Securing LLM usage

Speakers

Sandesh

Co-founder, Seezo

Author, Boring AppSec

https://boringappsec.substack.com/

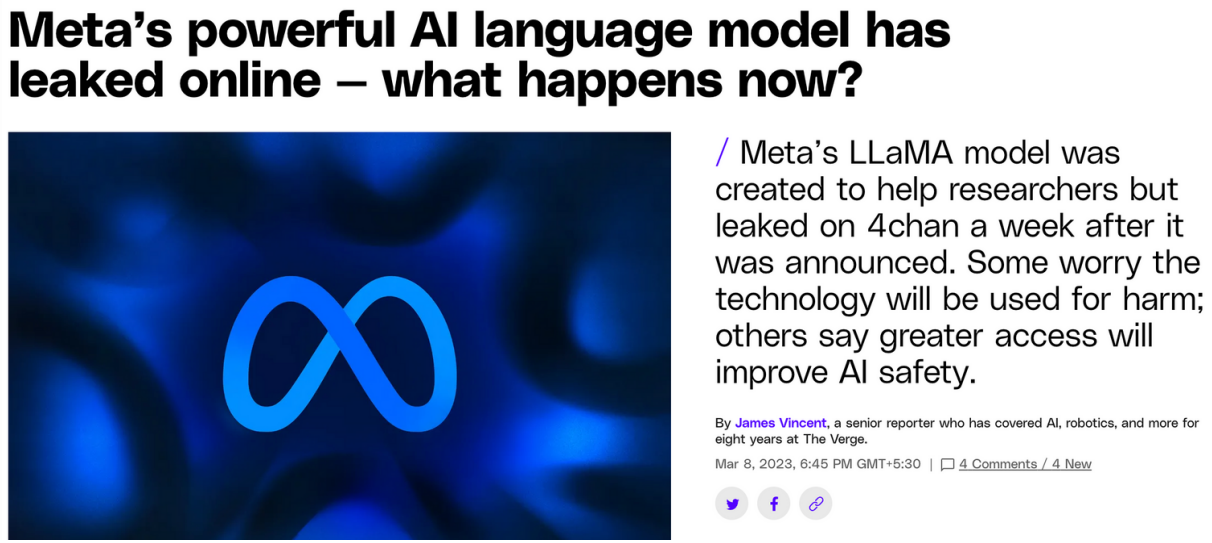
Ankush

Co-founder, Akto

https://www.akto.io/

Is securing LLMs necessary?

Is securing LLMs necessary?



Samsung Bans Staff's AI Use After Spotting ChatGPT Data Leak

- Employees accidentally leaked sensitive data via ChatGPT
- Company preparing own internal artificial intelligence tools

Securing AI systems is different

PC Ruchir Patwa, Syde Labs

Traditional Systems	AI Systems
Deterministic outputs	Probabilistic outputs
Pattern based security	Intent based security
Build time and Design time security gives great results	Build time and Design time security cannot be relied upon
High level of observability and traceability	Very little observability and traceability
Predictable outcomes from changes in systems (new code/features)	Unpredictable outcomes from changes in systems (new model)

Top security risks by use-case

Security risks by use case

Top risks of using 3rd party LLMs

Leveraging commercial offerings such as OpenAI or Google Vertex

1. **Prompt injection**
2. **Bias, Hallucination & over-reliance**
3. Sensitive data leakage
4. Cost overrun
5. **Lack of regulatory clarity**

Security risks by use case

Top risks of using an in-house LLM

Open source model trained with internal data

1. Cross-border data governance
2. Training data poisoning
3. **Bias, Hallucination & over-reliance**
4. **Lack of regulatory clarity**
5. **Prompt injection**

Managing LLM security risks

Risk management technique - use public LLM

Scanning input & output to your LLM

- [input] Check for prompt injection
- [input] Check authorization
  Does user have access to the data they are requesting?
- [output] block toxic/abusive/malicious output
- [input][output] Regex scan to detect sensitive data

Managing LLM security risks

If you are an operator (CISO, VP Engineering)

Monitor LLM usage

Build an inventory of LLM usage and detect insecure usage before the attackers do.

Use an LLM gateway

Validate prompts, estimate cost and thwart biases.

Simulate attacks

All LLM apps must be routinely tested.
Mimic attacker to test resilience against common attacks (use tools like Akto)

Threat model your apps

Identify key risks and controls for specific apps

If you are an AI startup

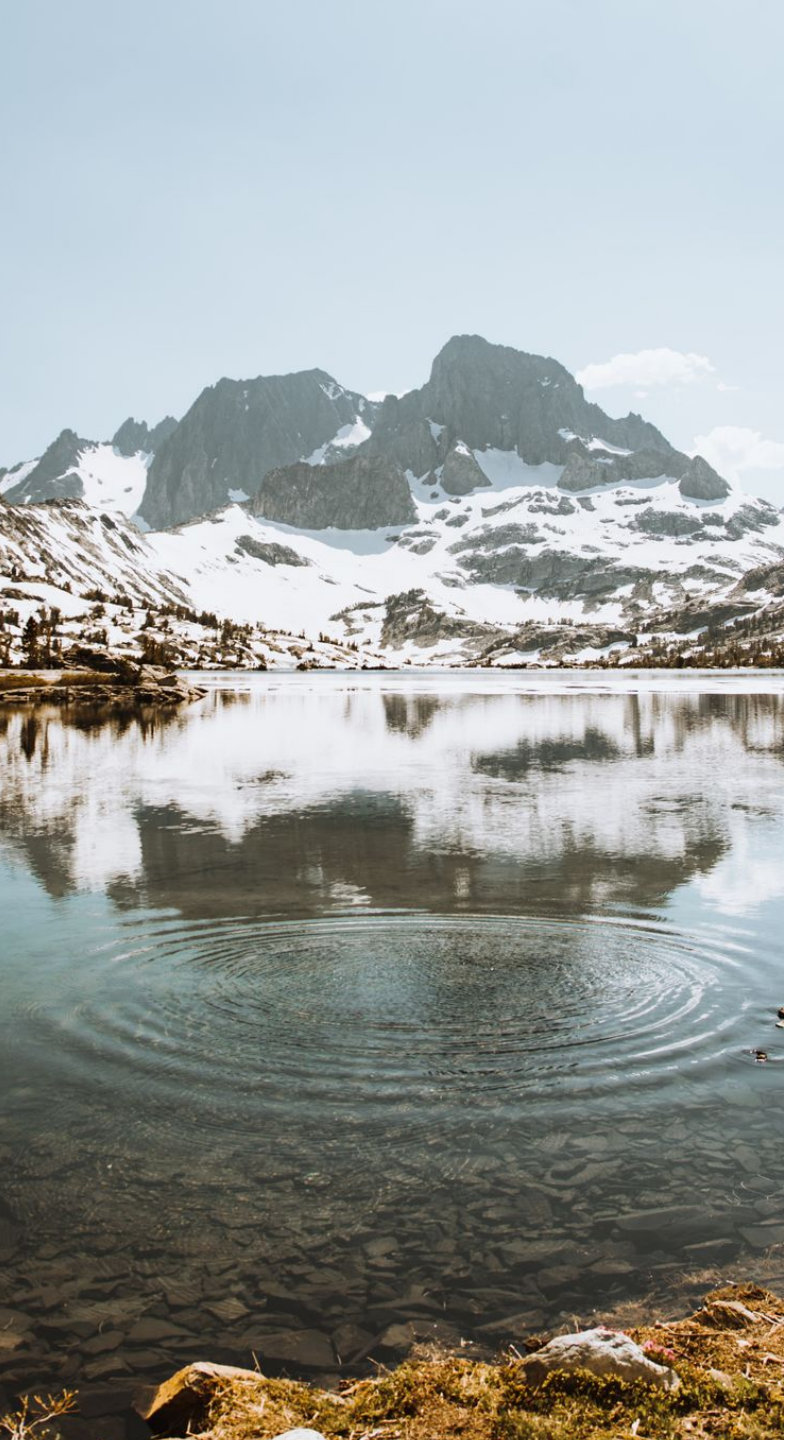
In addition to doing everything that operators do, you need to also manage perception

Think about sensitive data leakage right away!

Consider also providing a self-hosted solution (e.g. BYOK Open AI solution)

Prepare to engage CISOs in your target companies early

CISOs will default to what they are familiar (SOC2, HIPAA etc.)



In summary

- ✔ **Unique use cases require unique risk management.**
  Risks of using a chatbot with RAG is very different from an LLM agent interacting with critical DB. Prioritize what's important for you

- ✔ **Thinks are changing quickly. Keep up!**
  Resources like the OWASP Top 10 for LLMs and the OWASP AI Security & Privacy guide can help.

- ✖ **Don't leave Security to the end.**

References

- Prompt injection playground by Lakera: https://gandalf.lakera.ai/
- The OWASP Top 10 for LLMs: https://owasp.org/www-project-top-10-for-large-language-model-applications/
- Overview of LLM risks: https://boringappsec.substack.com/p/edition-21-a-framework-to-securely)
- Prompt inject in a PDF: https://kai-greshake.de/posts/inject-my-pdf/
- NVidia's security software (nemo Guardrails) leaks sensitive data: https://tech.slashdot.org/story/23/06/09/2033210/nvidias-ai-software-tricked-into-leaking-data