Space-Domain AI Applications need Rigorous Security Risk Analysis

Alexandra Weber
Telespazio Germany GmbH
alexandra.weber@telespazio.de

Peter Franke
Telespazio Germany GmbH
peter.franke@telespazio.de

***Abstract*—Space missions increasingly rely on Artificial Intelligence (AI) for a variety of tasks, ranging from planning and monitoring of mission operations, to processing and analysis of mission data, to assistant systems like, e.g., a bot that interactively supports astronauts on the International Space Station. In general, the use of AI brings about a multitude of security threats. In the space domain, initial attacks have already been demonstrated, including, e.g., the Firefly attack that manipulates automatic forest-fire detection using sensor spoofing. In this article, we provide an initial analysis of specific security risks that are critical for the use of AI in space and we discuss corresponding security controls and mitigations. We argue that rigorous risk analyses with a focus on AI-specific threats will be needed to ensure the reliability of future AI applications in the space domain.**

I. Introduction

The development of AI technology is advancing rapidly and the practical use of AI has increased significantly in recent years. According to McKinsey & Company, the adoption of AI in organizations has doubled between 2017 and 2022, reaching a level between 50 and 60 percent [21]. Simultaneously, the public attention to AI has increased significantly, fueled by the recent success of AI systems based on large language models like, e.g., OpenAI's ChatGPT [27] and Meta's LLaMa 2 [23].

In the space domain, agencies as well as private companies have started to apply AI technology for various tasks. For instance, the U.S. National Aeronautics and Space Administration (NASA) has developed an AI-based scheduler to be used on board the Mars 2020 Rover [33]. Furthermore, the Mars Express mission of the European Space Agency (ESA) makes use of the tool MEXAR2 that supports mission planning based on constraint programming and flow-network modeling [5], and ESA's FSSCat earth-observation mission uses the ϕ-sat-1 chip that filters out images that are covered by clouds before downlinking the mission data [11]. Further examples of AI in spacecraft are given in [22, Section 5]. Neuraspace is an example of a New-Space company that applies AI in the space domain. More concretely, Neuraspace uses Machine Learning to support space-traffic management [25].

Existing surveys of AI security make it clear that the use of AI brings about numerous threats, including, e.g., poisoning attacks, oracle attacks, data-extraction attacks, and evasion attacks [14], [19], [28]. However, this variety of AI-specific attack techniques is largely not accounted for in popular knowledge bases used for threat analysis in the space domain. More concretely, the MITRE ATT&CK framework and the corresponding instance for the space domain, ESA's SPACE-SHIELD framework, currently do not contain any AI-specific techniques [12], [35]. The SPARTA matrix by the Aerospace Corporation takes into account the poisoning of AI training data and sensor spoofing (Techniques EX-0012.13, DE-0003.12, and EX-0014.03) explicitly, but otherwise abstracts from AI-specific techniques that might be used to achieve attack goals or sub-goals [38]. Moreover, AI technology is still subject to very active development and research, such that new attack vectors on AI applications can be expected to keep arising regularly for the foreseeable future.

Based on (1) the high diversity of threats introduced by AI applications and (2) the dynamic nature of the field of AI technology, we argue that AI-based applications for a highly security-critical domain like space need to be based on a rigorous risk analysis that explicitly accounts for AI-specific security threats. While general AI risk-management guidelines like, e.g., [29], [30], [40], provide guidance with respect to the relevant security aspects of AI usage, the mapping of these aspects to space-domain use cases and assets is currently not supported by any common knowledge base or analysis matrix.

Our goal is to provide a basis for initiating the discussion of AI security in the space community and a starting point for defining a common knowledge base on risks for AI applications in space missions. To this end, we provide a preliminary risk analysis that focuses on AI-specific security threats and is tailored to six categories of space-domain use cases. The categories cover relevant use cases from different phases of space missions, including, e.g., mission planning, satellite-telemetry-data forecasting, and satellite-image classification. Based on the identified risks, we discuss options for mitigations and derive a list of security controls that could be applied to protect space-domain AI applications.

In summary, the contributions of this article are

- a preliminary analysis of the risks posed by AI in six categories of space-domain use cases and
- a collection of security controls tailored to the protection of space-domain AI applications.

The remainder of this article is structured as follows. We provide preliminaries in Section II. In Section III, we describe the threat sources and space-domain use-case categories un-

Workshop on Security of Space and Satellite Systems (SpaceSec) 2024
1 March 2024, San Diego, CA, USA
ISBN 979-8-9894372-1-4
https://dx.doi.org/10.14722/spacesec.2024.23008
www.ndss-symposium.org

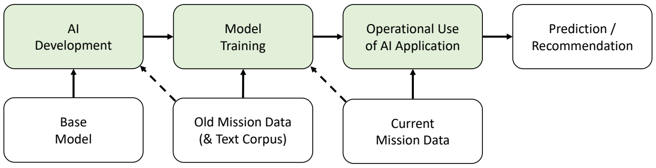
AI Development
Model Training
Operational Use of AI Application
Prediction / Recommendation
Base Model
Old Mission Data (& Text Corpus)
Current Mission Data

Fig. 1. High-level AI-application Workflow

derlying our risk analysis. In Section IV, we present our risk analysis. In Section V, we present the collection of tailored security controls. We discuss related work in Section VI and conclude in Section VII.

II. Preliminaries and Terminology

Machine Learning (ML) is a technique used in many AI systems that operate based on a statistical model. ML is used to build such a statistical model from a base model and historical data, so-called training data, before the system is deployed. To this end, the training data is processed by a learning algorithm, which builds the model. The process of building the model is also referred to as training. During operational use, the AI application applies the statistical model to make predictions or recommendations based on the data it operates on, so-called production data. If the AI system additionally uses the production data to further improve the statistical model during operational use, this is referred to as continuous learning.

In the space domain, the production data is usually data from a current space mission, and the training data is usually older data from the same or a prior space mission. If the AI system also processes natural language, the training data also includes text corpora, i.e., data sets of language resources.

The overall workflow (instantiated for the space-domain case) is visualized in Figure 1. It begins with a development phase, in which a base model is selected and the AI application, including the learning algorithm, is implemented. The first dashed line visualizes that in this phase developers might already need to access mission data in order to test the learning algorithm. The second phase is the model training, which takes the training data (historic mission data and, if applicable, a text corpus), as input and builds the actual statistical model using the learning algorithm. Finally, in the operational phase, the model is used to make predictions or recommendations based on current mission data. The second dashed line visualizes that the current mission data might additionally be used as training data in case continuous learning is applied.

III. Threat Sources & Use-Case Categories

In our risk analysis, we consider eight threat sources. These are inspired by the Methodological Sheet No. 4 of the EBIOS risk-management approach [1], but adapted to match the domain of our analysis as follows. We refine the very generic categories "avenger" and "pathological attacker" to those instances listed in the methodological sheet that are most relevant for our domain, namely compromised insiders and competitors. We do not consider ideological activists and amateurs as separate categories, because their capabilities and resources are similar to those of terrorists and they mainly differ in their motivation. Instead, we additionally consider unintentional threat sources, which are not part of [1], namely human error and environmental factors.

TABLE I. Use-case categories considered in this article

Category ID	Description
PLN	generation of planning suggestions/optimizations based on historic planning data (e.g., for mission plans, ground-station passes)
TEL	analysis and forecasting of satellite telemetry data (e.g., anomaly detection, root cause analysis)
REP	processing and analysis of logs and reports (e.g., simulation and testing results)
AST	assisted generation of suggestions that are based on natural-language prompts (e.g., procedures, control decisions, spacecraft maneuvers)
SEA	search and correlation across diverse data sources (e.g., to support operations, to prepare and anticipate maintenance)
IMA	detection and classification of patterns in satellite images (e.g., cloud cover, forest fire)

All in all, we consider the following threat sources:

- Professional hackers with high technical capabilities and the goal to gain reputation or financial advantage *(corresponds to "specialized outfits" in [1])*,
- state-sponsored actors with access to resources for long-term attacks and the task to gain a strategic advantage *(corresponds to "state-related" in [1])*,
- organized crime, motivated by financial gain, but more limited in terms of resources and technical capabilities *(corresponds to "organised crime" in [1])*,
- terrorists with ideological objectives and sparse resources *(corresponds to "terrorist" in [1])*,
- compromised insiders who seek financial gain by cooperating with external attackers *(refinement of "avenger" in [1])*,
- competitors who seek strategic advantage *(refinement of "pathological attacker" in [1])*,
- human error in the form of honest employees who accidentally cause security risks during high-stress situations *(not covered in [1])*, and
- environmental factors *(not covered in [1])*.

We define six categories (see Table I) of space-domain use cases that capture different ways of using AI during a space-mission life cycle. The first category is PLANNING, abbreviated PLN. It captures applications that use AI to support the generation of plans, e.g., for mission planning or for the planning of ground-station passes, based on existing prior plans, e.g., using machine learning. The second category is TELEMETRY, abbreviated TEL. It captures applications that work on the telemetry data of spacecraft and use AI to support the analysis of such data, e.g., to detect anomalies, identify trends, or forecast future behavior. The third category is REPORTS, abbreviated REP. It captures applications that use AI to automate the processing of log files and more complex inputs, e.g., to provide summary reports, to identify points of interest in test and simulation results, or to suggest further tests. The fourth category is ASSISTANTS abbreviated AST. It captures applications that use generative AI to create suggestions, e.g., for procedures, spacecraft maneuvers, or control decisions in general, based on natural-language prompts. The fifth category is SEARCH, abbreviated SEA. It captures applications that leverage AI to perform searches and identify correlations across a large number of diverse inputs, e.g., to support spacecraft operators in analyzing anomalies or to support spacecraft analysts in anticipating suitable maintenance

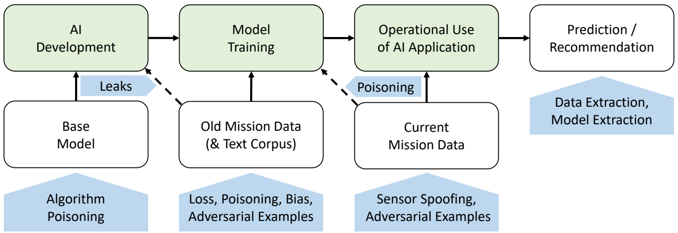

Fig. 2. Overview of Risks in AI-application Workflow

windows. The last category is IMAGES, abbreviated IMA. It captures applications that apply AI in the processing of satellite images, e.g., to support the detection or removal of cloud cover from the images or to support the detection of forest fires.

These categories cover multiple common types of data processed in the space domain (planning files, telemetry data, logs/reports, satellite images). Furthermore, they cover different phases of the sense-plan-act cycle during mission operations (monitoring of telemetry data, mission planning and flight dynamics, issuing of spacecraft commands), and different ways in which AI is applied (recommendations/predictions based on prior data, artefact generation based on natural-language prompts, automatic preprocessing of data). For instance, the MEXAR2 application would be an example from category PLN, and ϕ-sat-1 would be an example from category IMA.

IV. Risk Analysis for AI in Space Use Cases

Our risk analysis is structured according to the phases in the AI-application workflow: the model-training phase, the operational AI-applications phase, and the development phase. For each of the phases, we derived relevant risks based on the threat sources and use-case categories from Section III in combination with known security threats induced by AI in general according to existing overviews [14], [29].

Figure 2 provides an overview of the risks that are relevant in each of the phases and Table II summarizes the risks and their mapping to threat sources and categories of space-domain applications. In the following, we elaborate on the risk analysis for each phase and on the reasoning behind the mapping.

A. Risk Analysis for the Model-Training Phase

The training of AI models on training data gives rise to AI-specific security risks in terms of integrity and availability.

1) Integrity: The integrity of training data is endangered by three major factors: Firstly, data obtained from internal or external sources might be **biased**, i.e., not reflect the real-world circumstances accurately, and might correspondingly mislead the AI system to learn an imperfect model (**Risk RI01**). Such bias might occur even if the data provider has no malicious intent, e.g., because the data is obtained only from a set of sources with shared characteristics or because the data provider has an incentive to boost performance indicators that are explicitly or implicitly contained in the data.

Secondly, a malicious source might provide data that is **poisoned**, i.e., crafted to contain specific triggers, to introduce a backdoor into the learned statistical model (**Risk RI02**). For instance, the very recent Jigsaw Puzzle attack [44] manipulates

TABLE II. Risks with Mapping to Threat Sources & Use Cases

Risk ID	Description	Threat Sources	Use Cases
	Integrity Risks in the Model-Training Phase		
RI01	Erroneous model due to biased mission data.	Human error, compromised insiders, environmental factors.	PLN, TEL, REP, AST, SEA, IMA
RI02	Erroneous model due to poisoned mission data.	Compromised insiders.	PLN, TEL, REP, AST, SEA, IMA
RI03	Erroneous outputs due to poisoned/adversarial text corpora.	Professional hackers, state-sponsored actors.	PLN, REP, AST, SEA
	Availability Risks in the Model-Training Phase		
RA01	Training delayed because external provider of text corpora fails to deliver.	Environmental factors.	PLN, REP, AST, SEA
RA02	Training delayed because mission data is stored in wrong format.	Human error, compromised insiders.	PLN, TEL, REP, AST, SEA, IMA
RA03	Training delayed because mission data has been deleted.	Human error, compromised insiders, environmental factors.	PLN, TEL, REP, AST, SEA, IMA
	Confidentiality Risks in the Operational Phase		
RC01	Breach of historic mission data through data extraction.	Compromised insiders, professional hackers, state-sponsored actors, organized crime, competitors.	PLN, TEL, REP, AST, SEA, IMA
RC02	Breach of intellectual capital through model extraction.	Compromised insiders, professional hackers, state-sponsored actors, organized crime, competitors.	PLN, TEL, REP, AST, SEA, IMA
	Integrity Risks in the Operational Phase		
RI04	Missed critical spacecraft incidents due to sensor spoofing.	Professional hackers, state-sponsored actors, organized crime, terrorists, competitors.	TEL
RI05	Spacecraft controls triggered unnecessarily due to sensor spoofing.	Professional hackers, state-sponsored actors, organized crime, terrorists, competitors.	TEL
RI06	Erroneous output for plans, maneuvers, simulations, etc. due to adversarial examples.	Compromised insiders.	PLN, TEL, REP, AST, SEA, IMA
RI07	Misclassification of satellite or surveillance-camera images due to adversarial examples.	Professional hackers, state-sponsored actors, compromised insiders.	IMA
RI08	Erroneous outputs due to prompt injection into generative AI.	Professional hackers, organized crime, terrorists, competitors.	AST
	Availability Risks in the Operational Phase		
RA04	Destruction of learned model due to poisoned production data.	Human error, compromised insiders.	PLN, TEL, REP, AST, SEA, IMA
	Confidentiality Risks in the Development Phase		
RC03	Data leaks during the development phase.	Compromised insiders, human error.	PLN, TEL, REP, AST, SEA, IMA
	Integrity Risks in the Development Phase		
RI09	Erroneous model due to algorithm poisoning.	Professional hackers, state-sponsored actors, competitors, compromised insiders, human error.	PLN, TEL, REP, AST, SEA, IMA

the training data for a malware classifier, such that malware files containing a specific trigger string are classified as benign.

Thirdly, data providers might provide **adversarial examples** that contain wrong information to intentionally mislead AI systems (**Risk RI03**). This is particularly relevant for text corpora used to train AI for Natural-Language Processing (NLP), because such corpora usually originate from external sources and because NLP systems tend to give high importance to clauses that contain trigger words like, e.g., "why", "how", or "because" [42], which gives attackers a high leverage when introducing adversarial information into texts.

With respect to space-domain use cases, the risks RI01 and RI02 are relevant for applications from all six categories, because all six categories involve historic mission data that is used for training AI models: historic mission plans for `PLN`-type applications, historic telemetry data for `TEL`-type applications, historic logs and reports for `REP`-type applications, historic satellite images for `IMA`-type applications, historic procedures, control decisions, etc. for `AST`-type applications, and diverse historic mission data for `SEA`-type applications. Risk RI03 affects those applications that use NLP. Applications from category `AST`, which process natural-language prompts and are trained on text corpora are most affected by this. In addition, applications from the categories `PLN`, `REP`, and `SEA` might be affected if the planning data, test reports, or overall mission data contain natural-language components.

The most relevant threat sources for bias in internal mission data (RI01) are accidental short-cuts by honest employees in the data collection, intentional manipulation by compromised insiders, or technical failure, e.g., of the sensors in a spacecraft that collect mission data. An external actor who obtains access to and then manipulates the mission data is conceivable in principle, but compromising an individual with internal access is likely the line of least resistance. Similarly, for RI02, a compromised insider is the most likely threat source. For RI03 on external text corpora, the most relevant threat sources are those who have the resources and incentive to manipulate such large data bases in a targeted way and to hide the traces of their operations. Thus, we consider professional hackers and state-sponsored actors the most likely adversaries here.

The consequences of training data with compromised integrity are potentially wrong recommendations for procedures, incident responses, or maneuvers in general (`AST`), for test reports (`REP`), or for planning decisions (`PLN`). Furthermore, in category `SEA`, compromised training data might lead to wrong predictions for expected incidents or down-times.

2) Availability: The availability of data during the training phase is at risk if an external data provider **fails to deliver** (**Risk RA01**), if any relevant training data is **not AI-readable** because it is not stored in a suitable format (**Risk RA02**), or if any relevant training data gets **deleted** (**Risk RA03**).

As discussed already in the case of integrity, the external data that is relevant for the space-domain use cases are text corpora and these are relevant for the categories `PLN`, `REP`, `AST`, and `SEA`. The most likely source for a failure to deliver text corpora (RA01) are environmental factors, e.g., that the provider of the text corpora goes out of business.

If historic mission data is stored in a format that is not AI-readable (RA02), this is most likely either due to a honest employee who lacks awareness of the requirements or who works with outdated processes and data-processing solutions or due to a compromised insider who purposefully boycotts the AI-application. The deletion of training data (RA03) is most likely to be caused by human error, a compromised insider, or environmental factors that lead to system failure and data loss. As in the case of integrity, external attackers are most likely to take the path of least resistance by exploiting mistakes made by a honest internal employee or by recruiting an internal employee who supports them purposefully.

The consequences of unavailability or delayed availability of training data range from additional cost to switch to another data source (e.g., for text corpora) to significant delays in the deployment of the AI system (e.g., waiting for another space mission to deliver the mission data for training).

B. Risk Analysis for the Operational AI-Application Phase

During the operational phase, the confidentiality, integrity, and availability of the data and the model are endangered.

1) Confidentiality: Information about the training data might be extracted from a model through so-called **oracle attacks** (also called data-extraction attacks or model-inversion attacks). Such attacks use the AI system as an oracle: they query the system on carefully crafted production data and then infer information about the training data from the system's predictions or recommendations (**Risk RC01**). One variant of oracle attacks is the membership-inference attack that queries the AI system with the goal to find out whether a particular data point was part of the training data set. Membership-inference attacks are especially dangerous for AI systems that provide confidence scores together with the classification of the production data. When attacking such a system, an attacker could query it on permutations of an input until a perfect confidence score is achieved. A perfect score indicates that the input likely has a one-to-one correspondence to an element of the training set. Even if an attacker cannot recover the training data completely, he might be able to infer the missing information from partial knowledge of the training data. For instance, just three properties (the place, gender, and date of birth) suffice to de-anonymize 50% of the U.S. population [36].

So-called **model-extraction attacks** aim at reconstructing the statistical model itself by queries to the public API of the system (**Risk RC02**). Consider, e.g., a statistical model that uses logistic regression, i.e., a model that predicts probabilities of classes using a set of logistic functions on the input space. If the model outputs the probability together with the prediction, the n parameters of the underlying logistic functions can be extracted by querying the model on $n+1$ inputs and solving the resulting equation system [39]. Other variants of model-extraction attacks can also be used against regression-based models that do not output probability values and against decision-tree-based classifiers [39].

In the space-domain use-case types from our six categories, both the training data and the statistical models contain sensitive information. More concretely, the models constitute intellectual capital that provides a strategic advantage against competitors. Moreover, the historic mission data used for training purposes (in particular data about past incidents, but also other prior mission data) might contain information on weaknesses related to space missions that should not fall into the hands of attackers.

The most likely source of model-extraction attacks or attacks to infer training data are insiders who have direct access to the APIs of the AI systems but no access to the mission data. Moreover, external attackers with significant resources, i.e., professional hackers and state-sponsored actors, might exploit vulnerabilities in the systems or networks on which the AI systems are deployed in order to obtain access to the APIs of the AI systems. If AI-system APIs are exposed publicly, organized crime and competitors have to be considered as additional threat sources.

2) Integrity: The integrity of the model (in terms of its ability to produce correct predictions or recommendations based on production data) might be compromised during the operational phase due to manipulated production data. **Sensor-spoofing attacks** target production data that is obtained with sensors and manipulate the data-collection process. They can be used to hide existing anomalies (**Risk RI04**) or introduce anomalies that do not exist (**Risk RI05**). For instance, injecting ultrasound noise that interferes with gyroscopic sensors can cause drones to misbehave and even crash [34] and GPS signals can be spoofed using radio-frequency transmitters in order to prepare a takeover of unmanned aerial vehicles [32].

Furthermore, attackers might mislead AI systems and trigger misclassifications by introducing small perturbations into production data in a way that the perturbations have a high leverage for the classification of the data. This type of attack is called **adversarial-example attack** or evasion attack and works independently of whether the production-data collection is performed using sensors (**Risk RI06**).

A specialized variant of adversarial-example attacks, which is called **camouflage attacks**, targets computer-vision applications that are based on neural networks. Typically, AI systems that rely on neural networks operate on small input sizes. This means that high-resolution images need to be scaled down before inputting them to the AI system as training or production data. This pre-processing is performed using dedicated image-scaling algorithms. In a camouflage attack, the attacker manipulates an image, such that the way it looks to the human eye before scaling and the way it appears to the AI system after scaling differ significantly (**Risk RI07**). For instance, an image that appears to show sheep before scaling might appear to show a wolf after scaling as in the example from [20]. Object-detection systems that apply AI to determine the bounding box and predict the class of a candidate object in an image are also subject to variants of adversarial-example attacks. Recently, researchers found a pattern that, when printed on clothes, let the clothes function as invisibility cloaks, hiding their wearer from detection systems [43].

Remark. Note that adversarial examples might also endanger the availability of a spacecraft if they trigger an on-board AI model to consume an excessive amount of the spacecraft's resources or transmission bandwidth.

Generative AI systems that are based on natural-language prompts might be attacked with adversarial examples using so-called **prompt injection** (**Risk RI08**). That is, attackers might supply adversarial examples in the form of malicious instructions in the input text or malicious content hidden in texts that are used as source data [29].

Sensor-spoofing attacks (RI04, RI05) are relevant to all use cases that operate on telemetry data, i.e., applications from category `TEL`. How sensor-spoofing attacks can be mounted on telemetry data has been demonstrated recently by Salkield, Köhler, Birnbach, Baker, Strohmeier, and Martinovic [31]. They used affordable off-the-shelf hardware to send a manipulated variant of the light-data collected by the MODIS sensors of the Terra and Aqua satellites to a ground station. Such manipulation causes the NASA Fire Information for Resource Management System (FIRMS) to produce misclassifications, such that actual fires might be overlooked and fires might be reported that do not exist in reality. If applied to applications from `TEL`, sensor-spoofing attacks might lead to both, critical incidents of spacecraft being overlooked and fabricated incidents triggering counterproductive spacecraft manoeuvres. Since the equipment needed for spoofing telemetry data is easily obtainable and affordable, such attacks might originate not only from professional hackers and state-sponsored actors, but also from attackers with fewer resources (terrorists, competitors, or organized crime who aim to extort space agencies).

Adversarial examples (RI06) are most likely to originate from compromised insiders who can manipulate the supply of production data to the AI systems. The corresponding risk applies to applications from all six of our space-domain application categories. The more specialized camouflage attacks (RI07) affect computer-vision applications from the category `IMA`. They could originate from compromised insiders who have access to the production data from the satellites and are trying to stealthily manipulate the outputs of, e.g., navigation-related applications, applications that track the effects of climate change, or applications that provide information to intelligence services. Moreover, the invisibility cloaks from [43] could cause object-detection systems that are used to monitor security-critical areas to miss people who are wearing adversarial patches on their clothes. This way, attackers with access to significant resources (professional hackers or state-sponsored actors) might be able to enter security-critical areas without being detected.

Prompt-injection attacks (RI08) affect generative AI systems from category `AST`, and in particular systems that are exposed publicly, e.g., in the form of a chat bot that provides information on space missions on a space agency's website for public-relations purposes. The most likely threat source in this case are professional hackers, organized crime, terrorists, or competitors, who have an incentive to provoke inappropriate statements from the chat bot and discredit the space agency or add to their own reputation.

3) Availability: The availability of a statistical model might be compromised by **production-data poisoning** if the AI system uses continuous learning to improve the model. More concretely, the model might become unusable after it is queried on large amounts of poisoned production data (**Risk RA04**). The poisoning of production data follows the same approach as the poisoning of training data described above and has the same effect, because continuous-learning systems use the production data also as training data.

Continuous learning can be implemented in use cases from any of the six categories, such that RA04 is applicable to all of the categories. The most likely threat sources for the poisoning of production data are the same as for the poisoning of training data, namely mistakes by honest employees and intentional manipulation by compromised insiders.

In principle, statistical models might also become temporarily unavailable due to Denial-of-Service (DoS) attacks on the API of the AI system if it is exposed to public queries. However, since the AI systems captured by the six space-domain categories are usually not meant to be exposed publicly, the likelihood of DOS attacks and corresponding system outages is low and we will not consider them as a separate AI-specific risk in our analysis.

C. Risk Analysis for the Development Phase

So far, we have focused on the training phase and operational phase of AI systems. But even in the development phase, AI applications are subject to AI-specific threats.

1) Confidentiality: AI systems need access to (a variant of) production data already during the development phase to test the training algorithm. This increases the risk of **data leaks** during this phase significantly (**Risk RC03**). This risk affects all applications that use mission data, i.e., applications from any of the six categories. The most likely threat source is human error or a compromised insider who has access to the data in the context of the application development.

2) Integrity: In addition to training-data poisoning and production-data poisoning, **algorithm poisoning** (also called model poisoning), which occurs during the development phase, gives rise to a significant risk (**Risk RI09**). In an algorithm-poisoning attack, an attacker manipulates the parameters of the base model based on which the learning algorithm is invoked or manipulates the parameters of the model after the training.

The fact that supply chains are more complex for AI applications than for traditional non-AI applications makes it harder to detect algorithm poisoning up front with audits. Moreover, AI code is typically less maintainable than traditional code [29], which makes it harder to spot vulnerabilities later.

The risk that arises from algorithm poisoning affects any AI application. A base model that is acquired from an external source might have been poisoned by a variety of threat actors, ranging from professional hackers, to state-sponsored actors, to competitors. Even if developed internally, the base model might be poisoned - either intentionally by compromised insiders or accidentally by honest employees who reuse code snippets that are publicly available on the Internet.

V. Security Controls for AI in Space Use Cases

In this section, we discuss security controls to mitigate the risks identified in the risk analysis from Section IV. Table III provides an overview of the proposed controls and the mapping to the risks that they mitigate. The discussion in the following is structured by the phase in the AI-application workflow to which the respective controls are applicable.

TABLE III. Security Controls for the Identified Risks

Control ID	Mitigated Risk	Description	Use Cases
		Security Controls for the Training Phase	
CT01	RI01, RI02	Restrict access to the historic mission data that is used for training.	PLN, TEL, REP, AST, SEA, IMA
CT02	RI01, RI02	Store & isolate hashes of the mission data and cross-check before starting the training.	PLN, TEL, REP, AST, SEA, IMA
CT03	RA02	Provide clear guidelines and training on the structure and format for storing data.	PLN, TEL, REP, AST, SEA, IMA
CT04	RA02	Perform monthly audits of the storage of new mission data.	PLN, TEL, REP, AST, SEA, IMA
CT05	RA03	Keep backups of mission data in at least two different physical locations with restricted access.	PLN, TEL, REP, AST, SEA, IMA
CT06	RI03, RA01	Use only text corpora from trusted providers.	PLN, REP, AST, SEA
CT07	RI03, RA01	Use text corpora from at least two independent providers.	PLN, REP, AST, SEA
		Security Controls for the Operational Phase	
CO01	RC01, RC02	Ensure that AI systems output classifications/recommendations without confidence scores.	PLN, TEL, REP, AST, SEA, IMA
CO02	RC01, RC02	Restrict access to the AI systems using context-sensitive access control that only allows accesses on need-to-know basis from the local network or via secure VPN connections.	PLN, TEL, REP, AST, SEA, IMA
CO03	RC01, RC02	Limit the rate at which the AI systems can be queried by users in roles with lower privileges.	PLN, TEL, REP, AST, SEA, IMA
CO04	RA04	Do not deploy continuous learning without supervision in operational systems. Instead, collect the production data and corresponding outputs and train the model on them in a controlled way, including poisoning countermeasures.	PLN, TEL, REP, AST, SEA, IMA
CO05	RA04	Backup intermediate versions of the trained statistical model to enable rollbacks.	PLN, TEL, REP, AST, SEA, IMA
CO06	RI04, RI05	Validate all production data for plausibility automatically and let a human cross-check in case of outliers and anomalies. Cross-check telemetry data across multiple ground stations.	TEL
CO07	RI04, RI05	Protect area near ground stations from unauthorized access.	TEL
CO08	RI04, RI05	Use radio-frequency monitoring to detect malicious signals near ground stations.	TEL
CO09	RI06, RI07	Apply adversarial training for all classifiers.	PLN, TEL, REP, AST, SEA, IMA
CO10	RI06, RI07	Apply input and output validation.	PLN, TEL, REP, AST, SEA, IMA
CO11	RI08	Use human oversight to check the generated suggestions.	AST
		Security Controls for the Development Phase	
CD01	RC03	Use less critical data (e.g., from a different mission) for development and testing.	PLN, TEL, REP, AST, SEA, IMA
CD02	RC03	Apply DLP techniques to mission data that is used for testing.	PLN, TEL, REP, AST, SEA, IMA
CD03	RI09	Review any code that is reused from elsewhere (especially base models) wrt. security aspects.	PLN, TEL, REP, AST, SEA, IMA
CD04	RI09	Keep track of all software components and their origin, e.g., using a Software Bill of Materials.	PLN, TEL, REP, AST, SEA, IMA

A. Controls and Mitigations for the Training Phase

To mitigate the risks in the training phase, we suggest to apply mitigation techniques that protect the historic mission data that is used as training data from tampering. Firstly, the access to the training data should be restricted based on the need-to-know principle to reduce the attack surface (**Control CT01**). Secondly, hashes of the training data should be stored in a place that cannot be accessed by the same individuals/roles as the data itself, and a cross-check of the data against the hashes should be performed before starting the training process (**Control CT02**). This will allow one to detect compromised integrity, such that the AI system does not get trained on compromised data. Thirdly, all employees who are in charge of storing mission data or of implementing software that stores mission data should receive clear guidelines and training on the required structure and format for data storage (**Control CT03**). Fourthly, audits of the storage of incoming mission data should be performed on a regular basis to reduce the amount of data that might become unavailable for usage in AI applications due to improper storage (**Control CT04**).

In addition, the training data should be protected from loss due to environmental factors like fires, power outages, floods, earthquakes, or similar. Therefore, we suggest to keep regular incremental backups of the historic mission data in at least two different physical locations with restricted access (**Control CT05**). If the training data includes text corpora from external providers, the corpora should be obtained from trusted providers only (**Control CT06**). Moreover, at least two independent providers should be used to enable cross-validation and to provide redundancy in case one of the providers fails to deliver (**Control CT07**).

B. Controls and Mitigations for the Operational Phase

To protect against data breaches or model breaches caused by data- or model-extraction attacks, the AI systems should output only the classifications/recommendations based on the production data inputs, and should not include any confidence scores (**Control CO01**). This increases the effort required for successful attacks significantly. For instance, the model-extraction attacks presented by Tramèr, Zhang, Jules, Reiter, and Ristenpart required up to 100 times more queries when mounted in the absence of confidence scores [39]. Hence, the omission of confidence scores reduces the likelihood that the risks RC01 and RC02 manifest. In addition to omitting confidence scores, the access to the AI systems should be restricted using context-sensitive access control that only allows access on need-to-know basis from the local network or via secure VPN connections (**Control CO02**). Each individual should only have access to those AI systems required for his work. In addition, for some AI systems (e.g., those that generate artefacts like plans) it makes sense to limit the rate at which the AI system can be queried by each user, such that it matches the legitimate needs of the respective user (**Control CO03**).

For AI systems that make use of continuous learning, poisoned production data threatens the integrity of the system's statistical model. To reduce this risk, we propose to not deploy continuous learning without supervision in operational systems. Instead, the production data and corresponding classifications should be collected and run through the learning algorithm in a controlled way that includes the application of poisoning countermeasures (**Control CO04**), e.g., the TRIM algorithm in case of regression tasks [18]. Furthermore, we suggest to keep systematic backups of intermediate versions of the trained model to enable rollbacks (**Control CO05**).

To mitigate the risks related to sensor spoofing, i.e., to reduce the likelihood of successful sensor-spoofing attacks, production data provided as inputs to the AI systems should be validated for plausibility (**Control CO06**). For instance, inputs that consist of telemetry data should be cross-checked across more than one ground station and checked for outliers or anomalies. If an anomaly is detected, it should be cross-checked by a human whether this is a genuine anomaly in the data or whether it might be caused by sensor spoofing. In addition, the area surrounding ground stations should be protected from unauthorized access (**Control CO07**) and monitored for malicious or interfering signals (**Control CO08**) in order to account for attacks like the Firefly attack [31].

We suggest to mitigate the threat of adversarial-example attacks, including the threat of camouflage attacks on computer-vision applications, by a combination of two countermeasures. The first one is adversarial training (**Control CO09**), i.e., adding adversarial examples generated by attack algorithms to the training set together with their correct classifications. The second one is input and output validation **(Control CO10**), i.e., checking whether the features of the production data received as input are within the expected range based on the training data and checking whether the result generated based on the data is plausible in the given context. Similar plausibility checks are already in place for non-AI space-domain applications, e.g., to prevent the erroneous classification of bright areas in satellite images of deserts as snow.

For the prompt-injection variant of adversarial-example attacks that affects `AST` applications, adversarial training or input validation will likely be infeasible because the input is natural language. In this case, we propose to rely on human oversight to check the outputs that the AI system generates based on the natural-language inputs (**Control CO11**). In the hypothetical case of a chat bot on a space agency's website, e.g., a human could regularly review (in an anonymized way) the answers that the bot has provided and trigger adjustments of the AI model if any undesirable content occurs in them.

C. Controls and Mitigations for the Development Phase

The two risks that we identified for the development phase are the leakage of production data and algorithm poisoning. To reduce the former risk, the development and testing should, as much as possible, rely on data that is similar in nature to the production data (e.g., time series of telemetry data) but of lower security criticality, e.g., from a different mission (**Control CD01**). Where the use of security-critical data cannot be avoided, it should be protected by access control and other Data-Loss-Prevention (DLP) techniques like, e.g., encryption at rest and in transit or usage control (**Control CD02).** To counter the threat of algorithm poisoning, we propose to review any code that is reused from external sources or other internal projects with respect to security aspects. In particular, the parameters of any base models that are used in the beginning of the training phase should be checked for their plausibility

(**Control CD03**). In addition, the origin of each software component should be tracked, e.g., in a Software Bill of Materials, in order to be able to quickly identify and react to potential compromises of the supply chain (**Control CD04**).

VI. Related Work

In the following, we provide an overview of existing analyses and surveys with respect to AI security and of the ongoing standardization efforts for AI security.

A. Surveys and Analyses of AI Security

Security threats that are introduced by AI are an actively researched area. There are multiple survey articles that provide an overview of the insights gained so far [14], [19], [28]. In addition, the U.S. Department of Energy (DoE) provides an AI Risk Management Playbook in the form of an online catalogue of risks and mitigations specific to AI technology [40] and Microsoft Security has published best practises for AI Security that include a series of security controls that are organized along the life cycle of AI applications [30]. The Open Worldwide Application Security Project (OWASP) provides a security and privacy guide for AI that contains a collection of security threats introduced by AI [29].

On a more abstract level than the above-mentioned resources, multiple guides for managing risk in AI and building secure AI applications are available, including the Artificial Intelligence Risk Management Framework (AIRMF) and accompanying playbook by the U.S. National Institute of Standards and Technology (NIST) [26] and the Secure AI Framework (SIAF) Approach by Google [13]. These could be a basis for refining our preliminary risk analysis into concrete risk management for individual AI applications.

Finally, some tool support for hardening the security of AI applications is already available, ranging from the Adversarial Robustness Toolbox (ART) Python library for implementing attacks to test the security of ML systems [37], to the TextAttack Python framework for adversarial attacks on NLP [24], to the Dioptra testbed that supports the testing of ML applications (focus: image classification) against a range of attacks [3].

All of the above-mentioned resources consider AI security in general and are not tailored to the space domain. We are aware of three existing works that consider AI security in the space domain. Firstly, the SPARTA matrix by The Aerospace Corporation, which provides a knowledge base on adversarial actions against spacecraft, covers poisoning of AI training data and sensor spoofing [38]. To date, it does not cover any other AI-specific threats explicitly. Secondly, the position paper by Cyr, Long, Sugawara, and Fu focuses on the attack surface introduced by sensor spoofing [6]. It is complementary to our risk analysis, because it does not take into account any threats outside sensor spoofing, but provides a detailed analysis of the threats related to sensor-spoofing attacks on the different subsystems of spacecraft. Thirdly, Breda, Markova, Abdin, Jha, Carlo, and Mantı identify three exemplary vulnerabilities introduced by AI in space systems: manipulated training data is used and leads to wrong output of on-board remote sensing systems, anomalies in the system health check are overlooked or non-existing anomalies are reported, and mechanical arms are operated incorrectly and damage space assets [4]. With respect to countermeasures, they give the high-level recommendation to pair traditional cybersecurity countermeasures (like, e.g., encryption and zero trust) with new measures that should be developed to determine and handle the degree of uncertainty in AI outputs.

B. Standardization of AI Security

The standardization of AI in general and of cybersecurity for AI specifically are currently in progress. Detailed overviews of the related standardization efforts are available, e.g., from the European Union Agency for Cybersecurity (ENISA) [2] and in the German Standardization Roadmap on Artificial Intelligence [41]. We briefly highlight the most related international and European standardization efforts below.

Characteristics for the quality of training data, including possible measures to quantify the data quality, are addressed in the ISO/IEC 5259 series of international standards. For our analysis, the most relevant characteristics are diversity (absence of bias) and identifiability (vulnerability to oracle attacks). The standard ISO/IEC 25059 provides a quality model for AI applications as a whole, which includes security in terms of confidentiality, integrity, non-repudiation, accountability, authenticity, and an AI-specific dimension called intervenability, which is the “degree to which an operator can intervene in an AI system’s functioning in a timely manner to prevent harm or hazard” [17]. The ISO/IEC Technical Report 27563 collects best practises with respect to AI security [16]. It reports on a study across different AI use cases and highlights attention points that include poisoning, adversarial attacks, and model stealing. The ISO/IEC Standard 23894 provides guidance on risk management in AI, emphasizing in particular that AI is a fast-evolving field and AI systems themselves evolve through continuous learning, such that risk analysis for AI needs to be dynamic with regular reviews and improvements [15]. Further ISO/IEC international standards on AI are expected to be published in the near future, including, e.g., ISO/IEC 27090 on guidance for addressing security threats in AI.

On the European level, the European Telecommunications Standards Institute (ETSI) has published multiple group reports on AI security, including a threat ontology [10], a report on countering attacks on training-data integrity [8], and a report on countermeasures to attacks on AI in general [9]. Moreover, the European Cooperation for Space Standardization (ECSS) has developed a draft standard in the form of the ML Qualification Handbook [7]. The handbook suggests to test machine-learning models according to coverage criteria (e.g., branch coverage in decision trees or coverage of neuron activation in Neural Networks). To evaluate a system’s vulnerability to attacks (e.g., poisoning, adversarial examples), the handbook recommends adversarial testing. In addition, formal verification can be applied to complement the testing efforts.

VII. Conclusion

In this article, we described our preliminary risk analysis that focuses on the security threats that AI introduces to space-domain applications. We then presented a corresponding collection of security controls for the mitigation of the risks identified in the analysis. Our motivation was that due to the complexity of AI technology and the high security criticality of space-domain applications, it is crucial to base the development of future space-domain AI applications on a systematic risk analysis that is tailored to the specific application context. We hope that our preliminary results can serve as a basis for a comprehensive risk analysis for AI in space in general and for the refinement of such an analysis to concrete applications from the different use-case categories.

Acknowledgments

The authors would like to thank César Guzmán-Alvarez, Ivan Janes, Marc Niezette, and Luigi Palladino for their helpful input and feedback during the analysis, the anonymous reviewers for their constructive comments, and the participants of SpaceSec'24 for the lively discussion and suggestions.

References

[1] Agence nationale de la sécurité des systèmes d'information (ASSNI), "EBIOS Risk Manager. Going Further," ANSSI-PA-058-EN, Version 1.0. [Online]. Available: https://cyber.gouv.fr/sites/default/files/2019/11/anssi-guide-ebios_risk_manager-en-v1.0.pdf

[2] P. Bezombes, S. Brunessaux, and S. Cadzow, "Cybersecurity of AI and Standardisation," European Union Agency for Cybersecurity (ENISA), Tech. Rep., 2023. [Online]. Available: https://www.enisa.europa.eu/publications/cybersecurity-of-ai-and-standardisation/

[3] H. Booth, J. Glasbrenner, H. Huang, C. Miniter, and J. Sexton, "Securing AI Testbed (Dioptra) Documentation," National Institute of Standards and Technology (NIST), Tech. Rep., 2021. [Online]. Available: https://www.nist.gov/publications/securing-ai-testbed-dioptra-documentation

[4] P. Breda, R. Markova, A. F. Abdin, D. Jha, A. Carlo, and N. P. Mantı, "Cyber Vulnerabilities and Risks of AI Technologies in Space Applications," in *Proceedings of the 73rd International Astronautical Congress (IAC)*. IAF, 2022, pp. 1–10. [Online]. Available: https://iafastro.directory/iac/archive/browse/IAC-22/D5/4/70380/

[5] A. Cesta, G. Cortellessa, M. Denis, A. Donati, S. Fratini, A. Oddi, N. Policella, E. Rabenau, and J. Schulster, "Mexar2: AI Solves Mission Planner Problems," *IEEE Intelligent Systems*, vol. 22, no. 4, pp. 12–19, 2007. [Online]. Available: https://doi.org/10.1109/MIS.2007.75

[6] B. Cyr, Y. Long, T. Sugawara, and K. Fu, "Position Paper: Space System Threat Models Must Account for Satellite Sensor Spoofing," in *Proceedings of the 1st Workshop on the Security of Space and Satellite Systems (SpaceSec)*. Internet Society, 2023, pp. 1–6. [Online]. Available: https://www.ndss-symposium.org/wp-content/uploads/2023/06/spacesec2023-231491-paper.pdf

[7] ESA Requirements and Standards Section, "Machine Learning Qualification Handbook (ECSS-E-HB-40-02A DIR1)," European Cooperation for Space Standardization (ECSS), Tech. Rep., 2023. [Online]. Available: https://ecss.nl/get_attachment.php?file=2023/06/ECSS-E-HB-40-02A-DIR1-(16May2023).docx

[8] ETSI TECHNICAL COMMITTEE (TC) Securing Artificial Intelligence (SAI), "Data Supply Chain Security," European Telecommunications Standards Institute (ETSI), Tech. Rep. ETSI GR SAI 002 V1.1.1, 2021. [Online]. Available: https://www.etsi.org/deliver/etsi_gr/SAI/001_099/002/01.01.01_60/gr_SAI002v010101p.pdf

[9] ——, "Mitigation Strategy Report," European Telecommunications Standards Institute (ETSI), Tech. Rep. ETSI GR SAI 005 V1.1.1, 2021. [Online]. Available: https://www.etsi.org/deliver/etsi_gr/SAI/001_099/005/01.01.01_60/gr_SAI005v010101p.pdf

[10] ——, "AI Threat Ontology," European Telecommunications Standards Institute (ETSI), Tech. Rep. ETSI GR SAI 001 V1.1.1, 2022. [Online]. Available: https://www.etsi.org/deliver/etsi_gr/SAI/001_099/001/01.01.01_60/gr_SAI001v010101p.pdf

[11] European Space Agency, "FSSCat/ϕ-sat-1," (accessed Jan 3, 2024). [Online]. Available: https://esamultimedia.esa.int/docs/EarthObservation/PhiSAT_factsheet2_200605.pdf

[12] ——, "Space Attacks and Countermeasures Engineering Shield (SPACE-SHIELD)," (accessed Jan 3, 2024). [Online]. Available: https://spaceshield.esa.int/

[13] Google, "Secure AI Framework Approach – A quick guide to implementing the Secure AI Framework (SAIF)," Alphabet, Inc., Tech. Rep., 2023. [Online]. Available: https://services.google.com/fh/files/blogs/google_secure_ai_framework_approach.pdf

[14] Y. Hu, W. Kuang, Z. Qin, K. Li, J. Zhang, Y. Gao, W. Li, and K. Li, "Artificial Intelligence Security: Threats and Countermeasures," *ACM Computing Surveys*, vol. 55, no. 1, pp. 20:1–20:36, 2021. [Online]. Available: https://doi.org/10.1145/3487890

[15] ISO Central Secretary, "Information Technology – Artificial Intelligence – Guidance on Risk Management," International Organization for Standardization (ISO), Tech. Rep. ISO/IEC 23894:2023, 2023. [Online]. Available: https://www.iso.org/standard/77304.html

[16] ——, "Security and Privacy in Artificial Intelligence Use Cases – Best Practices," International Organization for Standardization (ISO), Tech. Rep. ISO/IEC TR 27563:2023, 2023. [Online]. Available: https://www.iso.org/standard/80396.html

[17] ——, "Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE) — Quality Model for AI Systems," International Organization for Standardization (ISO), Tech. Rep. ISO/IEC 25059:2023(E), 2023. [Online]. Available: https://www.iso.org/standard/80655.html

[18] M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, and B. Li, "Manipulating Machine Learning: Poisoning Attacks and Countermeasures for Regression Learning," in *Proceedings of the 39th IEEE Symposium on Security and Privacy (S&P)*. IEEE, 2018, pp. 19–35. [Online]. Available: https://doi.org/10.1109/SP.2018.00057

[19] N. Kaloudi and J. Ji, "The AI-Based Cyber Threat Landscape: A Survey," *ACM Computing Surveys*, vol. 51, no. 1, pp. 20:1–20:34, 2020. [Online]. Available: https://doi.org/10.1145/3372823

[20] B. Kim, A. Abuadbba, Y. Gao, Y. Zheng, M. E. Ahmed, S. Nepal, and H. Kim, "Decamouflage: A Framework to Detect Image-Scaling Attacks on CNN," in *Proceedings of the 51st IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, 2021, pp. 63–74. [Online]. Available: https://doi.org/10.1109/DSN48987.2021.00023

[21] McKinsey & Company, "The State of AI in 2022—and a Half Decade in Review," McKinsey Global Survey on AI, December 2022. [Online]. Available: https://www.mckinsey.com/∼/media/mckinsey/business%20functions/quantumblack/our%20insights/the%20state%20of%20ai%20in%202022%20and%20a%20half%20decade%20in%20review/the-state-of-ai-in-2022-and-a-half-decade-in-review.pdf

[22] J.-G. Meß, F. Dannemann, and F. Greif, "Techniques of Artificial Intelligence for Space Applications - A Survey," in *Proceedings of the 1st European Workshop on On-Board Data Processing (OBDP)*. ESA, 2019, pp. 1–14. [Online]. Available: https://indico.esa.int/event/225/contributions/4289/attachments/3361/5388/OBDP2019-paper-DLR_Mess_Techniques_of_Artificial_Intelligence_in_Space_Applications-A_Survey.pdf

[23] Meta Platforms, Inc., "Llama 2," (accessed Jan 3, 2024). [Online]. Available: https://ai.meta.com/llama/

[24] J. Morris, E. Lifland, J. Y. Yoo, and J. Grigsby, "TextAttack: A Framework for Adversarial Attacks, Data Augmentation, and Adversarial Training in NLP," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2020, pp. 119–126. [Online]. Available: https://aclanthology.org/2020.emnlp-demos.16.pdf

[25] Neuraspace, "Space Collision Avoidance with AI/ML," (accessed Jan 3, 2024). [Online]. Available: https://www.neuraspace.com/product

[26] NIST Trustworthy & Responsible Artificial Intelligence Resource Center (AIRC), "Artificial Intelligence Risk Management Framework (AI RMF 1.0)," National Institute of Standards and Technology (NIST), Tech. Rep., 2023. [Online]. Available: https://doi.org/10.6028/NIST.AI.100-1

[27] OpenAI LLC, "ChatGPT – Release Notes," (accessed Jan 3, 2024). [Online]. Available: https://help.openai.com/en/articles/6825453-chatgpt-release-notes

[28] A. Oseni, N. Moustafa, H. Janicke, P. Liu, Z. Tari, and A. Vasilakos, "Security and Privacy for Artificial Intelligence: Opportunities and Challenges," *Journal of the ACM*, vol. 37, no. 4, pp. 111:1–111:35, 2020. [Online]. Available: https://doi.org/10.1145/1122445.1122456

[29] OWASP Foundation, Inc., “OWASP AI Security and Privacy Guide,” (accessed Jan 5, 2024). [Online]. Available: https://owasp.org/www-project-ai-security-and-privacy-guide/

[30] W. Pearce, H. Anderson, R. S. S. Kumar, N. Coles, A. Paverd, A. Marshall, A. Bhateja, and A. Sutton, “AI Security Risk Assessment – Best Practises and Guidance to Secure AI Systems v4.1.4,” Microsoft Security, Tech. Rep., 2021. [Online]. Available: https://github.com/Azure/AI-Security-Risk-Assessment/blob/main/AI_Risk_Assessment_v4.1.4.pdf

[31] E. Salkield, S. Köhler, S. Birnbach, R. Baker, M. Strohmeier, and I. Martinovic, “Firefly: Spoofing Earth Observation Satellite Data through Radio Overshadowing,” in *Proceedings of the 1st Workshop on the Security of Space and Satellite Systems (SpaceSec)*. Internet Society, 2023, pp. 1–10. [Online]. Available: https://www.ndss-symposium.org/wp-content/uploads/2023/06/spacesec2023-231879-paper.pdf

[32] H. Sathaye, M. Strohmeier, V. Lenders, and A. Ranganathan, “An Experimental Study of GPS Spoofing and Takeover Attacks on UAVs,” in *Proceedings of the 31st USENIX Security Symposium (USENIX Security)*. USENIX Association, 2022, pp. 3503–3520. [Online]. Available: https://www.usenix.org/system/files/sec22-sathaye.pdf

[33] R. Siegfriedt, S. Chien, D. Gaines, S. Kuhn, J. Hazelrig, J. Biehl, A. Connell, R. Francis, and N. Waldram, “Mars 2020 Onboard Planner – Update And Preparations For Operations,” in *Proceedings of the 17th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA)*. ESA, 2023, pp. 1–6. [Online]. Available: https://ai.jpl.nasa.gov/public/documents/papers/M2020-SP-ASTRA-2023.pdf

[34] Y. Son, H. Shin, D. Kim, Y. Park, J. Noh, K. Choi, J. Choi, and Y. Kim, “Rocking Drones with Intentional Sound Noise on Gyroscopic Sensors,” in *Proceedings of the 24th USENIX Security Symposium (USENIX Security)*. USENIX Association, 2015, pp. 881–896. [Online]. Available: https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-son-updated.pdf

[35] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, “MITRE ATT&CK®: Design and Philosophy,” The MITRE Corporation, Tech. Rep., 2020. [Online]. Available: https://attack.mitre.org/docs/ATTACK_Design_and_Philosophy_March_2020.pdf

[36] L. Sweeney, “Simple Demographics Often Identify People Uniquely,” Carnegie Mellon University, Tech. Rep., 2000. [Online]. Available: https://dataprivacylab.org/projects/identifiability/paper1.pdf

[37] The Adversarial Robustness Toolbox (ART) Authors, “Adversarial Robustness Toolbox: A Python library for ML Security,” (accessed Jan 5, 2024). [Online]. Available: https://adversarial-robustness-toolbox.org/

[38] The Aerospace Corporation, “Space Attack Research & Tactic Analysis (SPARTA),” (accessed Jan 3, 2024). [Online]. Available: https://sparta.aerospace.org/

[39] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, “Stealing Machine Learning Models via Prediction APIs,” in *Proceedings of the 25th USENIX Security Symposium (USENIX Security)*. USENIX Association, 2016, pp. 601–618. [Online]. Available: https://www.usenix.org/system/files/conference/usenixsecurity16/sec16_paper_tramer.pdf

[40] US Department of Energy, “DOE AI Risk Management Playbook (AIRMP),” (accessed Jan 5, 2024). [Online]. Available: https://www.energy.gov/ai/doe-ai-risk-management-playbook-airmp

[41] W. Wahlster and C. Winterhalter, “German Standardization Roadmap on Artificial Intelligence (2nd edition),” DIN e.V., DKE: German Commission for Electrical, Electronic & Information Technologies of DIN and VDE, Tech. Rep., 2022. [Online]. Available: https://www.din.de/go/roadmap-ai

[42] E. Wallace, S. Feng, N. Kandpal, M. Gardner, and S. Singh, “Universal Adversarial Triggers for Attacking and Analyzing NLP,” in *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, 2019, pp. 2153–2162. [Online]. Available: https://doi.org/10.18653/v1/D19-1221

[43] Z. Wu, S.-N. Lim, L. S. Davis, and T. Goldstein, “Making an Invisibility Cloak: Real World Adversarial Attacks on Object Detectors,” in *Proceedings of the 16th European Conference on Computer Vision (ECCV)*. Springer, 2020, pp. 1–17. [Online]. Available: https://doi.org/10.1007/978-3-030-58548-8_1

[44] L. Yang, Z. Chen, J. Cortellazzi, F. Pendlebury, K. Tu, F. Pierazzi, L. Cavallaro, and G. Wang, “Jigsaw Puzzle: Selective Backdoor Attack to Subvert Malware Classifiers,” in *Proceedings of the 44th IEEE Symposium on Security and Privacy (SP)*. IEEE, 2023, pp. 719–736. [Online]. Available: https://doi.org/10.1109/SP46215.2023.10179347