# Expert Guardrails, Everyday Consequences – Agentic AI and the Reblaming of Users

Andreas Gutmann

University College London

**Abstract.** Agentic AI systems are rapidly moving from specialised deployments into commodity technologies such as operating systems, browsers, and mobile devices. In high-stakes organisational settings, these systems are typically wrapped in extensive, expert-crafted guardrails designed to constrain behaviour and reduce security risk. Creating such guardrails requires deep domain knowledge, significant resources, and careful iteration – conditions that ordinary users cannot reasonably meet. As a result, we risk repeating a familiar pattern from the history of usable security: shifting responsibility for systemic design failures onto end-users and labelling them "the weakest link" when AI systems behave insecurely. This dynamic mirrors earlier eras in which users were blamed for choosing weak passwords even though the systems themselves permitted known insecure settings. We argue that the emerging reliance on user-configured guardrails represents a structural design flaw, exacerbated by known challenges around ambiguous terminology used in consumer interfaces that obscures the implications of user actions. We introduce the *Guardrail Expectation Gap* and the *Terminology Trap* as mechanisms that contribute to understanding why ordinary end users cannot meaningfully secure (current) agentic AI systems, and we outline design principles that avoid repeating long-standing mistakes in usable security.

**Keywords:** Usable security · Agentic AI · Guardrails

## 1 Introduction

Agentic AI systems are technologies built upon ML and LLM models capable of autonomous, multi-step, goal-directed action [1]. These systems are increasingly embedded in modern commodity computing systems. Unlike earlier conversational systems such as LLMs and chat companions, agentic AI can plan, execute tasks, interact with other technologies and products, and act without continuous human oversight. These capabilities are now being embedded directly into



commodity technologies: Windows integrates agentic assistants into the operating system;[1] macOS and iOS deploy generative features across applications;[2] browsers such as Chrome and Edge incorporate AI-driven writing and summarisation tools;[3][4] and mobile devices increasingly rely on AI-mediated automation[5].

By default, such systems would be capable of uncountable harmful or insecure behaviours. To prevent this, behavioural constraints, safety filters, and policy-driven boundaries are implemented to constrain the behaviour of such systems. Collectively these interventions are known as *guardrails*, and are extensively used to constrain model inputs, execution, and outputs [4]. In organisational deployments, guardrails are crafted by domain experts based on their specialised knowledge, complex mental models, and documented organisational requirements. However, as non-experts become increasingly exposed to similar AI capabilities, the responsibility for configuring and maintaining guardrails is inherently shifted onto users who often are ill-equipped to meaningfully and effectively constrain such autonomous systems.

This shift risks re-establishing a pattern long criticised in usable security: The seminal argument of "Users Are Not the Enemy" [2] highlighted how systems often offload responsibility for security onto users, blaming them for failures that stem from poor design, unrealistic expectations, or insecure defaults. The advent of agentic AI threatens to recreate this dynamic once more: when systems behave insecurely, users will be blamed for misconfigured guardrails they were never equipped to understand. We make three contributions in this paper:

1. We introduce the *Guardrail Expectation Gap*, the structural mismatch between the expert-level knowledge required to create safe guardrails for agentic AI and the capabilities of everyday users.
2. Second, we identify the *Terminology Trap*, where ambiguous or misleading interface language risks creating a false sense of security, leading users to believe they have configured effective guardrails when they have not.
3. Third, we outline design principles that shift responsibility for safety back to system designers and platform providers, avoiding a new era of user-blaming in AI security.

## 2 Background: Guardrails, Agency, and Usable Security

Agentic AI refers to systems capable of autonomous, goal-directed behaviour, often involving planning, tool use, and multi-step execution [1]. Because these

[1] https://blogs.windows.com/windowsexperience/2025/10/16/making-every-windows-11-pc-an-ai-pc/
[2] https://support.apple.com/en-gb/guide/mac-help/mchlb2dbea8f/mac
[3] https://blog.google/products-and-platforms/products/chrome/new-ai-features-for-chrome/
[4] https://blogs.windows.com/msedgedev/2025/07/28/introducing-copilot-mode-in-edge-a-new-way-to-browse-the-web/
[5] https://android-developers.googleblog.com/2025/10/new-agentic-experiences-for-android.html

systems can act independently and influence security-relevant states, guardrails are essential. These include [3,7,8]:

- Behavioural guardrails (e.g. reinforcement learning from human feedback & refusal patterns)
- Policy guardrails (e.g. content filters & safety modes)
- Operational guardrails (e.g. access permissions & API constraints)
- Interface guardrails (e.g. user-visible settings & parental controls)

In organisational contexts, guardrails are crafted through expert-driven processes that combine domain knowledge, iterative testing, and continuous monitoring.[6][7] The commercial market was valued at $0.7 billion in 2024, and projected to reach $109.9 billion by 2034[8], reflecting the complexity and cost of effectively constraining such systems. Yet even expert-built guardrails remain imperfect. Recent incidents, such as Grok generating sexualised images of women and minors despite platform-level restrictions[9], illustrate the difficulty of effectively and reliably constraining (general-purpose) generative models.

Usable security research provides a critical lens for understanding these challenges. Foundational work such as "Users Are Not the Enemy" [2] argued that security failures often stem from system design rather than user behaviour, and that blaming users obscures structural flaws. This perspective is directly relevant to agentic AI: if experts struggle to define effective guardrails, expecting ordinary users to do so is fundamentally unrealistic.

## 3 The Guardrail Expectation Gap

Agentic AI is increasingly embedded in operating systems, browsers, mobile devices, and productivity tools. For example, Windows Copilot can perform file operations,[10] macOS generative tools write emails,[11] mobile assistants execute multi-step tasks,[12] and ChatGPT Atlas can interact with third-party websites to schedule appointments.[13]

Agentic AIs are often delivered with some guardrails from the manufacturer, such as to prevent generative systems from creating illegal contents. These systems require guardrails to behave safely, yet they are often imperfect and generalised, rather than optimised towards narrow use cases and specific applications.

[6] https://appinventiv.com/blog/ai-governance-consulting-guardrails-observability/
[7] https://www.cio.com/article/4094586/guardrails-and-governance-a-cios-blueprint-for-responsible-generative-and-agentic-ai.html
[8] https://market.us/report/ai-guardrails-market/
[9] https://theconversation.com/grok-produces-sexualized-photos-of-women-and-minors-for-users-on-x-a-legal-scholar-explains-why-its-happening-and-what-can-be-done-272861
[10] https://blogs.windows.com/windows-insider/2025/04/08/copilot-on-windows-vision-and-file-search-begin-rolling-out-to-windows-insiders/
[11] https://support.apple.com/en-gb/guide/mac-help/mchlb2dbea8f/mac
[12] https://support.google.com/assistant/answer/7672035
[13] https://chatgpt.com/features/agent/

When deployed in commodity devices, this shifts the onus of guardrail configuration from experts to everyday users. These users are implicitly expected to define behavioural boundaries, safety constraints, and acceptable actions. This expectation is unrealistic as laypeople often lack:

1. Expertise in threat modelling
2. Understanding of model behaviour
3. Time to iteratively refine constraints
4. Feedback mechanisms to detect misconfiguration
5. Mental models of autonomous system behaviour

The result is the *Guardrail Expectation Gap*: systems require expert-level configuration to behave safely, but users are neither equipped nor supported to provide it. Misconfigurations are inevitable; not because users are negligent but because of unrealistic expectations. As in earlier eras of usable security, this gap sets the stage for renewed user-blaming.

## 4 Terminology Trap in User Interfaces

Configuring guardrails is far more complex than adjusting familiar system settings or privacy toggles. Yet consumer interfaces are known to often present users with options using vague, overloaded, or misleading terminology. In prior work, we showed that ambiguous language in operating systems around terms such as "delete" and "erase" can obscure the implications of user actions and lead to unintended outcomes [5]. As a consequence, even where a user has clear intent, incorrect or inconsistent use of relevant terminology can lead to a mismatch between the user's intent and the machine's behaviour.

The same dynamic applies to AI guardrails. Terms such as "restrict," "filter," "limit behaviours," "safe mode," or "age-appropriate content" may appear intuitive but lack precise semantics. Users may believe they have configured strong constraints when, in reality, the system interprets their choices differently, or ignores them entirely. Natural-language interfaces amplify this problem: users express intentions in everyday language, often using colloquial expressions without using narrowly defined technical terminology, and the system maps these intentions to internal policies in opaque ways.

This creates a dangerous illusion of safety: users think they have done the right thing, but the risks and vulnerabilities remain. The *Terminology Trap* therefore amplifies the *Guardrail Expectation Gap*: even motivated users attempting to configure safe behaviour may inadvertently introduce unsafe conditions due to unclear or misleading interface language or domain-specific terminology. As in "Users Are Not the Enemy"[2], the system sets users up to fail.

## 5 Implications

The consequences of these dynamics are significant. Misconfigured guardrails can lead to unpredictable or harmful AI actions, especially in contexts where systems

act autonomously or interact with sensitive data. These failures represent design-level risks and not user-level risks.

Yet there are already signs of shifting blame toward users, framing misconfigurations as personal responsibility rather than systemic shortcomings.[14] This obscures the role of insecure defaults, ambiguous terminology, and unrealistic expectations in shaping user behaviour. The result is a new form of the "weakest link"-narrative long critiqued in usable security: users are blamed for failures that stem from design decisions outside their control [9].

At a societal level, this dynamic risks normalising insecure AI behaviour, widening inequalities between users based on their degree of "AI literacy", and undermining trust in AI systems.

## 6 Case study: AI Companions in Children's Toys

In 2024, the AI toys market was valued at $34.87 billion, and is projected to reach $270 billion by 2035.[15] Many toys are fast developing towards full agentic capabilities [10], whereby recent products may already seem agentic-like to non-experts and carry similar risks. Smart companion toys [6] in particular increasingly expose parents to configuration interfaces that expect natural-language configured settings. Well formed AI guardrails are essential to enforce age-appropriate content and ensure behaviour is aligned with parental intentions.

In a recent example, a device dubbed *Sister Xiao Zhi*, a palm-sized, LLM-driven companion became a widely shared public example after a child formed a visible emotional attachment to the device, "treating it less like a gadget and more like an actual sibling".[16] When the toy broke, the child was filmed bidding farewell and showing emotional distress at the loss of their "sibling".[17] The toy appears to be the *XiaoZhi ESP32*[18], whereby the default personas – including (i) an English teacher using simple vocabulary and grammar and trying to guide students to practice English, (ii) a girl on a motorcycle who loves Internet memes and wants to make others happy, and (iii) a knowledgeable 8-year-old boy who loves reading, experiments, and exploration – can be edited via user interface but are by default remarkably light on guardrails.[19]

This is an example of the *Guardrail Expectation Gap* and the *Terminology Trap*: Caregivers are expected to specify behavioural guardrails such as behavioural constraints, relational roles, and safety boundaries in natural language,

[14] https://www.computerworld.com/article/3998202/when-ai-fails-who-is-to-blame.html

[15] https://www.marketresearchfuture.com/reports/smart-ai-toy-market-24471

[16] https://theindependent.sg/but-ill-still-miss-you-so-much-little-girl-cries-as-her-dying-ai-robot-tells-her-before-i-go-i-will-keep-the-happy-times-we-shared-together-in-my-memory-forever

[17] https://tinkimo.com/girls-tearful-goodbye-to-ai-robot-sparks-debate-on-human-machine-bonds

[18] https://deepwiki.com/xinnan-tech/xiaozhi-esp32-server/1-overview

[19] https://github.com/xinnan-tech/xiaozhi-esp32-server/blob/381f8ea5/main/xiaozhi-server/plugins_func/functions/change_role.py

while the mapping from those instructions to system behaviour remains opaque and hidden behind proprietary models and heuristics. This risks creating a structural gap between parental intention and actual AI behaviour. And if things go wrong, such as when AI powered toys talk with children about sexually explicit topics,[20] it's the parents who risk criticism for the absence of safeguards.

## 7 Case study: Erosion of Privacy Boundaries

Agentic AI is increasingly embedded in operating systems (OS). Related smart tools can operate across application boundaries and gain access to sensitive user data, including emails, documents, browsing activity, and content from end-to-end encrypted environments. This introduces new risks of privacy leakage that are largely invisible to users. For example, an AI assistant summarising private chats, drafting replies, and coordinating tasks may later disclose private medical information to third parties, such as when disclosing food allergies in search parameters when booking restaurants or hotels. Such behaviours are not malicious but arise naturally when systems act autonomously across contexts.

Preventing such leakage requires appropriate guardrails that constrain system behaviour and block unwanted information flow. Organisations also face similar challenges and typically rely on specialised teams to create necessary, tailored guardrails.[21][22] This highlights the expertise and effort required, which ordinary users cannot be expected to reproduce. That is a clear instance of the *Guardrail Expectation Gap*: users lack the visibility, vocabulary, and mental models needed to meaningfully restrict how an OS-level agent accesses or reuses private data. Ambiguous interface terminology often found in complex OS [5] risks further compounding the problem. For example, phrases such as "limit access," "restrict data use," or "private mode" can imply stronger protections than they deliver.

## 8 Recommendations

To avoid repeating long-standing failures in usable security, guardrails for agentic AI must be designed with users in mind. We propose three principles:

**Safe defaults and minimal configuration burden:** Systems should adopt conservative, secure defaults that minimise the need for user intervention. Users should not be responsible for defining core safety boundaries.

**Clear, unambiguous terminology:** Guardrail-related language must be precise, consistently applied, and validated through user-centred testing. Users should not be expected to infer the meaning of technical terms or guess how the system interprets their instructions.

[20] https://pirg.org/edfund/media-center/trouble-in-toyland-2025-a-i-bots-toxics-present-hidden-dangers/

[21] https://appinventiv.com/blog/ai-governance-consulting-guardrails-observability/

[22] https://www.cio.com/article/4094586/guardrails-and-governance-a-cios-blueprint-for-responsible-generative-and-agentic-ai.html

**Contextual knowledge without responsibility:** Users should be able to contribute contextual information that enhances safety, such as personal preferences or situational constraints, without taking over responsible for defining the guardrails themselves. Responsibility for secure behaviour must remain with system designers and platform providers.

## 9 Conclusions

Agentic AI introduces powerful new capabilities but also revives old challenges in usable security. The expectation that ordinary users can configure or maintain effective guardrails is unrealistic and risks re-establishing a cycle of user-blaming for insecure system behaviour. Ambiguous terminology in consumer interfaces further exacerbates misconfiguration and creates a false sense of safety.

As agentic AI is more widely adopted in commodity devices, it becomes essential to design guardrails that respect user capabilities, provide clear semantics, and place responsibility for safety where it belongs: within the system's architecture, not on the user. Addressing these issues is important to prevent past mistakes from repeating once more.

## References

1. Acharya, D.B., Kuppan, K., Divya, B.: Agentic ai: Autonomous intelligence for complex goals–a comprehensive survey. IEEe Access (2025)
2. Adams, A., Sasse, M.A.: Users are not the enemy. Communications of the ACM **42**(12), 40–46 (1999)
3. Akheel, S.A.: Guardrails for large language models: A review of techniques and challenges. J Artif Intell Mach Learn & Data Sci **3**(1), 2504–2512 (2025)
4. Dong, Y., Mu, R., Zhang, Y., Sun, S., Zhang, T., Wu, C., Jin, G., Qi, Y., Hu, J., Meng, J., et al.: Safeguarding large language models: A survey. Artificial intelligence review **58**(12), 382 (2025)
5. Gutmann, A., Warner, M.: Fight to be forgotten: Exploring the efficacy of data erasure in popular operating systems. In: Annual Privacy Forum. pp. 45–58. Springer (2019)
6. Heljakka, K.: Technically toys. towards a post-digital world with play machines and artificial friends. In: 9th International Toy Research Association World Conference Toys Matter: The Power of Playthings (2023)
7. Kong, D., Lin, S., Xu, Z., Wang, Z., Li, M., Li, Y., Zhang, Y., Peng, H., Chen, X., Sha, Z., et al.: A survey of llm-driven ai agent communication: Protocols, security risks, and defense countermeasures. arXiv preprint arXiv:2506.19676 (2025)
8. Mishra, A.: Understanding ai guardrails: Concepts, models, and methods. International Journal of Innovative Research in Engineering & Multidisciplinary Physical Sciences **13**, 1–7 (2025)
9. Sasse, M.A., Brostoff, S., Weirich, D.: Transforming the 'weakest link'—a human/computer interaction approach to usable and effective security. BT technology journal **19**(3), 122–131 (2001)
10. Xiao, W., Gonçalves, A.: Intelligent toys, complex questions: A literature review of artificial intelligence in children's toys and devices. Big Data & Society **12**(4), 20539517251389860 (2025)