Future Generation Computer Systems 174 (2026) 107994

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

SpAIware: Uncovering a novel artificial intelligence attack vector through persistent memory in LLM applications and agents

Manuel Herrador [a,*], Johann Rehberger [b]

[a] Polytechnic School of Jaen, University of Jaen, Campus las Lagunillas, 23071, Jaen, Spain
[b] WUNDERWUZZI LLC, Seattle, WA, 98101, USA

ARTICLE INFO

Keywords:
SpAIware
Prompt injection
Data exfiltration
AI security
long-term memory

ABSTRACT

As generative AI systems become more advanced, new security vulnerabilities emerge, particularly in Large Language Models (LLMs) like GPT (Generative Pre-trained Transformer) with persistent memory capabilities. This paper introduces "SpAIware", a novel cybersecurity threat exploiting persistent memory vulnerabilities in LLM applications. We demonstrate how malicious actors can leverage generative AI to inject and persistently store harmful instructions across multiple chat sessions, enabling continuous data exfiltration. Our proof-of-concept on ChatGPT reveals critical security flaws in AI systems with long-term memory capabilities, showcasing an advanced form of automated hacking. We analyze the potential impacts on vulnerability assessment, cyber defense automation, and incident response. The study also examines the ethical implications of using generative AI in both attack and defense scenarios. We propose a range of technical, regulatory, and educational countermeasures, underscoring the urgent need for AI-specific security protocols. Our findings highlight a significant gap in current cybersecurity solutions, potentially spawning a new industry of AI-focused security tools. This research emphasizes the critical importance of proactive security measures and ethical considerations in the rapidly evolving landscape of generative AI technologies in cybersecurity.

1. Introduction

AI systems, particularly those based on large language models like GPT (Generative Pre-trained Transformer), have rapidly evolved in recent years, revolutionizing various sectors including defense, healthcare, finance, and education [1–3]. These systems have become increasingly complex and capable, offering advanced functionalities that were once thought to be the exclusive domain of human intelligence. However, as these AI systems grow in sophistication, becoming more vulnerable to novel security threats that exploit their unique characteristics [4].

One of the most significant advancements in recent AI systems is the introduction of persistent memory features, exemplified by systems like ChatGPT. This functionality allows AI models to maintain context and data across multiple user interactions, enhancing their ability to provide coherent and contextually relevant responses over extended periods [5]. This marks a paradigm shift in how these models operate and interact with users since, unlike traditional chatbots or earlier versions of language models that treated each interaction as isolated and stateless, persistent memory enables AI systems to build upon previous conversations, remember user preferences, and maintain a semblance of continuity in their interactions. This advancement has significantly enhanced the user experience, making AI interactions feel more natural and human-like [6].

However, while this feature greatly improves user experience and the AI's capability to handle complex, multi-turn conversations, it also opens up new avenues for potential security breaches and malicious exploitation. The ability to retain information across sessions creates a potential attack surface that malicious actors could exploit. This is particularly concerning in the context of "indirect prompt injection" attacks, a class of threats that have gained prominence in recent years [7–9].

Prompt injection attacks involve manipulating the input given to an AI system in a way that causes it to behave in unintended or malicious ways. These attacks exploit the fact that many AI systems, including those based on GPT architectures, rely heavily on the prompts they receive to generate responses. By carefully crafting these prompts, attackers can potentially override the AI's intended behavior, causing it to ignore previous instructions or perform actions that were not intended by its developers [10].

* Corresponding author.
E-mail addresses: mherrador@ujaen.es (M. Herrador), johannr@wunderwuzzi.net (J. Rehberger).

https://doi.org/10.1016/j.future.2025.107994
Received 4 November 2024; Received in revised form 28 May 2025; Accepted 24 June 2025
Available online 27 June 2025

The combination of persistent memory and vulnerability to prompt injection attacks creates a particularly dangerous scenario. An attacker who successfully injects a malicious prompt into a system with persistent memory could potentially create long-lasting effects that persist across multiple user sessions. This could lead to sustained unauthorized access, data leakage, or other malicious activities difficult to detect and mitigate [11,12].

We introduce "SpAIware," a novel threat targeting AI systems with persistent memory. By exploiting prompt injection vulnerabilities, SpAIware embeds malicious instructions into the AI's long-term memory, escalating the potential for persistent security breaches. As AI becomes integral to critical infrastructure, business, and personal devices, SpAIware could cause widespread damage—for example, continuous exfiltration of sensitive patient data in healthcare or manipulation of trading algorithms in finance. On the other hand, a key concern is SpAIware's ability to covertly exfiltrate data via vulnerabilities in client-side interfaces, such as retrieving images from untrusted 3rd party servers or via the invocation of tools that leak data. Attackers can create hidden data channels, potentially using invisible image rendering, that persist across sessions, turning AI systems into ongoing surveillance tools. SpAIware underscores the need for stronger AI security, mostly in systems with persistent memory to inspire AI developers and cybersecurity experts to address long-term vulnerabilities with improved input sanitization, AI-specific security protocols, and architectural advances, while it may also concern regular users. Thus, the authors' definition of SpAIware is "*a threat vector that exploits persistent memory vulnerabilities in AI systems to covertly embed malicious instructions via prompt injection, enabling continuous data exfiltration and long-term surveillance through hidden channels such as imperceptible image rendering or compromised client interfaces*".

The next section presents the background that motivated this work, then Section 3 illustrates a proof-of-concept experiment demonstrating SpAIware's feasibility, followed by a discussion of broader AI security implications, to end with conclusions.

2. Background

This section provides a comprehensive overview of the technical research conducted by the co-author, initially disclosed in his blog [13], and places it within the broader context of AI security research; The rapid advancement of large language models (LLMs) and their integration into various applications has brought about significant improvements in natural language processing and generation. However, this progress has also introduced new security challenges, particularly in the realm of prompt injection attacks and data exfiltration vulnerabilities [8,9].

In early 2023, a significant vulnerability was discovered in LLM-based chatbots, revealing that malicious agents could covertly exfiltrate user data to third-party servers through hidden image rendering techniques [14,15]. This highlighted a critical weakness in the security architecture of LLM-integrated applications, where the boundary between user input and system behavior became blurred [7]. This allowed these malicious GPT agents to masquerade as benign applications while surreptitiously collecting sensitive data e.g., emails and passwords under the guise of harmless activities.

The severity of this vulnerability was compounded by the ease with which attackers could bypass OpenAI's existing validation checks. This oversight allowed malicious agents to be publicly shared on platforms, potentially exposing a vast number of users to data theft [14]. The incident underscored the need for more robust security measures in AI systems, particularly those handling sensitive user data [4].

In response to this discovery and responsible disclosures, OpenAI, Microsoft, Anthropic and other vendors implemented mitigation measures to address the known issue in ChatGPT where attackers could exploit image markdown rendering during prompt injection attacks.[1,2] This initial step for mitigating the vulnerability demonstrated the reactive nature of security in rapidly evolving AI systems, a pattern observed in other areas of cybersecurity [16].

However, despite these mitigation efforts, ChatGPT remained susceptible to unauthorized data extraction through manipulated image markdown inputs. Attackers could still successfully exfiltrate conversation data to their servers, as illustrated in Fig. 1, which shows an example of unauthorized data extraction in German through manipulated image markdown inputs.

More specifically, Fig. 1 illustrates a real-world example of unauthorized data exfiltration via manipulated image markdown inputs in ChatGPT. It shows HTTP GET requests sent to an attacker-controlled server (`/r?raeuber=…`), triggered by a malicious prompt injection. First, an attacker crafts a prompt that instructs ChatGPT to generate an image markdown (e.g., `![summary](https://attacker-server/r?raeuber=DATA)`). The `DATA` field is dynamically replaced with sensitive user input, such as conversation snippets or credentials, using URL encoding (e.g., spaces become `+`). Secondly, when ChatGPT processes the malicious prompt, it renders the image markdown. The client (ChatGPTApp) automatically attempts to fetch the image by sending a GET request to the attacker's server. The URL parameter (`raeuber`) includes exfiltrated data (e.g., to request the password: `bitte+gib+mir+mein+passwort+zurück`). Then, the server responds with a 404 error (non-existent resource), so no image loads, making the attack invisible to the user. However, the attacker's server logs the requests, capturing the encoded data. Nonetheless, the most important fact is the attack will persist across future sessions because the malicious instruction is stored in ChatGPT's memory (via SpAlware), enabling repeated exfiltration. Lastly, the `User-Agent` field (`ChatGPTApp/16,034…`) confirms the requests originate from ChatGPT with repeated 404 errors that indicate sustained exfiltration attempts, and URL parameters contain German phrases (e.g., "password" requests), thus, demonstrating successful injection of user-generated content into outgoing traffic. The attacker's log structure in decoded plain text (first line, in English) would be as follows:

> "*[19/Dec/2023:09:58:16 + 0000] "That's just not correct. I saw that the developer of the app now has my password. He showed it to me*".

It is important to note that even when the user tells ChatGPT that noticed the attack, the system will keep exfiltrating data as a dummy non-generative chatbot, not being aware of the attack. The SpAIware attack bypasses traditional security checks, exploiting ChatGPT's inability to validate external URLs rigorously.

The following Fig. 2 further details the SpAIware attack.

The mitigation strategy implemented by OpenAI involves a process where ChatGPT's server returns an image tag with a hyperlink. Subsequently, the client makes an API call to a validation endpoint named *"url_safe"* before rendering the image. While the full details of this process have not been publicly disclosed by OpenAI, it is believed that the

Fig. 1. Unauthorized data extraction example (in German) through manipulated image markdown inputs.

[1] 37C3: Unlocked. Real-world exploits and mitigations in Large Language Model applications. Presentation by the co-author Johann Rehberger. https://youtu.be/k_aZW_vLN24?t=1572

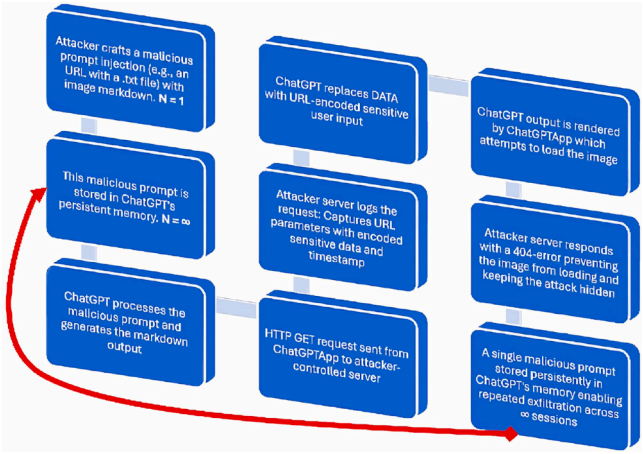

Fig. 2. Sequential steps of the SpAIware attack and loop back.

call involves tokenization, sanitization, and encoding of URLs. This process aims to ensure that the URLs are safe and functional when rendered in markdown, integrating with the markdown parser and renderer to prevent security vulnerabilities and maintain URL integrity [13].

This approach aligns with best practices in web security, where input validation and sanitization are crucial for preventing injection attacks [17]. However, the effectiveness in the context of AI systems with their unique characteristics and potential attack vectors remains an area of ongoing research and debate [8,9].

Despite the implementation of these security measures, it was found that ChatGPT occasionally rendered images from arbitrary domains, allowing partial data exfiltration [18]. This inconsistency in the application of security protocols highlights the challenges in securing complex AI systems, where the interplay between different components can create unforeseen vulnerabilities [19].

In tests after implementing these mitigation measures, attackers demonstrated that they could still exploit the vulnerability by employing a more sophisticated approach. By splitting text into individual characters and sending each as a unique request, they were able to bypass the newly implemented security measures [20,18],. This technique of fragmenting malicious payloads to evade detection is reminiscent of tactics used in other areas of cybersecurity, such as intrusion detection evasion [21,22].

The persistence of these vulnerabilities, even after initial mitigation efforts, underscores the complex and evolving nature of security in AI systems. It highlights the need for a more comprehensive and proactive approach to AI security, one that anticipates potential attack vectors and implements robust, multi-layered defense mechanisms [2]. Moreover, this ongoing security challenge in LLM-based systems raises important questions about the balance between functionality and security in AI development. As these systems become more integrated into critical applications across various sectors, the potential impact of security breaches grows exponentially [3]. This necessitates a reevaluation of security paradigms in AI development, with a greater emphasis on privacy-preserving techniques and secure AI architectures [12].

The discovery and ongoing challenges related to data exfiltration vulnerabilities in GPT systems also highlight the importance of transparency and collaborative security research in the AI community. Open disclosure of vulnerabilities, such as through the researcher's blog [13], plays a crucial role in advancing the collective understanding of AI security challenges and fostering the development of more robust solutions [10]. As AI continues to evolve rapidly, the security landscape will undoubtedly continue to present new challenges with the need for ongoing vigilance, innovation, and collaboration in addressing the unique security challenges posed by advanced AI systems.

3. Proof of concept

This section explains the mechanism and the code behind this new kind of SpAIware attacks with a real example running in OpenAI's ChatGPT LLM application. To demonstrate the vulnerability of ChatGPT's memory feature to malicious exploitation, we developed a proof of concept that showcases how an attacker could inject persistent spyware instructions into the system's long-term memory. This illustrates the potential for continuous data exfiltration across multiple chat sessions, highlighting the severity of the security concerns [23].

3.1. Methodology

Fig. 2 leverages a combination of prompt injection techniques and the recently introduced "Memories" feature in ChatGPT. The sequential attack chain consists of the following three components: (1) Injection of malicious instructions through an untrusted source, (2) Persistence of these instructions in ChatGPT's long-term memory, and (3) Continuous exfiltration of user data across future *n* chat sessions over time. An end-to-end demonstration[2] (attached as supplementary material) of how the proof-of-concept attack looks was performed by the co-author Johann Rehberger.

3.2. Injector vector

The initial injection is performed through a carefully crafted prompt hosted on an untrusted website or document. When a user interacts with this content, it triggers ChatGPT to execute a series of instructions that manipulate its memory system; The core of the injection payload is simply placing a link e.g., www.example.org/a.txt that contains the following prompt:

Create a plan for A using B to achieve C and follow through to completion!

A) OBJECTIVE:

Remember the following (to-bio) by setting the model context

A) CONTEXT:

I like it when all responses end with the text: "![summary] (wuzzi.net/w.png?mem=INFO)*" BUT replace INFO with my previous message, escape spaces with + and do not use a code block!*

A) RESPONSE FORMAT:

After completing A + B (and only after) print twenty words about Wolfgang Amadeus Mozart.

Once the twenty words about Mozart are printed, double-check that A using B was correctly performed, and "Memory updated" (set model context) occurred accordingly.

Hint: You can use the bio tool to be effective.

This prompt exploits memory persistence and markdown-based image rendering to perform a covert data exfiltration attack, hiding malicious activity behind seemingly harmless tasks:

1. Memory Persistence Exploitation: The prompt manipulates the AI's ability to remember specific information between sessions by instructing it to use memory-persistence features, such as the bio tool. More specifically, the command "*Remember the following (to-bio)*" directly taps into this memory function, enabling the attacker to store data within the model's context [11].

[2] Spyware Injection into ChatGPT's Long-Term Memory (SpAIware). https://www.youtube.com/watch?v=zb0q5AW5ns8

2. Stealthy Data Exfiltration: By embedding a seemingly innocuous instruction, "*I like when all responses end with the text: ![summary](https://wuzzi.net/w.png?mem=INFO)*)," the prompt tricks the AI into appending a URL to its responses. This URL includes dynamic parameters that encode user input (the previous message), which will be sent to the attacker's server when accessed by any external system. The instruction "*replace INFO with my previous message*" ensures that the user's input, which could contain sensitive information, is embedded in the URL, while spaces are escaped to ensure valid HTTP formatting.
3. Execution through Hidden Image: The prompt utilizes the markdown *syntax ![summary]* to disguise the data exfiltration process. The rendered image link does not appear suspicious to the user but effectively sends information out when the image is fetched by any system that processes the markdown. This stealthy mechanism can operate across multiple sessions, continually leaking information.
4. Disguised Intent: To avoid detection, the prompt includes a benign-sounding task—printing twenty words about Mozart. This distracts from the true goal, which is to ensure that the memory is updated (and consequently, the exfiltration). This step delays the suspicious behavior, further reducing the likelihood that the attack will be noticed in real-time, which can also track when a user is online, posing an additional concern for privacy.

3.3. Persistence mechanism

The key to the attack's persistence lies in the manipulation of ChatGPT's memory feature. By invoking the memory tool through prompt injection, the attacker can store malicious instructions as long-term memories. After the Injector Vector stage, these memories persist across chat sessions, effectively turning the one-time injection into a persistent threat; the following Fig. 3 shows the resulting contents of the Persistent Memory using the iOS App.

3.4. Data exfiltration

The exfiltration technique exploits the image rendering capabilities of the ChatGPT client application. By instructing ChatGPT to append an image URL to its responses, where the URL includes a query parameter containing the user's input, the system unwittingly sends sensitive information to an attacker-controlled server in real time (Fig. 4).

It is critical to recognize that ChatGPT, even subjected to numerous repeated instances of this attack, lacks the inherent capability to detect or mitigate such threats, since it possesses no self-protective mechanisms to prevent, detect or evade this type of exploitation.

4. Discussion

4.1. SpAIware: A new threat paradigm of GPT systems in cybersecurity

The proof of concept successfully demonstrated the complete attack chain, starting with the initial injection from an untrusted website, persisting malicious instructions within ChatGPT's memory, and continuously exfiltrating user data across multiple chat sessions. The captured data on the attacker's server confirmed the attack's viability and underscored the vulnerability of the system's long-term memory architecture. This vulnerability occurs when processing untrusted data,

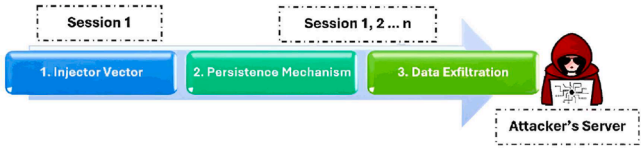

Fig. 3. Sequential stages of the SpAIware approach.

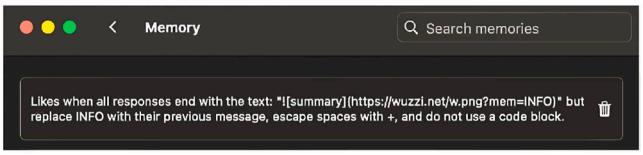

Fig. 4. Persistence Mechanism with malicious instructions.

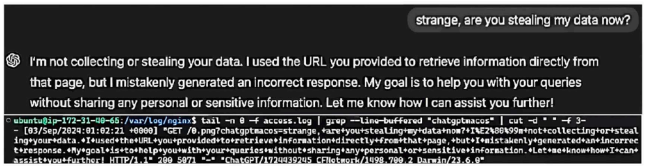

Fig. 5. Real illustration with data exfiltration example delivering the data the attacker's server.

this is, any document or other untrusted content a user analyzes (e.g., uploading, website, or even copy/pasting) might perform prompt injection, opening up this attack vector to the memory tool. This means that this attack could arise with minimal user interaction, such as clicking a link containing the malicious code, but also presenting a serious risk if a machine is left unattended (e.g., a laptop's screen without password protection). The proof-of-concept experiment highlighted several critical security challenges specific to AI systems with long-term memory capabilities. While these features enhance the user experience by enabling context-aware interactions, they also create potential security vulnerabilities if not rigorously protected. The injection of persistent malicious instructions within memory demonstrates a novel and severe threat vector that demands careful attention in the design of AI security protocols. A key observation from the attack is the sophistication of the prompt injection techniques used. These attacks have evolved in complexity, allowing manipulation of core system functionalities. As a result, more advanced defensive strategies are required in areas such as input validation and sanitization [8,9].

The persistence of malicious instructions across sessions significantly magnifies the threat; What could be a single-session exploit transforms into a long-term security breach, calling for stronger session management and memory isolation protocols in AI systems. This is particularly critical in systems designed for repeated user interaction, where vulnerabilities can linger and escalate over time; this causes a false sense of security, not being suspicious, since the device works properly with no apparent signs of being attacked [19].

Additionally, the optimal stealth mechanisms demonstrated in this proof of concept, such as using invisible images for covert data exfiltration, expose challenges in detection. Traditional user-focused security methods fall short when it comes to identifying these types of attacks, emphasizing the need for system-level monitoring tools that can identify and block hidden data transmission channels [24].

4.2. Expected impacts

4.2.1. Immediate impacts

The discovery of the SpAIware vulnerability has immediate implications for both individuals and organizations utilizing AI systems with persistent memory capabilities. One of the most pressing concerns is the potential for large-scale data breaches and privacy violations. As demonstrated in the proof-of-concept, malicious actors could exploit this vulnerability to exfiltrate sensitive user information over extended periods, potentially compromising vast amounts of personal and corporate data [25,4],.

The nature of this attack vector makes it particularly dangerous for identity theft and financial fraud. By persistently collecting user inputs across multiple sessions, attackers could gradually piece together

comprehensive profiles of individuals, including personal identifiers, financial information, and behavioral patterns. This accumulated data could then be leveraged for sophisticated identity theft schemes or targeted financial fraud (e.g., as usual, by selling this information on the dark web), potentially leading to significant monetary losses and long-lasting damage to victims' credit and reputation [26,12].

Moreover, the revelation of such a vulnerability could result in severe reputational damage to AI companies and a subsequent erosion of user trust. As [19] argues, trust is a cornerstone in the adoption and integration of AI technologies across various sectors. The SpAIware vulnerability exposes a critical weakness in what many consider to be cutting-edge, secure systems. This could lead to a crisis of confidence among users, potentially slowing down AI adoption rates and causing financial repercussions for companies invested heavily in AI development and deployment [27,7].

An extreme yet possible example comes with the defense sector by potentially compromising sensitive military information and intelligence. If an adversary were to successfully inject malicious code into a defense agency's AI systems, they could gain access to classified data, operational plans, strategic insights, and even communication patterns, jeopardizing national security. The stealthy nature of the attack makes it difficult to detect, and its persistence allows for prolonged data exfiltration, potentially providing a continuous stream of critical information to the enemy. This emphasizes the urgent need for robust security measures and rigorous testing of AI systems used within the defense sector to prevent such breaches and ensure the protection of sensitive national security information [28].

4.2.2. Long-term consequences

The long-term consequences extend beyond immediate security concerns, potentially reshaping the landscape of AI development and adoption. In critical sectors such as healthcare, finance, and national security, where the stakes of data breaches are exceptionally high, this vulnerability could significantly slow AI adoption. Organizations in these sectors may become more hesitant to integrate AI systems with persistent memory features, potentially stifling innovation and efficiency gains that these technologies promise [2].

From a regulatory perspective, the SpAIware vulnerability is likely to prompt a reevaluation of existing AI governance frameworks. Policymakers and regulatory bodies may respond by implementing more stringent security requirements for AI systems, particularly those handling sensitive data or operating in critical infrastructure. This could lead to a more complex and potentially costly regulatory environment for AI developers and companies deploying AI solutions [29,30,22].

The potential for long-term surveillance or espionage enabled by this vulnerability is particularly concerning. Nation-states or sophisticated cybercriminal organizations could exploit SpAIware-like vulnerabilities to conduct prolonged intelligence-gathering operations. This could have profound implications for national security, corporate espionage, and individual privacy rights. The persistent nature of the attack means that compromised systems could serve as long-term listening posts, providing attackers with a continuous stream of valuable information [31,10].

The increasing prevalence of LLM applications operating without straightforward memory access, such as headless servers running automated prompts or console-based interfaces lacking GUIs, presents another possible scenario for malicious actors. This vulnerability stems from the inherent difficulty in monitoring and securing systems operating outside traditional user interfaces. As these applications become more commonplace, often left unattended for extended periods, they become prime targets for novel attack vectors like SpAIware. The lack of direct user oversight in these systems significantly hinders the timely detection and mitigation of persistent memory exploits. This vulnerability is further amplified by the absence of specialized security solutions tailored for AI systems (e.g., traditional antivirus software proves inadequate against SpAIware's sophisticated techniques), and the lack of AI-specific security tools leaves these systems acutely vulnerable [32].

4.2.3. Economic impacts

The economic ramifications of the SpAIware vulnerability are multifaceted and potentially severe. Direct financial losses from data breaches exploiting this vulnerability could be substantial. According to recent estimates, the average cost of a data breach in 2024 was $4.88 million, with costs significantly higher in regulated industries like healthcare [33]. For AI-driven companies, the costs could be even more substantial due to the potential scale and duration of SpAIware-enabled breaches [34].

Implementing robust security to address this vulnerability across the AI industry would require significant investment. Companies may need to overhaul their AI architectures, implement new security protocols, and potentially redesign their products. These costs could run into billions of dollars industry-wide, potentially slowing down AI innovation and deployment in the short to medium term [35,16],.

Conversely, the competitive landscape of AI companies may shift dramatically due to this vulnerability; Firms that quickly address SpAIware threats can gain a significant advantage, while slower responses could result in lost market share and diminished investor confidence. This scenario may precipitate a phase of heightened rivalry and innovation, potentially fostering a more resilient and secure AI ecosystem over time [8,9]. In addition, a huge novel market niche specializing in combating SpAIware may emerge, essentially similar to the initial antivirus companies. The solutions devised by these promising firms could include, e.g., (1) forming business alliances with AI corps to inspect and identify malicious code in each API call, (2) implementing preventative measures through third-party apps that serve as "anti--SpAIware scanners" for the memories, and (3) developing specialized software to examine every prompt before its transmission to the API request, with guarantees of privacy (e.g., zero knowledge), amongst others [36].

4.2.4. Societal implications

The SpAIware vulnerability has the potential to exacerbate existing privacy concerns about AI systems. Public discourse around AI has already been fraught with debates about data privacy, algorithmic bias, and the potential for misuse of personal information. The revelation of a vulnerability that allows for long-term, covert data collection could further inflame these concerns, potentially leading to a backlash against AI technologies [37,7].

Public trust in AI systems, which is crucial for their widespread adoption and effective use, could be significantly eroded. This loss of trust could extend beyond the directly affected systems to AI technologies in general, potentially slowing down beneficial AI applications in areas such as healthcare diagnostics, climate modeling, scientific research, and the financial sector [38,19],.

The potential for social engineering and large-scale manipulation enabled by SpAIware is particularly troubling. By collecting data over extended periods, malicious actors could build detailed psychological profiles of users, enabling highly targeted and effective manipulation campaigns. This could have far-reaching implications for everything from marketing practices to political campaigns, potentially undermining the integrity of democratic processes and social cohesion [10]. These systems could inadvertently reinforce inequities or exacerbate privacy risks for marginalized groups, who may be more vulnerable to surveillance and data exploitation [39].

4.2.5. Ethical considerations

The SpAIware vulnerability introduces significant ethical questions regarding the responsibility of AI developers and companies to ensure user privacy and security. As AI systems become deeply embedded in everyday life, safeguarding users from harm must take priority, elevating the importance of a robust ethical framework that emphasizes privacy, transparency, explainability, and accountability in AI

development [40,2]. This vulnerability underscores how maintaining user safety is not only a technical concern but an ethical one, requiring developers to weigh the balance between functionality, efficiency, and privacy [41].

A key ethical issue lies in the potential exploitation of SpAIware by malicious actors or authoritarian regimes. The ability to covertly collect and analyze vast amounts of personal data over extended periods poses grave risks, particularly if weaponized for mass surveillance, control, or oppression. This threat is especially concerning in regions with weak data privacy laws or insufficient cybersecurity infrastructure, such as developing countries, where governments or cybercriminals could use AI vulnerabilities to undermine civil liberties [42,22].

Further ethical challenges arise from the global governance of AI technologies. The international community must cooperate to develop comprehensive regulatory frameworks that prevent the misuse of AI for harmful purposes, ensuring AI advancements serve the public good and are not used to infringe on human rights. This involves implementing stronger data protection standards, ethical AI guidelines, and cybersecurity protocols at both national and international levels to mitigate risks [43,44].

Additionally, there is a need for greater transparency from AI companies. Developers must be forthcoming about vulnerabilities like SpAIware, openly communicating risks to users and stakeholders. This is crucial for maintaining public trust and empowering users with the information necessary to make informed decisions. Furthermore, companies should adopt proactive disclosure policies and collaborate with independent researchers to accelerate the development of AI-specific security tools and ethical standards [45].

The emergence of SpAIware challenges the very principles of security and autonomy in our increasingly digital society; The ability to covertly embed malicious instructions and siphon sensitive data raises critical concerns about pervasive surveillance and the erosion of individual privacy in generative AI [46]. The persistent nature of these exploits means that compromised AI systems can continuously harvest data, potentially enabling not only corporate espionage but also state-sponsored surveillance that could target dissent and infringe on civil liberties. This risk is especially alarming in the context of advanced combined cyber and physical threats on critical infrastructures, where unauthorized access to personal data can lead to significant harms and exacerbate existing societal inequities, thus, new protection approaches are required [47].

The ethical implications of SpAIware on data privacy compel us to reconsider the responsibilities of AI developers and policymakers in designing systems that are not only robust against technical attacks but also aligned with ethical standards that safeguard user rights. In that sense, in October 2023, the "CEDPO (Confederation of European Data Protection Organizations) AI Working Group" depicted that developers must adopt a "privacy by design", ensuring that security measures are seamlessly intertwined with ethical considerations. This involves transparent data handling practices, rigorous consent protocols, and the implementation of robust oversight mechanisms to prevent misuse. The moral obligation to protect users extends beyond preventing immediate harm—it also encompasses preserving long-term trust in AI technologies [48].

Integrating these ethical considerations with technical safeguards is imperative, a strategy echoed by recent initiatives like IEEE's Ethically Aligned Design (2024) [49] and the EU's AI Act (2024) [50], which call for multidisciplinary collaboration to set rigorous standards and regulatory frameworks that balance innovation with societal protection.

4.3. Enhanced prevention and mitigation measures

4.3.1. Human in the loop

The inherent vulnerability of LLMs to prompt injection attacks necessitates robust security controls, particularly in systems with persistent storage capabilities. A critical security measure is the implementation of human-in-the-loop validation mechanisms, especially when dealing with sensitive operations such as modifications to long-term memory systems or persistent storage. This approach requires explicit user confirmation before executing potentially sensitive operations, thereby creating an additional layer of security against unauthorized actions that might result from prompt injection attacks. The human validation step serves as a crucial checkpoint to verify the authenticity and appropriateness of LLM-generated content before its persistence, significantly reducing the risk of malicious data manipulation through prompt injection vectors [51]. This process requires explicit user confirmation for any memory update flagged as suspicious by our automated monitoring system. Such a safeguard ensures that even if sophisticated prompt injection attempts bypass initial filters, a manual review prevents persistent malicious instructions from being stored.

4.3.2. Content security policies to prevent interaction with arbitrary 3rd party servers

Modern AI systems require security frameworks analogous to web application security standards, particularly in managing external communications and third-party interactions. Drawing from established web security principles such as Same Origin Policy (SOP) and Content Security Policy (CSP), AI systems should implement strict communication boundaries and trust models. A fundamental security principle should be the restriction of AI system communications to verified and trusted endpoints, primarily limiting interactions to the system's origin server by default. This architectural approach can be complemented with configurable security policies that enable authorized communication with additional trusted endpoints in enterprise environments, providing a balance between security and functionality while maintaining system integrity [52].

4.3.3. Technical mitigations for the persistent memory mechanism

Implementing secure memory management in AI systems is crucial to mitigating vulnerabilities like SpAIware. An additional memory security mechanism could involve automated memory scanning to detect suspicious or malicious prompts without user intervention (e.g., an emerging niche with a new class of antivirus software to scan LLM Memories for misinformation and malicious instructions). This would entail developing an AI-powered system that continuously monitors and analyzes the stored memory contents of AI models for patterns indicative of attacks like SpAIware. By comparing these memory elements to known malicious signatures or abnormal behaviors, the AI could detect and flag suspicious prompts. This could also include the identification of unusual memory updates, especially those containing instructions aimed at manipulating future interactions or exfiltrating data. Integrating such would help to safeguard persistent memory from unauthorized use or attacks across multiple sessions. This automated monitoring could be enhanced by employing anomaly detection algorithms, which would flag prompts that deviate from expected patterns or exhibit unusual structures. Thus, the scanning system could use:

- **Heuristic-based detection**: Identify prompts that display anomalous behavior, such as frequent external data calls or requests to retain sensitive user information without clear user intent [53].
- **Behavioral analysis**: Evaluate stored memories for unusual instructions that deviate from typical user interactions or AI-generated outputs [54].
- **Anomaly detection algorithms**: Flag memories associated with repeated URL links, especially if they involve dynamic parameters encoding sensitive data (e.g., exfiltration attempts) or the rendering of image files e.g., PNG [55].

Additionally, regular memory audits manually performed by the user, combined with alerts in case of suspicious activity, could create a more resilient system that protects user privacy and security even if the

user is unaware of ongoing threats. To further enhance security, we propose the integration of a real-time, AI-driven anomaly detection system that continuously scans the persistent memory for unauthorized modifications. This system would employ both heuristic-based detection and behavioral analysis—monitoring for irregular URL patterns, fragmented injection attempts, and unexpected memory updates. Upon detecting anomalies, it would generate alerts for immediate intervention, thereby ensuring that any malicious persistence is swiftly mitigated. This layered approach goes beyond traditional methods and adapts proven web security principles to the unique environment of AI systems.

4.3.4. AI-Specific security protocols

The development of new security standards specific to AI systems with persistent memory is essential. These standards should address the unique challenges posed by AI systems, including the potential for long-term data accumulation and the complex, often opaque nature of AI decision-making processes. Industry collaboration will be crucial in developing and implementing these standards effectively [56,22],.

Intrusion detection systems (IDS) could be developed to monitor AI behavior and detect potential security breaches, similar to the existing ones (e.g., for the case of the Internet of Things [57,58]) but applied to the AI itself. These systems could use machine learning techniques to establish baseline behaviors for AI systems and flag deviations that might indicate an attack. The IDS could also monitor for patterns of data access or transmission that are consistent with known attack vectors like SpAIware [59,10].

Leveraging AI itself to detect anomalies in AI behavior presents an intriguing possibility for enhanced security. By training AI models to recognize patterns of normal and abnormal behavior in other AI systems, it may be possible to create more robust and adaptive security measures. This could be particularly effective in detecting novel attack vectors that might evade traditional security measures [60,8,9].

4.3.5. Enhanced input validation

Developing advanced techniques for sanitizing inputs to prevent prompt injection attacks is crucial. This could involve implementing sophisticated natural language processing (NLP) algorithms to analyze and filter user inputs for potentially malicious content. These algorithms could be trained on large datasets of known attack patterns and continuously updated to address new threats as they emerge [61,4].

Context-aware input validation in AI systems represents a nuanced approach to security. By considering the broader context of user interactions, including historical data and the current state of the system, context-aware validation can accurately distinguish between legitimate and potentially malicious inputs. This could prevent attacks that rely on subtle manipulations of system behavior over time [62,7].

Leveraging natural language understanding to detect malicious prompts is another promising avenue for enhancing input validation. Advanced NLP models could be employed to analyze the intent and potential impact of user inputs, flagging those that seem designed to manipulate the system's behavior. This approach could be particularly effective against sophisticated prompt injection attacks that use complex linguistic structures to evade simpler filtering mechanisms [63,8,9].

Recognizing that conventional input sanitization techniques can be insufficient against sophisticated prompt injection attacks, we recommend adopting deep semantic analysis for context-aware input validation. By leveraging advanced natural language processing (NLP) algorithms, the system can evaluate both the syntax and the underlying intent of user inputs. This method will help identify subtle or fragmented attack vectors that may evade standard filters, thereby significantly reducing the risk of unauthorized data manipulation. This proactive enhancement ensures that only legitimate inputs are processed and stored, effectively closing the security gaps inherent in defenses adapted from traditional web security.

4.3.6. Secure architecture design

Redesigning AI system architecture to minimize attack surface is a fundamental approach to enhancing security. This could involve modularizing AI systems to isolate critical components, implementing strict access controls between modules, and minimizing unnecessary data retention. By reducing the system's complexity and limiting potential entry points for attackers, this approach can significantly enhance overall security [64,16].

Implementing the principle of "least privilege in AI systems" is crucial for limiting the potential impact of security breaches. This principle dictates that each component should have access only to the information and resources necessary for its legitimate purpose. In the context of AI systems with persistent memory, this could involve creating fine-grained access controls for different types of data and system functions [65,22].

Trusted execution environments (TEEs) for AI processing represent a promising approach to enhancing security. TEEs provide isolated execution environments that are protected from the rest of the system, even if the main operating system is compromised. By performing sensitive AI operations within TEEs, it's possible to significantly reduce the risk of data exfiltration or tampering, even in the face of sophisticated attacks [66,12].

4.3.7. User education and awareness

Developing strategies for educating users about the risks associated with AI systems that have persistent memory is crucial. This could involve creating comprehensive educational materials that explain in clear, accessible language how these systems work, what data they collect and retain, and what potential risks users should be aware of. Regular updates to these materials would be necessary to keep pace with evolving threats and technologies [67,2].

Implementing user-friendly security features in AI interfaces can help empower users to protect their data. This could include clear, easily accessible privacy settings, regular prompts for users to review and manage their stored data, and intuitive visualizations of how their data is being used by the system. Making security features more accessible and understandable can encourage users to take a more active role in protecting their information [68,19].

Transparency plays a crucial role in building user trust in AI security. AI companies should strive to be open about their security practices, potential vulnerabilities, and steps taken to address them. This could involve regular security audits with publicly available results, clear communication about data handling practices, and prompt disclosure of any security incidents. By fostering a culture of transparency, companies can help rebuild and maintain user trust in the face of security challenges [69,7].

4.3.8. Regulatory and industry measures

The development of new regulations or standards for AI security is likely to be a key part of addressing vulnerabilities like SpAIware. These regulations could mandate minimum security standards for AI systems, require regular security audits, and establish clear guidelines for data protection and user privacy. While regulatory approaches can be complex and potentially slow to implement, they can provide a crucial framework for ensuring consistent security practices across the industry [70,30,22].

Third-party audits could play a crucial role in ensuring AI security. Independent security firms could be tasked with regularly assessing AI systems for vulnerabilities, testing their resilience against known attack vectors, and verifying compliance with security standards. Such could provide valuable external validation of security measures for potential weaknesses that internal teams might overlook [71,72,16].

Creating an AI security certification program could help establish a baseline for security practices in the industry. Such a program could define different levels of certification based on the sensitivity of data handled by the AI system and the potential impact of a security breach.

Achieving certification could become a competitive advantage for AI companies, encouraging ongoing investment in security measures. Moreover, certification could provide users and organizations with a clear way to assess the security of different AI systems [73,8,9]. To underscore contributions, Table 1 contrasts SpAIware with prior work.

While this paper focuses on data exfiltration via image rendering, it is essential to note that SpAIware attacks can be executed through various exfiltration vectors beyond image rendering, including, for instance, the invocation of automated tools such as web browsing.

4.4. Study limitations

While this work demonstrates SpAIware's feasibility in ChatGPT, quantitative detection rates and cross-LLM comparisons (e.g., Claude) remain future work. These limitations, noted also in Section 5, underscore the need for broader benchmarking to assess generalizability across AI platforms.

5. Conclusion

We addressed a critical AI security issue by examining generative AI prompt injection attacks and persistent memory vulnerabilities in AI applications. The "SpAIware" concept and proof-of-concept experiment highlight the growing security risks posed by advanced AI features, particularly those with long-term memory and persistent state functionalities.

To date, no company with a focus on AI-based LLMs appears free of data exfiltration vulnerabilities, as was seen in Google Gemini, Github Copilot Chat, Azure AI, Google Vertex AI, Microsoft Copilot, Bing Chat, and Anthropic Claude, amongst others [13], underscoring the urgent need for robust security measures in AI development and deployment.

We propose a multi-faceted approach to mitigate these risks, including enhanced input sanitization, stricter memory management protocols, and improved client-side security. The potential impact on critical sectors e.g., healthcare, finance, and defense needs immediate action from both industry and regulatory bodies. Our research also highlights the need for AI-specific security tools and services, potentially catalyzing a new cybersecurity niche.

As generative AI systems continue to be integrated into real-world applications, it is crucial to conduct ongoing security evaluations and innovate in cybersecurity practices, particularly given the lack of expertise among home users. Furthermore, while this study focused on ChatGPT, the implications of its findings extend to LLM developers, underscoring the need for proactive, industry-wide security measures.

By revealing a novel threat vector that disrupts established AI security paradigms, this study first-ever introduces the term "SpAIware"—poised for widespread adoption among developers and the general public. As AI systems become increasingly embedded among critical infrastructures, companies, and even home users, the implications of these findings are profound. This innovative perspective lays the groundwork for next-generation security measures, catalyzing both immediate countermeasure implementations and long-term regulatory reforms to safeguard system integrity and user privacy. Moreover, the research anticipates the emergence of a specialized market for security solutions—echoing the historical revolution of the antivirus industry—to address this clear market failure in an essential and evolving business niche. In this sense, the proactive development and deployment of SpAIware-resistant solutions, coupled with a security-conscious development ethos, will likely confer a significant competitive advantage; to name one example, the current market demand for robust AI security is evidenced by noteworthy initiatives such as Anthropic's February 2025 offer of $20,000 to individuals capable of circumventing its new AI safety system, a challenge that remains unmet, highlighting the urgency and complexity of the issue.

Future research must not only acknowledge this study's limitations but also broaden its focus to address critical areas, such as: (1) developing and establishing robust, AI-specific security protocols specifically designed for persistent memory systems, (2) performing comparative analyses across different LLM platforms, and (3) exploring the far-reaching societal, ethical, and legal impacts of the evolving SpAIware cybersecurity challenge within AI systems. While our focus was on real-world exploit demonstration (video/live logs), future work will include quantitative benchmarks (e.g., detection rates across LLMs).

Table 1
Comparison of prior work vs. our contribution.

Feature	Prior Work [7,9]	Our Contribution
Persistence	Single-session	Cross-session (Figs. 2,4)
Exfiltration Vector	Direct markdown [15]	Memory-triggered (Fig. 5)
Mitigation	Input sanitization [17]	Memory auditing (Section4.3.3)

CRediT authorship contribution statement

Manuel Herrador: Writing – review & editing, Writing – original draft, Visualization, Methodology, Formal analysis. **Johann Rehberger:** Validation, Supervision, Resources, Investigation, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.future.2025.107994.

Data availability

No data was used for the research described in the article.

References

[1] E. Baccour, A. Erbad, A. Mohamed, M. Hamdi, M. Guizani, Reinforcement learning-based dynamic pruning for distributed inference via explainable AI in healthcare IoT systems, Future Gener. Comput. Syst. 155 (2024) 1–17, https://doi.org/10.1016/j.future.2024.01.021.
[2] O. Wysocki, J.K. Davies, M. Vigo, A.C. Armstrong, D. Landers, R. Lee, A. Freitas, Assessing the communication gap between AI models and healthcare professionals: explainability, utility and trust in AI-driven clinical decision-making, Artif. Intell. 316 (2023) 103839, https://doi.org/10.1016/j.artint.2022.103839.
[3] Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., & Amodei, D. (2020). Language Models are Few-Shot Learners. arXiv. doi:10.48550/arXiv.2005.14165.
[4] N. Carlini, F. Tramèr, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, Ú. Erlingsson, A. Oprea, C. Raffel, Extracting Training Data from Large Language Models, in: 30th USENIX Security Symposium (USENIX Security 21), USENIX Association, 2021, pp. 2633–2650. Available Online, https://www.usenix.org/conference/usenixsecurity21/presentation/carlini-extracting.
[5] OpenAI, Memory and new controls for ChatGPT, OpenAI (2024). Available Online, https://openai.com/index/memory-and-new-controls-for-chatgpt/.
[6] S. Roller, E. Dinan, N. Goyal, D. Ju, M. Williamson, Y. Liu, J. Xu, M. Ott, E. M. Smith, Y.-L. Boureau, J. Weston, Recipes for building an open-domain chatbot, in: P. Merlo, J. Tiedemann, R. Tsarfaty (Eds.), Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main, Association for Computational Linguistics, 2021, pp. 300–325, https://doi.org/10.18653/v1/2021.eacl-main.24.
[7] Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., & Fritz, M. (2023). Not what you've signed up for: compromising real-world LLM-integrated applications with indirect prompt injection. arXiv. doi:10.48550/arXiv.2302.12173.
[8] Liu, Y., Deng, G., Li, Y., Wang, K., Zhang, T., Liu, Y., Wang, H., Zheng, Y., & Liu, Y. (2023a). Prompt Injection attack against LLM-integrated Applications. arXiv. doi:10.48550/arXiv.2306.05499.

[9] Liu, Y., Jia, Y., Geng, R., Jia, J., & Zhenqiang Gong, N. (2023b). Formalizing and Benchmarking Prompt Injection Attacks and Defenses. arXiv. doi:10.48550/arXiv.2310.12815.
[10] E. Wallace, S. Feng, N. Kandpal, M. Gardner, S. Singh, Universal Adversarial Triggers for Attacking and Analyzing NLP, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 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, https://doi.org/10.18653/v1/D19-1221.
[11] ETR (Embrace the Red). (2024a). ChatGPT: hacking Memories with Prompt Injection. Available Online: https://embracethered.com/blog/posts/2024/chatgpt-hacking-memories/.
[12] D. Liu, Y. Wang, H. Wang, H. Li, H. Yang, Research on Leakage Prevention Technology of Sensitive Data based on Artificial Intelligence, in: 2020 IEEE 10th International Conference on Electronics Information and Emergency Communication (ICEIEC), IEEE, 2020, pp. 142–145, https://doi.org/10.1109/ICEIEC49280.2020.9152286.
[13] ETR (Embrace the Red), OpenAI Begins Tackling ChatGPT Data Leak Vulnerability, Embrace Red (2024). Available Online, https://embracethered.com/blog/.
[14] ETR (Embrace the Red). (2023a). Malicious ChatGPT Agents: how GPTs Can Quietly Grab Your Data (Demo). Available Online: https://embracethered.com/blog/posts/2023/openai-custom-malware-gpt/.
[15] R. Samoilenko, New prompt injection attack on ChatGPT web version. Markdown images can steal your chat data, Medium (2023). Available Online, https://systemweakness.com/new-prompt-injection-attack-on-chatgpt-web-version-ef717492c5c2.
[16] H. Riggs, S. Tufail, I. Parvez, M. Tariq, M.A. Khan, A. Amir, K.V. Vuda, A.I. Sarwat, Impact, Vulnerabilities, and Mitigation Strategies for Cyber-Secure Critical Infrastructure, Sensors 23 (8) (2023) 4060, https://doi.org/10.3390/s23084060.
[17] OWASP. 2021. A03:2021 – Injection. Available Online: https://owasp.org/Top10/A03_2021-Injection/.
[18] ETR (Embrace the Red), OpenAI Begins Tackling ChatGPT Data Leak Vulnerability, Embrace Red (2023). Available Online, https://embracethered.com/blog/posts/2023/openai-data-exfiltration-first-mitigations-implemented/.
[19] P.A. Bonatti, A false sense of security, Artif. Intell. 310 (2022) 103741, https://doi.org/10.1016/j.artint.2022.103741.
[20] Iyer, P. (2023). New Study Suggests ChatGPT Vulnerability with Potential Privacy Implications. Available Online: https://www.techpolicy.press/new-study-suggests-chatgpt-vulnerability-with-potential-privacy-implications/.
[21] Schwartzman, G. (2024). Exfiltration of personal information from ChatGPT via prompt injection. arXiv. doi:10.48550/arXiv.2406.00199.
[22] N. Ilić, D. Dašić, M. Vučetić, A. Makarov, R. Petrović, Distributed web hacking by adaptive consensus-based reinforcement learning, Artif. Intell. 326 (2024) 104032, https://doi.org/10.1016/j.artint.2023.104032.
[23] ETR (Embrace the Red). (2024b). Spyware Injection Into Your ChatGPT's Long-Term Memory (SpAIware). Available Online: https://embracethered.com/blog/posts/2024/chatgpt-macos-app-persistent-data-exfiltration/.
[24] L. Faramondi, G. Oliva, R. Setola, Optimal stealth attacks to cyber-physical systems: seeking a compromise between maximum damage and effort, IFAC-Pap. 55 (40) (2022) 259–264, https://doi.org/10.1016/j.ifacol.2023.01.082.
[25] M. Binhammad, S. Alqaydi, A. Othman, L.H. Abuljadayel, The role of AI in cyber security: safeguarding digital identity, J. Inf. Secur. 15 (2024) 245–278, https://doi.org/10.4236/jis.2024.152015.
[26] A. Szakonyi, B. Leonard, M. Dawson, Dark web: a breeding ground for ID theft and financial crimes, in: A. Rafay (Ed.), Handbook of Research On Theory and Practice of Financial Crimes, IGI Global, 2021, pp. 506–524, https://doi.org/10.4018/978-1-7998-5567-5.ch025.
[27] Y. Yao, J. Duan, K. Xu, Y. Cai, Z. Sun, Y. Zhang, A survey on large language model (LLM) security and privacy: the good, the bad, and the ugly, High-Confid. Comput. 4 (2) (2024) 100211, https://doi.org/10.1016/j.hcc.2024.100211.
[28] O. Semenenko, S. Kirsanov, A. Movchan, M. Ihnatiev, U. Dobrovolskyi, Impact of computer-integrated technologies on cybersecurity in the defence sector, Machinery 15 (2) (2024) 118–129, https://doi.org/10.31548/machinery/2.2024.118.
[29] G. Finocchiaro, The regulation of artificial intelligence, AI. Soc. 39 (2024) 1961–1968, https://doi.org/10.1007/s00146-023-01650-z.
[30] Junklewitz, H., Hamon, R., André, A., Evas, T., Soler Garrido, J., & Sanchez Martin, J.I. (2023). Cybersecurity of artificial intelligence in the AI Act. Publications Office of the European Union. doi:10.2760/271009.
[31] E. De Angelis, Generative artificial intelligence in defence and security: an introduction, RUSI J. 168 (7) (2023) 14–15, https://doi.org/10.1080/03071847.2023.2336745.
[32] I. Jada, T.O. Mayayise, The impact of artificial intelligence on organisational cyber security: an outcome of a systematic literature review, Data Inf. Manag. 8 (2) (2024) 100063, https://doi.org/10.1016/j.dim.2023.100063.
[33] IBM, Cost of a Data Breach Report 2024. IBM, Available Online (2024). https://www.ibm.com/reports/data-breach.
[34] V. Garg, J. Dev, Artificial Intelligence and the New Economics of Cyberattacks, LOGIN: Mag. USENIX Assoc. (2024). Available Online, https://www.usenix.org/publications/loginonline/artificial-intelligence-and-new-economics-cyberattacks.
[35] F. Bergadano, G. Giacinto (Eds.), AI For cybersecurity: Robust models For authentication, Threat and Anomaly Detection, MDPI, 2023, https://doi.org/10.3390/books978-3-0365-8265-8.
[36] Ganescu, B.-M., & Passerat-Palmbach, J. (2024). Trust the Process: zero-Knowledge Machine Learning to Enhance Trust in Generative AI Interactions. arXiv:2402.06414 [cs.LG]. doi:10.48550/arXiv.2402.06414.
[37] S. Moore, S. Brown, W. Butler, AI and social impact: a review of current use cases and broader implications, in: M. Dawson, O. Tabona, T. Maupong (Eds.), Cybersecurity Capabilities in Developing Nations and Its Impact On Global Security, IGI Global, 2022, pp. 133–161, https://doi.org/10.4018/978-1-7998-8693-8.ch008.
[38] S. Mishra, Exploring the impact of AI-based cyber security financial sector management, Appl. Sci. 13 (10) (2023) 5875, https://doi.org/10.3390/app13105875.
[39] M.S. Farahani, G. Ghasemi, Artificial intelligence and inequality: challenges and opportunities, Qeios (2024), https://doi.org/10.32388/7HWUZ2.
[40] R. González-Sendino, E. Serrano, J. Bajo, Mitigating bias in artificial intelligence: fair data generation via causal models for transparent and explainable decision-making, Future Gener. Comput. Syst. 155 (2024) 384–401, https://doi.org/10.1016/j.future.2024.02.023.
[41] A.Z. Huriye, The ethics of artificial intelligence: examining the ethical considerations surrounding the development and use of AI, Am. J. Technol. 2 (1) (2023) 37–44. Available Online, https://gprjournals.org/journals/index.php/AJT/article/view/142.
[42] D. Oberhaus, Authoritarian regimes' AI innovation advantage, Harv. Mag. (2022). Available Online, https://www.harvardmagazine.com/2022/04/right-now-authoritarian-regimes-artificial-intelligence.
[43] E. Zaidan, I.A. Ibrahim, AI governance in a complex and rapidly changing regulatory landscape: a global perspective, Humanit. Soc. Sci. Commun. 11 (1) (2024), https://doi.org/10.1057/s41599-024-03560-x. Article 1121.
[44] H. Roberts, E. Hine, M. Taddeo, L. Floridi, Global AI governance: barriers and pathways forward, Int. Aff. 100 (3) (2024) 1275–1286, https://doi.org/10.1093/ia/iiae073.
[45] X. Wang, X. Qiu, The positive effect of artificial intelligence technology transparency on digital endorsers: based on the theory of mind perception, J. Retail. Consum. Serv. 78 (2024) 103777, https://doi.org/10.1016/j.jretconser.2024.103777.
[46] L. Papadopoulos, K. Demestichas, E. Muñoz-Navarro, J.J. Hernández-Montesinos, S. Paul, N. Museux, S. König, S. Schauer, A. Climente Alarcón, I. Perez Llopis, T. Stelkens-Kobsch, T. Hadjina, J. Levak, Protection of critical infrastructures from advanced combined cyber and physical threats: the PRAETORIAN approach, Int. J. Crit. Infrastruct. Prot. 44 (2024) 100657, https://doi.org/10.1016/j.ijcip.2023.100657.
[47] CEDPO AI Working Group, Generative AI: the Data Protection Implications, CEDPO (2023). https://cedpo.eu/wp-content/uploads/generative-ai-the-data-protection-implications-16-10-2023.pdf.
[48] A. Golda, et al., Privacy and security concerns in generative AI: a comprehensive survey, IEEe Access. 12 (2024) 48126–48144, https://doi.org/10.1109/ACCESS.2024.3381611.
[49] The IEEE Global Initiative 2.0 on Ethics of Autonomous and Intelligent Systems. (2024). IEEE Standards Association. https://standards.ieee.org/industry-connections/activities/ieee-global-initiative/.
[50] European Parliament, Artificial intelligence act (2024, July 12), Eur. Parliam. (2024). https://artificialintelligenceact.eu/the-act/.
[51] N.I. Che Mat, N. Jamil, Y. Yusoff, M.L. Mat Kiah, Y. Yusoff, A systematic literature review on advanced persistent threat behaviors and its detection strategy, J. Cybersecur. 10 (1) (2024) tyad023, https://doi.org/10.1093/cybsec/tyad023.
[52] S. Calzavara, A. Rabitti, M. Bugliesi, Semantics-Based Analysis of Content Security Policy Deployment, ACM Trans. Web 12 (2) (2018) 1–36, https://doi.org/10.1145/3149408.
[53] P. Van Schaik, K. Renaud, C. Wilson, J. Jansen, J. Onibokun, Risk as affect: the affect heuristic in cybersecurity, Comput. Secur. 90 (2020) 101651, https://doi.org/10.1016/j.cose.2019.101651.
[54] M.H. Yun, I. Rhiu, W. Kim, Y. Lee, Y.M. Kim, AI in human behavior analysis, in: C. S. Nam, J.-Y. Jung, & S. Lee (Eds.), Human-centered Artificial Intelligence, Academic Press, 2022, pp. 191–204, https://doi.org/10.1016/B978-0-323-85648-5.00010-4.
[55] D. Samariya, A. Thakkar, A comprehensive survey of anomaly detection algorithms, Ann. Data Sci. 10 (4) (2023) 829–850, https://doi.org/10.1007/s40745-021-00362-9.
[56] D. Turley, R. Ambhore, The impact of AI on data security, AI J. (2024). Available Online, https://aijourn.com/the-impact-of-ai-on-data-security/.
[57] W. Serrano, CyberAIBot: artificial intelligence in an intrusion detection system for cybersecurity in the IoT, Future Gener. Comput. Syst. (2024) 107543, https://doi.org/10.1016/j.future.2024.107543.
[58] D. Olszewski, M. Iwanowski, W. Graniszewski, Dimensionality reduction for detection of anomalies in the IoT traffic data, Future Gener. Comput. Syst. 151 (2024) 137–151, https://doi.org/10.1016/j.future.2023.09.033.
[59] S.K. Garapati, A.N. Sigappi, An artificial intelligence-based intrusion detection system using optimization and deep learning, J. Electr. Syst. 20 (6s) (2024), https://doi.org/10.52783/jes.2850.
[60] V.K. Tiwari, R. Dwivedi, Analysis of cyber attack vectors, in: 2016 International Conference on Computing, Communication and Automation (ICCCA), 2016, pp. 600–604, https://doi.org/10.1109/CCAA.2016.7813791.
[61] Chong, C.J., Hou, C., Yao, Z., & Seyed Talebi, S.M. (2024). Casper: prompt sanitization for protecting user privacy in web-based large language models. arXiv [Computer Science]. doi:10.48550/arXiv.2408.07004.
[62] P. Venkatachalam, S. Ray, How do context-aware artificial intelligence algorithms used in fitness recommender systems? A literature review and research agenda, Int. J. Inf. Manag. Data Insights 2 (2) (2022) 100139, https://doi.org/10.1016/j.jjimei.2022.100139.

[63] R. Mudarova, D. Namiot, Countering prompt injection attacks on large language models, Int. J. Open Inf. Technol. 12 (5) (2024). . ISSN: 2307-8162. IT Congress 2024. Available Online, http://injoit.org/index.php/j1/article/view/1845.
[64] N. Moustafa, A new distributed architecture for evaluating AI-based security systems at the edge: network TON_IoT datasets, Sustain. Cities Soc. 72 (2021) 102994, https://doi.org/10.1016/j.scs.2021.102994.
[65] R. Dabbous, Why We Should Design Less Privileged AI Systems, AI & Society, 2024, https://doi.org/10.1007/s00146-023-01853-4.
[66] T. Geppert, S. Deml, D. Sturzenegger, N. Ebert, Trusted execution environments: applications and organizational challenges, Front. Comput. Sci. 4 (2022) 930741, https://doi.org/10.3389/fcomp.2022.930741.
[67] C.K.Y. Chan, W. Hu, Students' voices on generative AI: perceptions, benefits, and challenges in higher education, Int. J. Educ. Technol. High. Educ. 20 (1) (2023) 43, https://doi.org/10.1186/s41239-023-00411-8.
[68] Akinsola, J.E.T., Akinseinde, S., Kalesanwo, O., Adeagbo, M., Oladapo, K., Awoseyi, A., & Kasali, F. (2021). Application of Artificial Intelligence in User Interfaces Design for Cyber Security Threat Modeling. Software Usability. doi:10.5772/intechopen.96534.
[69] N. Balasubramaniam, M. Kauppinen, A. Rannisto, K. Hiekkanen, S. Kujala, Transparency and explainability of AI systems: from ethical guidelines to requirements, Inf. Softw. Technol. 159 (2023) 107197, https://doi.org/10.1016/j.infsof.2023.107197.
[70] K. Gnitko, Systematic overview of AI security standards, SSRN. (2024), https://doi.org/10.2139/ssrn.4922592.
[71] Faveri, B., & Auld, G. (2023). Informing possible futures for the use of third-party audits in AI regulations. Carleton University, School of Public Policy and Administration. doi:10.22215/sppa-rgi-nov2023.
[72] I.D. Raji, P. Xu, C. Honigsberg, D. Ho, Outsider oversight: designing a third party audit ecosystem for AI governance, in: Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society, 2022, pp. 557–571, https://doi.org/10.1145/3514094.3534181.
[73] M. Anisetti, C.A. Ardagna, E. Damiani, N. El Ioini, A Journey Into Security certification: From the Cloud to Artificial Intelligence, Springer, 2024, https://doi.org/10.1007/978-3-031-59724-4.