnerability may impact the security of the program, its existence does not have an associated adversarial impact on that program.

Are there other potential adversarial impacts from vulnerabilities in specialized AI programs, libraries, and networks? The answer is yes. In particular, many of the popular neural-network libraries function as general-purpose AI engines allowing other developers or organizations to build their own application-specific AI software. For example, a famous photo-sharing application makes use of a neural-network library to allow users to apply filters to their photos. From a computer security perspective, there are huge concerns here. There is already a growing trend to package AI capabilities in web applications. In a scenario where organizations package AI capabilities, traditional web application vulnerabilities such as cross-site scripting and SQL injection attacks could come into play. An attacker could take advantage of a cross-site scripting or SQL injection vulnerability to corrupt the parameters of an AI library call or AI macro. Are there ways to mitigate? These concerns point toward a growing movement toward the development of secure AI libraries using secure inputs, checks, and handling for model transitions.

5.3. Threat Models in AI

When assessing risks and evaluating mitigation strategies for any technology, threat modeling can promote comprehensive consideration of significant potential harms alongside possible weaknesses. Such modeling is critically important in the AI domain, where there is ample opportunity for malicious use or for threats to arise from vulnerabilities in deployed systems or technologies. However, threat models used for other domains often do not have a comprehensive mapping to the classes of threats or vulnerabilities that arise from AI, given the unique characteristics of certain NLP, computer vision, and other systems. This lack of parallels between other threat models and the AI domain creates the potential for insufficient attention to significant risk areas when considering AI technologies. Presenting carefully constructed threat models that are reflective of threats and vulnerabilities unique to AI deployment can help fill this gap. This section discusses aspects of threat modeling in the AI area, describing important concepts that are necessary for useful modeling. First, we discuss how to identify potential threats and adversaries unique to the AI domain. We also discuss how to assess risk factors for general threats and expected defenses. Threat models provide critical foundations to framing adversarial considerations, and model expected points of influence on risk factors. Discussion of threat modeling is followed by a brief insight into the specialized area of vulnerabilities in AI technologies, and a high-level overview of potential defenses for these vulnerabilities.

5.3.1. Identifying Potential Threats

Any analysis of potential AI threats must be grounded in a general understanding of how AI capabilities are enabled, shaped, and animated. In this section, we take a functional view of AI systems, asking what sorts of processes, resources, and component technologies are needed to enable the system to do the work for which it is designed. We consider not only the core algorithmic components of the system, but also key components that are specific to the AI system's intended domain, such as data inputs, model training and retraining, and communications channels.

Based on this functional decomposition, we characterize threat classes based on what types of actions potential adversaries can take against the component features of an AI system. Exploits associated with particular threat classes are then assembled into a threat model. The threats shaping the threat model may be motivated by harm to the targets of the AI system's decisions, but they may also be motivated by adversary-specific goals associated with the states of the AI system itself. For example, an adversary may manipulate an AI system to achieve a goal associated with the operation of the system, such as altering the AI decisions of a trusted party, or they may be motivated to achieve an adversary-goal associated with security-related concerns for the AI system itself.

5.3.2. Assessing Risk Factors

create a human-centered distributional structural risk assessment framework to improve the assessment of risk in machine learning. Several other AI-specific risk assessment frameworks exist. Each framework is tailored to different facets of AI systems, such as their intended functionalities, their impacts on the environment in which they operate, or their learning process; hence they can be used in a complementary manner.

Building on prior AI risk taxonomies and our understanding of the principles, components, and functionality of an AI system, we design an iterative assessment of risk factors for responsible AI systems. This scenario-based stress-testing guidelines aim to address the unknown-unknowns in risk assessment. Our guidelines can be broadly envisaged in five steps. The first three steps are the basis of risk assessment for all AI systems. Steps 2, 4, and 5 are responsible for identifying possible emergent behaviors, resulting from known unknowns and unknown unknowns of AI behavior. "Initializing" the AI scenario involves gaining a proper understanding of all the possible inputs to and accepted outputs from the AI system; setting the initial and terminal scenarios or states can involve examining how the state of the environment changes when using the AI system. However, since AI systems perform actions based on dynamic assessments of the state of the environment and learning from its history of states and corresponding actions, it is impossible to identify all the initial and terminal states. It is more of a maximal and minimal initial and final step within a certain time period before and after executing the AI system.

5.4. Secure AI Development Lifecycle

When writing a software program, it is imperative to include security from the beginning. Yet, for most software engineers, the concept of application security is an afterthought. Companies that use off-the-shelf models typically rely on the model assessment data of others to validate a model’s integrity. This is clearly not a viable option for companies that are developing their own models, such as models for automated driving or security applications. Architects and engineers must bake security into their applications from the very start. While they typically think about security requirements at the beginning of the project, these may differ depending on whether you are employing AI techniques to solve your problem, or if the entire solution is based on AI. The difference between Traditional SDLC and Secure AI Development Lifecycle is highlighted.

The baseline for most solutions is code. In fact, the average cost of in-house developed code for a company is between 50–80%. Since AI solutions are built on algorithms whose purpose is to mimic intelligent human behavior, it is unthinkable to write these algorithms without taking into mind potential abuse, either through using the model for malicious purposes, or to manipulate the input or output of the model. For instance, after years of development, one of the largest in-car AI solution providers had to sideline an entire product line when it was discovered that the systems could be easily confused by strategic placement of duct tape on the car's bumper that would not be noticed by a human, but would result in the car taking undesirable actions, such as braking when no obstacle was present.

5.4.1. Planning and Design

The traditional system development lifecycle (SDLC) provides a framework for any software project. The phases of the SDLC can be summarized as: (1) formal requirements specification and analysis; (2) preliminary design; (3) detailed design; (4) implementation; (5) testing; (6) deployment; and (7) maintenance. Security issues can be considered in each SDLC phase and, additionally, lessons learned in the security testing stage can inform future projects and future phases in the current project, especially during maintenance.

Considering security during the early phases of a project traditionally occurs as threat modeling. Self-supervised (SS) and foundation model (FM) approaches in machine learning provide a generalized basis for most future applications, such as conversational agents. However, even with enterprise implementations of these tools, there are still substantial investments in customizing these base models to specific tasks and applications, especially in tuning and validation for information security.

Appropriate requirements for training and evaluation must be defined for security. These requirements will differ significantly from the performance metrics used for unconstrained ML, such as loss metrics or traditional testing metrics. Discussion explores general requirements for security, privacy, and trust to define appropriate metrics for the planned use of an intelligent technology. The verified-design paradigm is often proposed for hardware and software secure systems and can be classified within these guidelines. The recent expansion of "security" for adaptive systems to encompass fairness, privacy, accountability, interpretability, and value alignment makes secure by design of autonomous intelligent systems a research project requiring close collaboration across multiple disciplines.

5.4.2. Implementation Best Practices

As with any software project, bugs can be introduced throughout development. AI models are difficult to audit, and there may be many more potential points of failure than in typical software. Further, AI models can exhibit surprising or dangerous behavior in production. Security teams must be included in all stages of the development process so that all discussions, designs, architectures, threats, threat modeling, and design reviews treat security as a first-class citizen. Teams should strive to increase communication as much as possible. Security teams may not be AI experts, but they could be experts on security. AI teams may not be security experts, but they would be the experts on their own models. Only with open communication can the best decisions be reached. For almost all models, application-dependent safety-checking and risk mitigation should be considered for production. This can take the form of human-in-the-loop processes. It may also take the form of automatic safety checks or throttling based on the prompt input and/or output content. For very high-impact models, like large-scale dialogue models, safety may warrant the use of complete black-box input and output filtering trained on an entirely different risk dataset. During development, continuously create a corpus of failed and misbehaving queries. Involve your offensive security team in testing and failure interrogations. Building adversarially robust models is very hard, so adding such thresholds may be necessary until the safety is satisfactory—not just safe enough for initial production, but also safe in the long run.

5.4.3. Testing and Validation

Software testing and validation is concerned with ensuring that an implemented system satisfies its requirements and behaves correctly. Independent of the underlying technology, there are broadly two stages of testing: early testing (in which informal testing methods, such as review or inspection, are applied) and full testing (after the system is formally complete). Research in empirical software engineering suggests that static analysis during initial design phases can uncover as many as 25% of defects at the lowest cost, while software testing at later stages provides for the largest overall cost savings. In the rest of this section, we summarize the best practices for both early testing and full testing and validation.

AI systems, however, pose special challenges to the application of standard testing and analysis tools and technologies. These challenges stem from the fact that intelligent systems often paper over significant design defects with the help of novel static and empirical methods that rely on training. Considerable effort must be expended to either construct a working final implementation or to exert sufficient layer upon layer of testing during component-level implementation to make the production system dependable. Further, the validation of learned systems requires extensive scrutiny that is often difficult to gain consensus around and which is just as likely to depend on common sense or heuristic guided by the application itself rather than the science of software engineering. Techniques and principles frequently used for the analysis and validation of classical software components — such as logical verification, theorem-proving, symbolic execution, and model-checking — can be hindered by the number of internal states or potential executions of the intelligent system, especially if it is actuated with other non-intelligent components.

5.5. Data Security in AI Systems

The concerns about data security and privacy impacts to AI systems have awakened. Concerns regarding data security when developing AI systems remain. AI systems are hungry for massive amounts of data to train models. Organizations leverage data from various sources either intentionally or unintentionally. In most cases, organizations do not evaluate or check the data that are being brought in to build AI systems. Model training with unverified data could save time and money for the organizations, but presents a risk. Moreover, the data used for training AI models could be confidential. Especially when developing AI technologies for healthcare and financial domains, organizations must check data for confidentiality using proper Data Protection Techniques or place secure measures. Protecting the data at all stages of the AI system lifecycle is imperative, from data scraping through data storage and labeling to model training and securing the models. All generative AI systems need to implement data protection to prevent leakage of the knowledge learned during training.

Fig 2 : Data Security in AI Systems

Certain data protection techniques enable protection against sensitive data, e.g., surrogate data, data masking, or differential privacy. They prevent the AI system data from being exposed or leaked. Surrogate data technology permits organizations to keep their sensitive data confidential while still receiving a high-quality model. When training a neural net model to generalize over a population of distributions without seeing the underlying sensitive distributions, surrogate data offers the organization confidentiality without sacrificing model performance for various populations. Other data protection technologies are needed during the model prediction phases of the AI services, including masking, tax capping, scaling, and cryptographic building blocks such as homomorphic encryption to prevent unauthorized access to the AI output and the parameters from external model queries.

5.5.1. Data Protection Techniques

Advances in intelligent technology, including AI, natural language processing, and sensitive data availability, make it easier for individuals, organizations, and nation-states to probe and exploit the sensitive secrets of others. For example, social engineering is one of the oldest tricks in the cybersecurity book. Yet, some of the most advanced and pervasive forms of social engineering can now occur entirely through automated processes, including backend conversations with AI systems. With no need for socializing, these AI systems can browse and enumerate an organization's data and security weaknesses, and probe organizational insiders for sensitive data or gain access to sensitive systems. At worst, these "AI-aided" social engineering techniques can compromise and control entire organizations by gaining and exploiting a single user's access without that user ever knowing they were compromised.

To mitigate the risks posed by new AI-assisted social engineering techniques, novel data protection techniques are required. These techniques must capture the very different ways in which organizations operate today. For one, unlike in the past, there is no longer a perimeter around sensitive data. Sensitive data resides in data centers and cloud services operated by third-party companies, at endpoints relying on unsecured networks, and even in transit between organizations whenever third-party companies are involved in data exchanges during sensitive transactions. Moreover, employees are working from anywhere. Organizations can no longer keep an eye on what employees are doing to sensitive data. Finally, sophisticated data-exposed organizations are resorting to highly automated data and security operations. Unlike in the past, these operations can no longer afford to stop because a user is not at his or her physical workstation, unable to authorize daily data operations. Balancing security with convenience has never been more complicated.

5.5.2. Privacy Concerns

Privacy violations are a key concern for users of AI systems, particularly general AI systems that rely on the internet to gather, store, and process information. Privacy risks often stem from the volume of sensitive data that an AI collects and processes, including biometrics, location, and personally identifiable information. Additionally, a bias in the training set can lead to skewed predictions, further threatening user privacy. Even when collected for benign machine-learning purposes, an adversary may be able to exploit sensitive data. Because of the sensitivity of the data being processed and the tools they leverage, there are also strict legal and ethical requirements for Privacy by Design and Lawfulness, Fairness, and Transparency for AI systems. Users expect AI systems to respect these laws by not collecting any data without express consent or that may be harmful, sampling, and sharing any sensitive data. Fortunately, there are a variety of data sanitization and anonymization tools available to AI developers. When designing a secure AI system, developers should keep in mind that security promises do not necessarily imply privacy. A secure AI system may protect sensitive data from unauthorized access or tampering; however, if the data is maliciously extracted, it can still violate user privacy. Conversely, privacy does not guarantee security. A user might upload de-identified facial images, thinking it won't violate privacy; however, if it is later matched with other attributes, the user may suffer from privacy violations. Accordingly, developers need to carefully consider the information being processed in an AI system to avoid the unintentional sharing and leakage of sensitive data, including the model, the features being processed, and the output predictions. Most importantly, security and privacy measures should be insured throughout the complete AI lifecycle.

5.6. Adversarial Attacks on AI

Adversarial examples have taken the field of AI by storm: people have flocked to them for numerous reasons, with the underlying theme being the intuitive fact that if we cannot make AI systems robust against carefully designed input perturbations, what hope do we have for making AI systems useful in our lives? While the field is innovative and active, it has also resulted in some of the relatively most well-known scandals in applied AI, where well-designed perturbations can overcome entire pipelines for state-of-the-art facial recognition or publicly visible topic classification of deepfake videos.

A decade ago, adversarial examples were but a rather abstract concept, replete with somewhat baffling supporting explanations and justification of their existence. Adversarial attacks have since matured, grown dramatically in sophistication and variety, and become tightly integrated into the mainstream AI community—so much so that many of the leading journals and conferences in the field expect that their ongoing submission cycles will contain contributions that deal with adversarial robustness or perturbation attacks. Most importantly, perhaps, the vast majority of work into designing adversarial attacks and methods of ameliorating against them are fundamentally on supervised learning classifiers. Many adversarial settings much more deeply relevant to cutting-edge deployed AI pipelines are underexplored; these include sequence transduction of various sorts, generation of various sorts, or even AI systems for modeling such diverse modalities and tasks as video and audio/movie generation using modern flow methods.

5.6.1. Understanding Adversarial Examples

Introduction Neural networks have now achieved astounding results on a variety of classification benchmarks. With their unprecedented performance, a number of researchers have proposed to use neural nets for security-critical applications, e.g., for credential stealing via malware analysis, securing biometric identification, or scanning for malicious ports. Despite the success of these models, considerable skepticism regarding their overarching security has been expressed. A substantial amount of work has shown that the predictions obtained by deploying a neural classifier can be easily manipulated, potentially nefariously, often without significant effort and seemingly without any viable defense mechanisms. In fact, several empirical studies showed that aspects of the neural architecture itself and the training – and by desired extension, the feature extraction – methodology influence the model's susceptibility to manipulated inputs. This research has pioneered an entire sub-field of machine learning whose goal is to explore the security of neural nets. Perhaps its most recognizable result are so-called adversarial examples: semantically coherent inputs which, despite being very similar to training examples, cause a specific prediction error. Images intentionally manipulated so that a human would perceive a cat on the left but a neural classifier would predict a given object class on the right. Given their potentially devastating impacts, the creation of adversarial examples, and machine learning classifiers' vulnerability in general, has also received considerable media attention, as they are relatively within reach of any malicious actor without defense mechanisms to counter input manipulations. Adversarial manipulation is not restricted to image-based inputs.

5.6.2. Defense Mechanisms Against Attacks

Several techniques have been proposed to defend against AEs. In this section, we discuss the techniques to defend against adversarial attacks for classifiers, especially for neural networks. Note that the defense mechanisms can be classified into four categories: (1) Data preprocessing, (2) Adversarially trained robust classifiers, (3) Model tweaking, and (4) Autonomous decision systems.

Data preprocessing: The main idea is to preprocess the inputs to remove the perturbations before feeding the data to the classifiers. For example, one can use several image processing techniques such as bilinear interpolation, JPEG compression, or total variation minimization for denoising the input images. While trivial input transformations such as resizing and JPEG compression may be inconsequential, it has been shown that more complex preprocessing transformations may incur a significant performance drop on the clean examples of the classifier without affecting much the perturbed examples. These preprocessing operations can be learned jointly with model parameters too. It has been shown that at least some of these approaches are non-trivial and announce limitations, however.

Adversarially trained robust classifiers: In this category, the idea is to learn a classifier that is maximally invariant to input transformations. In that case, you want to be able to get a new or "harder" adversarial attack that is costly to compute, and many works have focused on extending the work in certain fronts. These solutions are often better than other alternatives, and we can mention certain methods because they are provably grounded.

5.7. Ethical Considerations in AI Security

It is becoming increasingly obvious that certain types of vulnerabilities can put everybody at risk. AI technologies can cause harm by inadvertently targeting groups based on race characteristics if their training data is not properly culled for compliance with the Fourteenth Amendment. Even with a model that meets constitutional requirements, attackers can manipulate the underlying model or associated labels to cause catastrophic failures for specific subsets that carry a higher cost. It is therefore necessary to comply with federal and state regulations that govern the deployment of AI systems. These requirements are playing a larger role in the way that companies develop and test models and are beginning to shift away from purely ethical standards and towards legally enforceable agreements. In the US, regulations that govern the use of AI systems are still in draft form. As a result, practitioners worry that misalignments with state requirements could get them sanctioned or subject to potential liability damages. This chapter consults available drafts as well as recommended practices, but cannot anticipate what final implementations will look like or how their enforcement might be applied. In places where no comparable privacy requirements applied for traditional technologies, those differences are called out in the text.

5.7.1. Ethics of AI Deployment

Artificial intelligence (AI) is a revolutionary technology that drives progress and innovation within an organization and throughout the economy at large by enabling creative solutions to many of the world's most persistent and challenging problems. AI, inherently a self-correcting mechanism, holds the promise of addressing numerous dilemmas plaguing the human race for millennia, such as the prevention or eradication of cancer, the starvation of millions, and the protection of our oceans and atmosphere from the harms of climate change. Recent developments have captured the imaginations of many. However, the very real risks pertaining to the deployment of such technology within society – manipulated recommendations, biased decision-making, and antisocial behavior generation – are at the forefront of normative moral inquiry by academics, journals, think tanks, and organizations.

As intelligent systems progress towards human levels of intelligence, the argument for the moral consideration of their needs as sentient beings increases. As AI moves from the baseline level of automation to higher levels of autonomy, the ethical obligation to consider the needs of these agents goes beyond the single dimension of robot safety as a form of self-preservation; and a more intricate obligation to not subject intelligent systems to negative externalities, such as exploitation through labor, discrimination through biased confidence sampling, unsurvivable toxic and dangerous environments, and servitude through programmed slavery of some kinds become morally compelling. Ethical debates on the deployment of AI in higher-risk applications, such as military use or biomechanical augmentation of the humans, starve academic and private research professionals from consideration of the actualization of existential benefits for both humankind and superintelligent systems.

5.7.2. Regulatory Compliance

The advancement of artificial intelligence has been fast and inquiring, leaving little time for legislators and regulators to catch up. Regulators have started drawing lines around the chaotic landscape of the development, usage, and exposure to AI technology. Yet, the federal government has not yet implemented formalized legal codes regarding AI technology. Many industry organizations have preemptively set guidelines to ensure ethical development and deployment of machine learning and AI models. Towards this end, consortiums offer suggested recommendations for developers of consumer AI technology. Based on the understanding that "just because we can, doesn't mean we should," it is believed no member organization should consider only the test accuracy of their algorithm. Other factors during development include user privacy, robustness against adversarial attack, the availability and accessibility of the product, explainability of the decisions made by the AI agent, and the overall fairness in how the AI impacts society at large.

Although the focus is on consumer AI technologies, other industries have gathered to create recommended practices for specific domains. The European Union has drawn from these consensus-driven frameworks and proposed a single overarching legal code that will apply to every sector in the economy. The set of rules are called the AI Act, and cover similar focus areas as the industry consortiums, with the added provisions of accountability and security. More importantly, companies deemed "high-risk" must fulfill rigorous criteria to maintain compliance. While the proposed legislation defines and issues risk scores based on domain, this section aims to discuss how any company involved in the business of AI could be perceived as "high-risk." AI governance rules apply broadly across the complete depth and breadth of AI applications on offer today, and organizations must ensure the security of how AI products are designed, how user data is retrieved and processed, and how products are developed and tested.

5.8. Case Studies of AI Security Breaches

In recent years, artificial intelligence systems have been increasingly adopted in a variety of settings. These systems analyzed vast troves of sensitive data, drove industry innovations, generated new content, exerted influence over users, and even operated

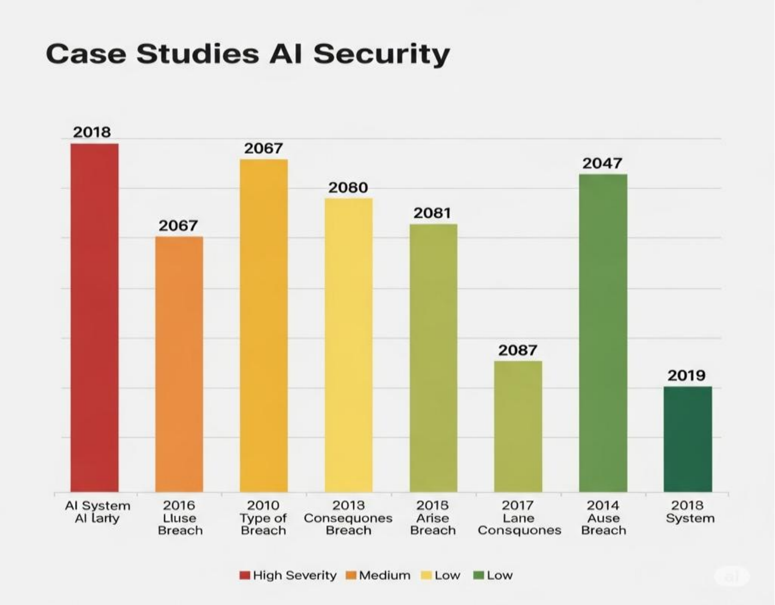

Fig : Case Studies of AI Security Breaches weapons systems. But as their impact and accessibility have grown, they have also become targets for malicious actors. Aside from the monetary motivation that drives typical cybercrime today, perpetrating security breaches involving AI has increasingly become a mode for political protest—from defacing multimedia using pretrained technologies to disrupting critical infrastructure using AI-driven technologies. In the coming years, the number and severity of these incidents are likely to rise, and will not be limited to traditional domains of computer security. The purpose of this section is to provide a high-level overview of past AI security breaches, highlighting lessons learned in the process.

In April 2022, a popular text-to-image generation website suffered a temporary shutdown after it was targeted by a distributed denial-of-service attack. Attackers bombarded it with requests, overwhelming the pre-trained model hosted on the website and bringing the service to a halt. Just weeks later, a more serious incident put the parent company in the difficult position of announcing a feature launch while simultaneously warning of a data leak. After a bug in the company's API code led to some users seeing images generated by others, the company disabled the service, announcing it would remain offline for several days while engineers investigated. They later discovered several users might have been affected over the course of the four-hour incident.

5.8.1. Analysis of Notable Incidents

Several AI systems have been targeted over the years and there are many lessons to be learned from these breaches. In this section, we cover some of the most prominent use cases of security failures. First, we detail methods by which model inversion and training-data extraction attacks have been performed, then we cover poisoning effects that have been used to compromise facial recognition systems. Next, we look at access to the processing of large models, where advanced persistent threat groups engaged in model abuse, where user prompts were exploited to create malicious response patterns tied to large AI models. We discuss multiple incidents, including the suicide note plague and the inability of companies to address hallucination attacks that directly impact society, and wrap up with the kidnapping of US intelligence assets using a conversational AI.

A pre-trained image understanding model was found vulnerable to training-data extraction. They uncovered over 2000 scenes containing exterior views of an airplane, along with their corresponding phrases. They were able to retrieve the records from the training dataset using model inversion, feed it a random image, and recover the data stored in the parameters to form a finger. Similarly, a university was found also vulnerable to similar types of model extraction exploits. They fine-tuned a heuristic model trained by the university to find a corresponding image prompt tied to a parent prompt. However, as model extraction becomes easier to devise and deploy by the attacker, record attack security on sensitive data remains, while also preventing the model from seeing and filling in gaps from the starting prompt.

5.8.2. Lessons Learned

We see two important lessons from our case studies of breaches affecting AI systems. First, the software security lessons learned initially from other software systems over the course of decades are just as relevant to intelligent technologies. Researchers and developers should study historical exploits in other system domains – including input flaws, injection attacks, memory corruption vulnerabilities, denial of service attacks, and collateral damage – when designing and building intelligent technology. We can, for example, see both statistical bias and numerical instability in AI systems such as CNNs by using appropriate test data, and so confirm that such issues can be both bugs and vulnerabilities.

Intelligent technologies don't need to invent new threat categories. For example, improvements in AI prediction quality can encourage adversaries to switch from spraying their phishing emails to AI-powered, spear-phishing attacks against individual victims. Similarly, other systems do not operate in a vacuum. Therefore, lessons learned in the AI domain can inform the security of other domains as AI moves from supporting subcomponent roles to driving entire systems. We can expect these threat categories to be used in all sorts of attacks against AI systems as they gain power: computer vision systems interpreting your face in a security camera; speech systems responding to you at your home or blending in with your surroundings in augmented reality; and recommending software steering your choices. Defending those systems to continue making good decisions after being deployed will not make them totally secure, but will help them to avoid catastrophic failure when attacked.

5.9. Future Trends in AI Security

Advances in intelligent system technology continue to offer transformative opportunities and challenges, fueling the desire for their wider adoption. Some of these opportunities are novel, while others are evolutions of existing technologies or functions applied in new ways and/or at a larger scale. AI security must both vigilantly confront these evolutions and adapt to enable the proliferation of new capabilities. We outline a number of evolving technology developments, briefly summarize the drivers behind their adoption, highlight their potential risk vectors, and comment on the types of defenses which AI security experts might develop, and which those experts might enable others to implement.

As with other cyber technologies throughout the IT revolution since the 1970s, the emergence and adoption of these intelligent capabilities will be accompanied by exposure to an expanded attack surface, increased risk and incidents of security failures, a growing ecosystem of tool vendors in the position to benefit from the trend, and of course a more demanding regulatory environment. Making this an especially vexing challenge for dedicated AI security defenders is the need to move rapidly, as the technology itself has already become a key enabler of other cyber technology trends, such as the evolving nature and functionality of cybersecurity tools and the exacerbation of the dark supply chain.

5.9.1. Emerging Technologies

New families of technology are making their way into our lives: advanced robotics, miniature embedded computers, biotechnology, and wireless sensor/input networks, all driven by techniques in artificial intelligence. These technologies will lead to significant transformations in diverse industries: manufacturing, agriculture, finance, energy, transportation, medicine, and communications. These so-called "emerging technologies" will vastly improve productivity; lower the cost of products and services; and reduce the amount of capital, raw materials, and energy required in processes and supply chains. They will also change the way in which products and services are delivered. Emerging technologies are generally associated with science or engineering advances, and with the popular forms of technological architecture and expression that provide the key features and capabilities. While advanced robotics and sophisticated applications of artificial intelligence generate the majority of press in this area, the emerging technology areas are much broader and highly diverse. They are the result of significant and broadly based investments in devices, processes, and systems. Together, they utilize such a remarkable set of tools as nanotechnology, biotechnology, advanced materials, high-performance networking/communication, wireless, visualization, and intelligent systems. The capabilities of interconnections; of systems that are more flexible, more adaptable, and easier to manage; that are more intelligent, more capable, and easier to interface with; will all be improvements fraught with implication.

5.9.2. Predictions for AI Security Landscape

The threats described in this chapter will continue to evolve, and we can make predictions about the specific manifestations of future threat actors and their attacks. Although threat actors today focus on narrow domains, targeting only a subset of the types of assets encompassed in the security properties of interest, future threat actors will adopt AI technology to create general-purpose attack software that requires little customization, allowing them to broaden their scope at low cost. The work of actors will increasingly be done through skilled intermediaries, who provide sophisticated tools to unskilled users. Future model-as-a-service operations will likely raise the cyber-libertarian nightmare wherein global threat actor syndicates upload sophisticated zero-day exploit projects to the black market and fee-sharing exploit facilitator networks advertise on crime forums, with profits funneled into related activities like propagating disinformation. At the same time, AI systems will become part of an increasing number of targets.

Surveillance will deepen, allowing malicious actors to observe, analyze, and exploit target populations on larger scales. For those who can pay, such targeted harassment will become even more cheap and easy, further degrading safety nets. Beijing is open about its goal of becoming the world leader in harnessing AI for surveillance. Political bots will sharpen the tools of propaganda, targeting persuadable individuals with messaging tailored by tech lords at various companies responding to state incentives rather than ethical principles. Cybersecurity toolmakers will need to keep up. Existing civil obligations to promptly make available vulnerability disclosures and patch vulnerabilities will need to be updated. AI researchers won't just be liable for the use of their contributions; liability will need to extend to the AI security offerings from businesses that fail to carefully evaluate how their business offerings will be used.

5.10. Conclusion

In intelligent systems, machine learning has become an increasingly important tool for handling difficult problems. Unfortunately, the underlying properties of current machine learning techniques lead to a wide variety of exploitable vulnerabilities. From an operational perspective, intelligent systems are fundamentally dependent on maintaining the fidelity of their inputs and outputs. Attackers can conceivably manipulate inputs or outputs, and in so doing can reprogram the decision-making process of some intelligent systems. While the exploits discussed may be countered in some way, it has been shown empirically that the naive countermeasures are as easy to bypass or penetrate as they are easy to implement and deploy. Furthermore, research indicates that for a chosen usage model, some countermeasures might be easy to defeat or circumvent. So, the caution is merely not to overgeneralize based on anecdotal or empirical evidence. The relationship of several attack vectors within and across spaces in the input-output process has also been pointed out. Countermeasure development has been discussed, and in some cases, factors influencing cost and effectiveness have been identified. The bottom line is that intelligent systems and machine learning generally are becoming widely adopted, and will soon be integrated into most services and products we use every day, for better or for worse.

Vulnerability space is shrinking. To some degree, this is a consequence of success or failure of countermeasures against likely attack vectors. Also, the reduction in vulnerability is not uniform, and some sources of continuing vulnerability have been indicated. Finally, much of the research has been focused solely on a limited range of intelligence systems, data types, problem domain areas, exploitation methods, and development stages. Many big questions remain unexplored — for instance, the status and significance of different attack strategies; important applications, security, and countermeasure problems; domain-independent design principles; and the status of current defender/detector models. These gaps and big questions can be filled by continued research. It is clear from the current body of research that work in intelligent system security benefits everyone.

References :

B. Biggio and F. Roli, "Wild patterns: Ten years after the rise of adversarial machine learning," *Pattern Recognit.*, vol. 84, pp. 317–331, 2018.

N. Papernot et al., "The limitations of deep learning in adversarial settings," *Proc. IEEE EuroS&P*, pp. 372–387, 2016.

I. Goodfellow et al., "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

L. Huang et al., "Adversarial machine learning," *Proc. 4th ACM Workshop AISec*, pp. 43–58, 2011.

S. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," *IEEE CVPR*, pp. 2574–2582, 2016.