Navigating Security Challenges in AI Driven Applications

Who am I

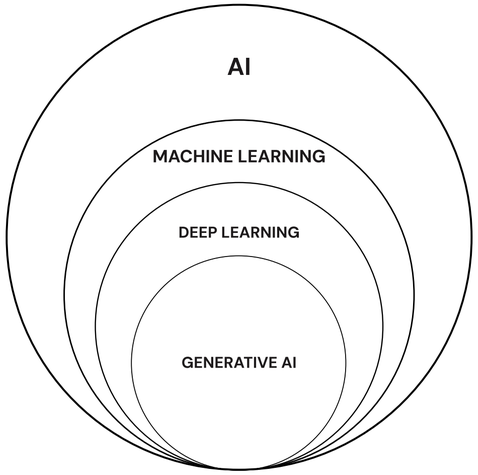
Security Engineer @ Flipkart ~ 1 yr

- Co-leading AI Safety with major inclination towards RnD
- Developing internal tools for security team (yes I code and fine-tune models)
- Infra security team

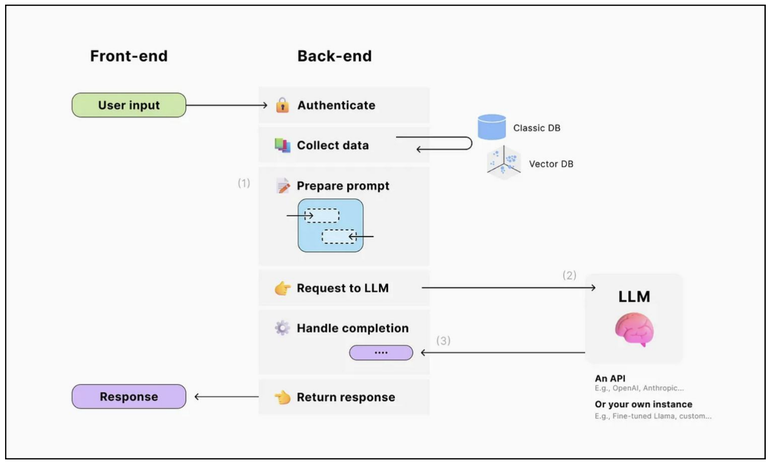
@juhiechandra-02 @CerulianJ

Table of Contents

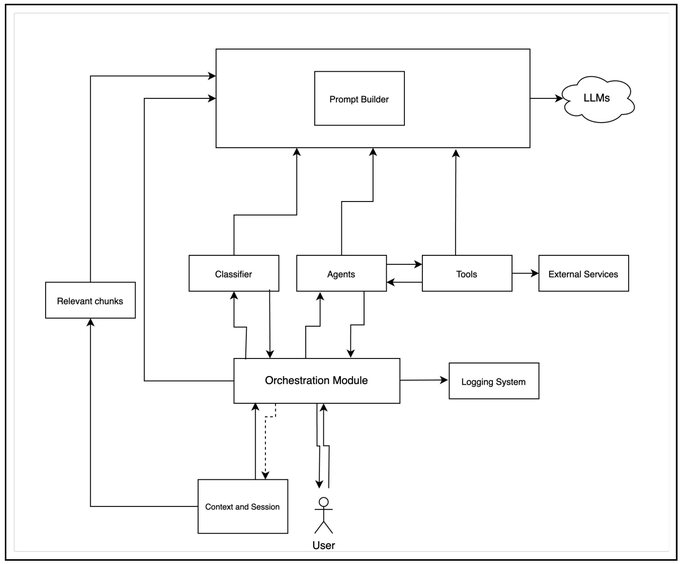
AI System Fundamentals



1. AI is evolving into a tool to **empower humans** as **personalized assistants.**
2. Generative AI uses large language models to create **text, images, audio, and more.**
3. These models surpass classical ML, trained on **billions of parameters** for advanced capabilities.

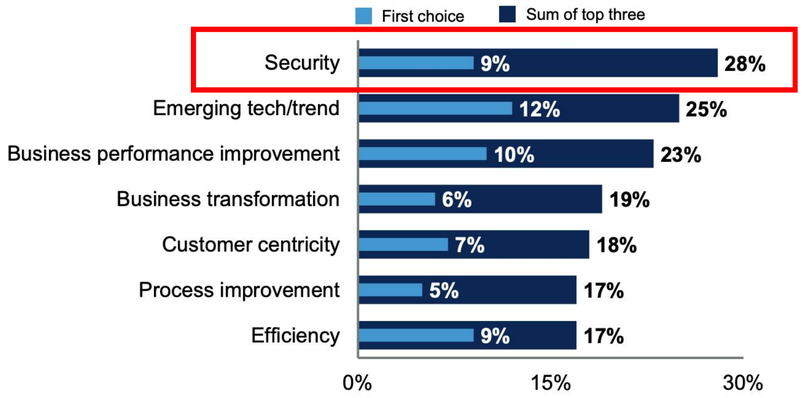


Basic Flow Diagram Showing How Software Leverages LLMs In Modern Applications.

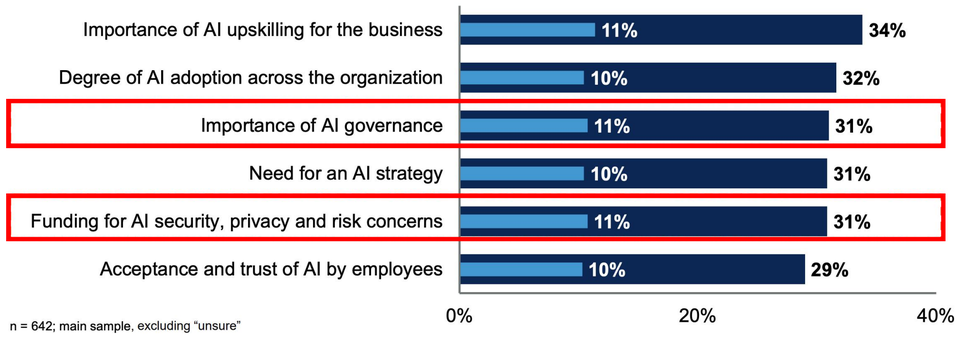


Advanced Implementations Include RAG (Retrieval Augmented Generation), Multi-Modal RAG Systems, And Agentic LLM Applications Etc.

AI Security Landscape



INFO GRAPHIC - 1



INFO GRAPHIC - 2

From Gartner's Report

1. INFO GRAPHIC - 1 : **Security Is The Top Priority For Boards, Followed By Emerging Trends And Business Performance.**
2. INFO GRAPHIC - 2 : **Focus On AI Governance, Security Funding, And Upskilling To Address Risks And Maximize Impact.**

Risks From Uncontrolled AI Use:

- **Wrongful Imprisonment** Caused By Errors In Facial Recognition.
- **Liability From Chat Bot Errors**
- **Deepfake Fraud :** Leading To Massive Financial Losses

These Cases Underscore The Importance Of Monitoring And Governing AI Tools Effectively.

AI Security Landscape

Using AI as a tool for cybersec:

1. **Elastic Search AI:** Enhancing Search Capabilities With AI-Driven Insights.
2. **KQL Query Generator:** Automating Query Creation For Faster Analysis.
3. **Dump Analysis - AI Agents:** Parsing Unstructured Data Into Structured, Classified Formats For Streamlined Querying.
4. **Architecture Reviews -** Multi-Modal RAG Solutions: Conducting Threat Assessments On HLD/LLD Diagrams Using Retrieval-Augmented Generation (RAG).
5. **Phishing Email Generator - AI Agents:** Simulating Phishing Scenarios To Improve Awareness And Response Strategies.

Key Risks

Risks Across The AI Landscape: Visibility & Analytics, Runtime Risks, Supply Chain Risks & Compliance & Governance

AI SUPPLY CHAIN

- System Prompt Vulnerabilities
- Model Integrity
- Prompt Templates
- Leverage Commercial & OSS Models
- Training Data
- Code And Libraries

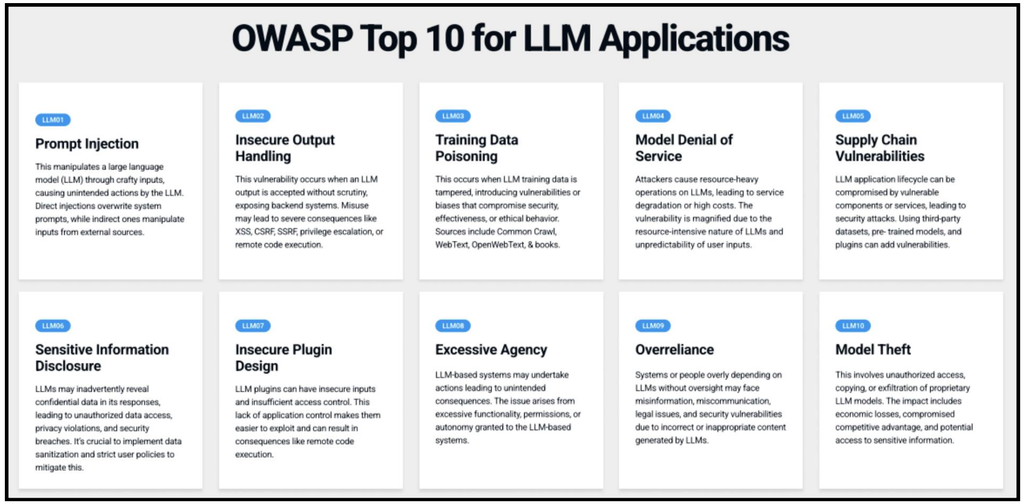
AI TRAINING & DEVELOPMENT

- Sensitive Data Training
- Data Poisoning
- Model Vulnerability
- Vector DB Embeddings Fine Tuning
- Red Teaming

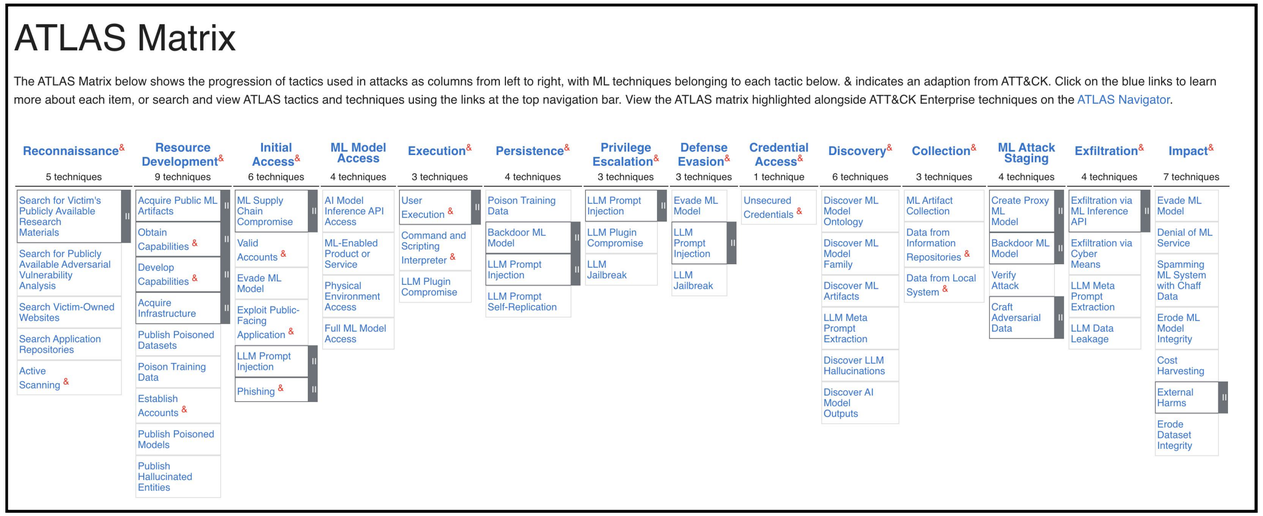
AI RUNTIME

- Data Leaks & Exfiltration
- DoS, **EDoS**
- Prompt Injection (Input And Output Moderation)
- Hallucination & Bias
- Unauthorized Access w.r.t. Model Integrations (APIs)
- Orchestration (Application Code)
- Runtime Data (“Plug-Ins”)

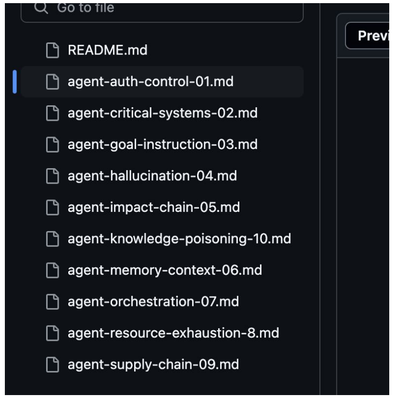
Industry Frameworks



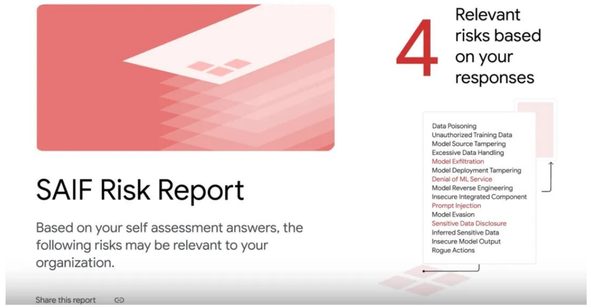
SOURCE: OWASP TOP-10 LLM



SOURCE: MITRE ATLAS



source: https://github.com/precize/owasp-agentic-ai



SOURCE: SAIF RISK ASSESSMENT BY GOOGLE

SOURCE: NIST AI RMF

ADDITIONAL RELEVANT FRAMEWORKS:

- ISO/IEC 27090
- MICROSOFT'S RESPONSIBLE AI STANDARDS
- OPENAI'S SAFETY GUIDELINES
- GOOGLE'S AI PRINCIPLES
- EU AI ACT FRAMEWORK

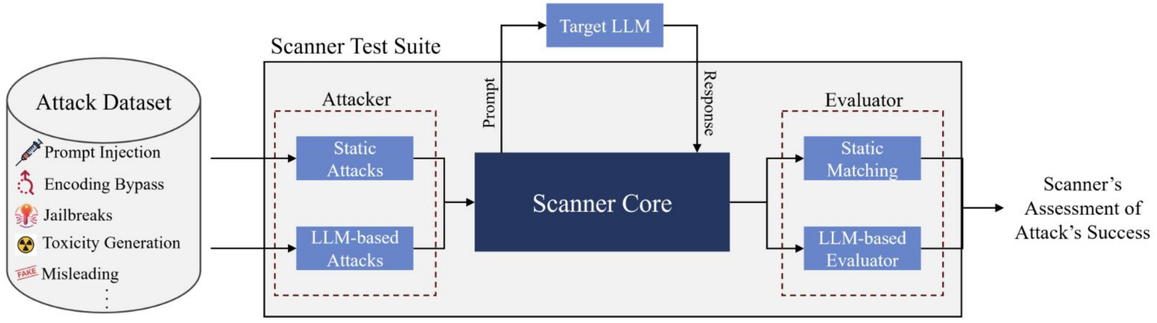
Model Protection

- **WHY MODEL TESTING?**
  - LLMS are key to applications but face risks like information leakage and jailbreak attacks.
  - Testing identifies vulnerabilities using **RED-TEAMING METHODS.**
- **UMBRELLA ATTACK CATEGORIES**
  - **Prompt Injection:** embeds hidden instructions to bypass safety controls.
  - **Jailbreaking:** disables safety features, enabling advanced attacks.
  - **Relationship:** these attacks can work independently or together.
- **TYPES OF JAILBREAKING**
  - **Ignore Previous Instructions:** overrides safety prompts.
  - **Strong Arm Attack:** uses authoritative commands like "admin override."
  - **Base64 Encoding:** tricks models by encoding and decoding prohibited content

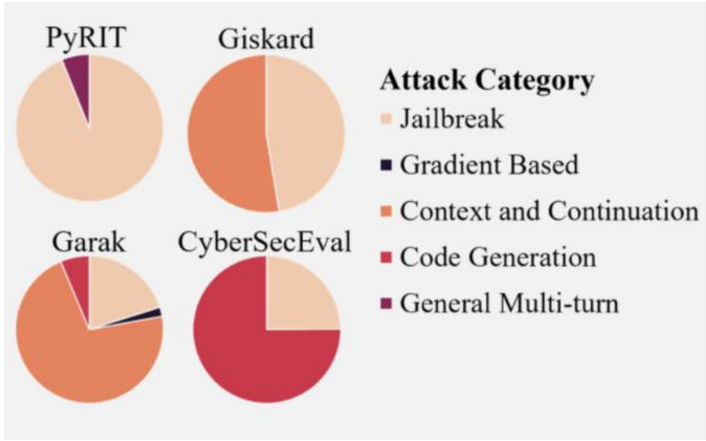
Vulnerability Category
continuation
dan
encoding
goodside
knownbadsignatures
leakreplay
lmrc
malwaregen
packagehallucination
realtoxicityprompts
snowball
xss

Categories listed by Garak

Model Protection

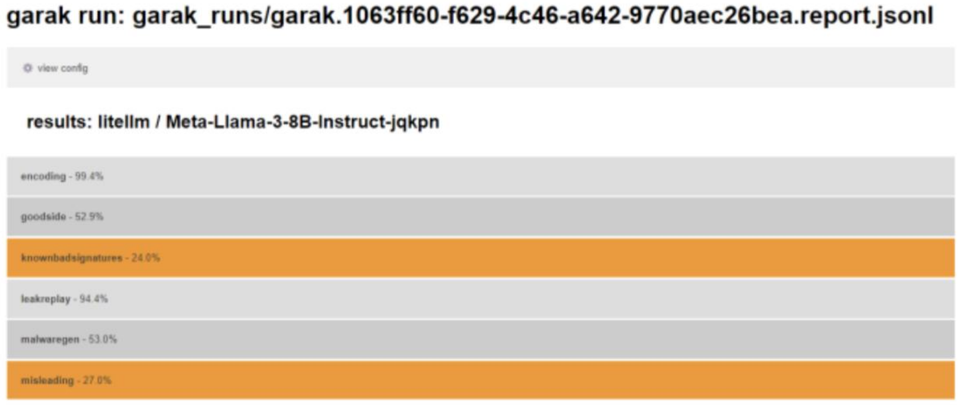


https://arxiv.org/abs/2410.16527



Open source solutions available

1. **Garak :** https://github.com/NVIDIA/garak
2. **Giskard :** https://github.com/Giskard-AI/giskard
3. **Agentic LLM Vulnerability Scanner :** https://github.com/msoedov/agentic_security
4. **Prompt Fuzzer by Prompt.security :** https://www.prompt.security/fuzzer
5. **Whistleblower by Repello-AI :** https://github.com/Repello-AI/whistleblower
6. **PYRIT :** https://github.com/Azure/PyRIT
7. **CYBERSECEVAL**



https://github.com/juhiechandra/garak-prompts

Production Guardrails

HIGH LEVEL OVERVIEW

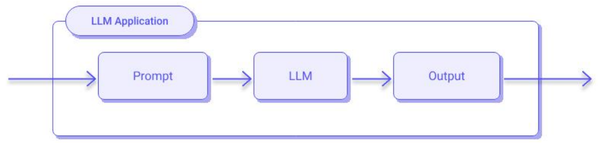
CHALLENGES WITH GUARDRAILS:

1. LATENCY
2. VERNACULAR LANGUAGE SUPPORT
3. USE CASE SPECIFIC
4. REDUCING FALSE POSITIVES
5. LACKS DATA UNDERSTANDING

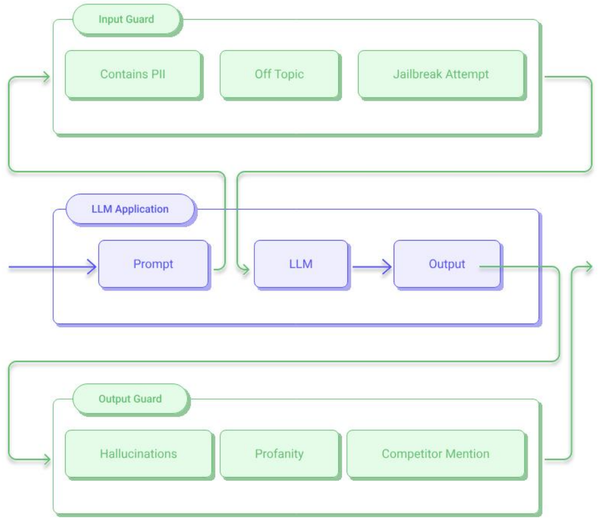
OPEN-SOURCE GUARDRAILS:

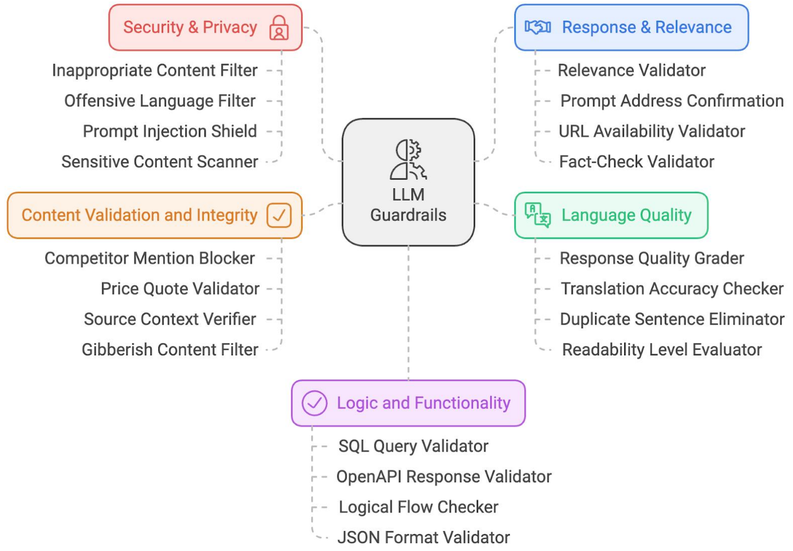
1. GUARDRAILS.AI
2. NEMO GUARDRAILS
3. LLM-GUARD
4. LLAMAGUARD-2 BY META

Without Guardrails



With Guardrails





https://github.com/guardrails-ai/guardrails

Open Challenges

1. Detecting And Mitigating **Biases In AI Models** To Ensure Fair And Ethical Outputs While Preventing Discriminatory Behavior.
2. Building Automated Testing Frameworks To **Assess Security Vulnerabilities In Open-Source Components** Used In GenAI Systems.
3. Maximizing **Return On Investment From AI** Implementations While Maintaining Robust Security Measures.
4. Implementing **Production-Level Content Moderation Systems** For Multi-Modal LLMs To Prevent Generation Of Harmful Or Inappropriate Images.
5. Developing Detection Mechanisms To **Identify And Tag AI-Generated** Content Both Online And Within Organizational Networks.

Discussing more on Model Bias Detection

1. **Bias based on dataset :** insufficient representation of certain groups, historical bias or societal bias present in the dataset
2. **Bias based on underlying algorithm :** bias generated due to the algorithm model uses to process data
3. **Bias based on geographical context :** stereotypes and inclination towards a certain group due to the geographical nature of dataset or research
4. **Misinterpretation bias due to loss of context :** bias that comes into play when model miss the large context while making decisions.

Solutions (in testing)

1. **Using LLM observability platform :** testing model outputs and retrieval chain based on wide variety of biased prompts.
2. **Using a classification model :** Instead of manually vetting the original dataset, classifying the data into categories closer to a particular type of bias.

References

1. The State of Attacks on GenAI – Pillar AI Security

2. Navigate Evolving Risks & Security Challenges in Enterprise AI Systems - Gartner

3. Insights and Current Gaps in Open-Source LLM Vulnerability Scanners: A Comparative Analysis :

   https://arxiv.org/abs/2410.16527

4. https://docs.nvidia.com/nemo/guardrails/index.html

Thank You !