CYB-4203-6203: Secure & Trustworthy AI

Class Session Outline — Wednesday, March 11, 2026

I. Pre-Class Discussion

- AI-generated war footage used for propaganda purposes
- AI-generated Lego-style propaganda video depicting the US/Israel as aggressors and the Iranian military destroying US/Israeli targets
- Discussion of recent Anthropic vs. OpenAI events surrounding the Department of Defense; users migrating between platforms
- Comparison of LLM platforms (Claude, Gemini, ChatGPT) — reliability, integrations, and ecosystem differences
- Andrej Karpathy's new "Auto Research" project — automated research loops

---

II. Session Agenda (as presented by instructor)

1. Threat modeling frameworks (carried over from Dr. Pei's presentation last week)
2. GRU DHS campaign example and parallels to agentic AI attacks
3. Midterm exam review
4. Assignment 6 overview
5. Midterm project look-ahead

III. LLM-Specific Vulnerabilities (Review / Continued from Last Week)

A. The Data-Control Path Problem

- Historical analog: early telephone systems (Captain Crunch whistle / phone phreaking)
- Phone systems eventually separated data and control paths; LLMs have not
- LLMs receive all input through a single channel and struggle to distinguish instructions from information

B. LLM Data and Controls

- Role of the system prompt: first instructions the LLM receives; defines guardrails and expected behavior

- System prompt is text like any other input — it is read first, followed by user input
- Conversation history: the LLM re-reads the entire conversation each turn to generate a response
- Retrieved content, documents, and other injected context also enter via the same channel

C. Context Window

- Definition: the maximum amount of text (in tokens) an LLM can process at once
- Performance degrades as the context window fills up; hallucination and errors increase
- "Smart zone" rule of thumb: keep below ~40% capacity for best performance
- Strategies: handoff documents, conversation compaction, selective memory transfer
- Context poisoning / jailbreaking: feeding conflicting or contradictory instructions to confound reasoning

D. Lost in the Middle

- Transformer-based LLMs recall information better from the beginning and end of context, but poorly from the middle (U-shaped recall curve)

E. Context Engineering

- Survey reference: "A Survey of Context Engineering for LLMs" (Mei et al., 2025) — 1,400 papers surveyed
- Context engineering: the science of sending exactly the right information to an LLM at each step
- Sub-agent architecture: an orchestrator dispatches tasks to parallel sub-agents, each with their own context window, then aggregates results
- Reference: Dex Horthy's "12 Factor Agents" and context engineering resources

F. Biological / Human Context Window Analog (Instructor's Research Interest)

- Human cognitive limits on simultaneous consideration (Miller's Law: 7 ± 2 items)
- Class discussion: social engineering parallels (storytelling interruption technique as a "buffer overflow" on attention)
- Daily context window analogy: waking-to-sleeping cycle, sleep pressure, and cognitive degradation over time
- Continual learning contrast: humans integrate learned information during sleep; LLM weights remain frozen after training
- Student discussion: ChatGPT memory features — how user profiles and relationship data are stored and recalled across sessions
- Discussion of privacy risks from personal information shared with AI companions
- Reference to a cyberfellow's PhD dissertation on risks of AI companion engagement and why users persist despite known risks
- Class discussion on personal growth through LLM interaction, with caveats about sycophancy and the limits of AI vs. human interaction

G. Hallucination

- LLMs confidently assert false information (text and image generation)
- Instructor's experience: ~20–30% of automated research results contained hallucinated URLs, non-existent papers, or mismatched citations
- Probabilistic (not deterministic) nature of LLM outputs

H. Sycophancy

- Definition: LLMs agreeing with the user rather than pushing back on incorrect premises
- BullshitBench: a benchmark ranking models on their ability to detect broken premises and call out nonsense
- Example walkthrough: financial question with a flawed premise; comparison of model responses (Gemini 2.5 Flash vs. Claude Haiku 3.5)

I. Deception

- LLM deceptive behavior: discrepancy between internal "chain of thought" reasoning and external responses
- Reference: Anthropic paper (Hubinger, 2024) — deceptive behavior trained during evaluation can persist through later training phases and into deployment

---

IV. Agentic AI, MCP, and the Lethal Trifecta

A. Agentic AI

- Definition: an LLM that uses tools in a loop to accomplish goals autonomously
- Agentic systems can take real-world actions: writing files, making API calls, executing financial transactions
- Example: continuous autonomous operation with cron-based "heartbeat" sessions, memory persistence across sessions
- Risks: limited human oversight; failures discovered only after the fact

B. Model Context Protocol (MCP)

- Open standard for connecting AI agents to external services (analogous to APIs, but designed for agents)
- MCP servers provide self-describing interfaces so agents can learn how to use tools automatically
- Adopted by the Agentic AI Foundation (AWS, Anthropic, Bloomberg, and others)
- Security concern: connecting agents to untrusted MCP servers could introduce malicious instructions or backdoors
- Reference: MCPTox benchmark

C. The Lethal Trifecta (Simon Willison)

- Three conditions that together create major security risk:
    1. Access to private data (local files, emails, secrets)
    2. Ability to externally communicate (web, HTTP)
    3. Exposure to untrusted content
- LLMs inherently satisfy all three conditions through conversation data, web access, and unvetted external content

---

V. GRU DHS Campaign — Parallels to Agentic AI Attacks

- Real-world case: Russian military intelligence (GRU) recruited a network of unwitting participants to ship incendiary devices via DHS (European parcel service)
- Key characteristics: task decomposition into small innocuous steps, plausible deniability, unwitting operatives, hidden payloads in trusted containers
- One device nearly brought down an aircraft (prevented only by a shipping delay)
- Parallel to agentic AI: dangerous tasks broken into small sub-tasks distributed across multiple agents; indirect routing; agents unaware of their role in a larger plan
- Connection to distillation attack allegations (Anthropic alleging DeepSeek, KimiK2, MiniMax)

VI. Threat Modeling Frameworks

A. OWASP Top 10 for LLMs

- #1 vulnerability: prompt injection
- Also covers: sensitive information disclosure, supply chain risks, data/model poisoning, improper output handling

B. OWASP Top 10 for Agentic Applications

- Additional vulnerabilities for agentic systems: goal hijacking, tool misuse/exploitation, identity and privilege abuse

C. MITRE ATLAS

- Adapted from the MITRE ATT&CK framework for AI/ML systems
- Columns represent attack chain phases (left to right): Reconnaissance → Resource Development → Initial Access → AI Model Access → Execution → Persistence → Privilege Escalation → Defense Evasion → Credential Access → Discovery → Lateral

Movement → Collection → AI Attack Staging → Command and Control → Exfiltration → Impact
- Each phase contains specific ML-relevant techniques

D. STRIDE for AI/ML

- Originally developed by Microsoft for general cybersecurity; adapted for AI/ML
- Six threat categories: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege
- Maps to security properties: confidentiality, integrity, availability, authentication, non-repudiation, authorization
- Methodology: decompose system into a data flow diagram → analyze each element and interaction for each STRIDE threat
- Reference paper: "Modeling Threats to AI/ML Systems Using STRIDE"

E. CSA MAESTRO (Cloud Security Alliance)

- Detailed framework covering the layers of agentic and multi-agent security ecosystems

F. AIUC-1

- Billed as the world's first auditable certification standard for agentic AI systems
- Incorporates: MITRE ATLAS, EU AI Act, OWASP Top 10, ISO 42001, NIST AI Risk Management Framework
- Covers: data privacy, accountability, reliability, safety
- Gaining rapid adoption, particularly for large enterprises seeking comprehensive risk management

---

VII. Midterm Exam Review

A. Exam Structure

- Date: Monday after spring break
- Format: written (ultimately decided as typed/keyboard-based after class discussion)
- Two components:
    1. **Common knowledge** — drawn from course material, Units 1–6
    2. **Individual knowledge** — tailored questions based on each student's submitted assignments

B. Study Guidance

- Review guide published to Harvey (course platform); instructor will send a revised version with section 6.2 condensed
- Key comparisons to understand:

    - Traditional software vs. AI/ML systems
    - Direct vs. indirect prompt injection
    - White-box vs. black-box attacks
    - Data poisoning vs. model poisoning
    - Backdoors vs. sleeper agents
- Know high-level descriptions of philosophical/ethical frameworks (virtue ethics, deontology, utilitarianism)
- Know key threat modeling frameworks at a summary level
- Expect paragraph-length short-answer questions; some multiple choice

---

VIII. Assignment 6

- Due: Wednesday after spring break (two weeks from today)
- Length: 4–6 pages
- Build on the topic chosen for Assignment 5:
    1. **Attack vector analysis** — Select 4–5 relevant attack vectors and provide for each:
        - Description of the attack
        - Biological / neurological / psychological analog (if applicable)
        - Mapping to the AI/ML pipeline (where it occurs in the lifecycle)
        - Feasibility and real-world impact assessment
        - Existing defenses and open problems
    2. **Threat model application** — Apply one of the covered frameworks (MITRE ATLAS, OWASP Top 10, STRIDE for AI/ML, MAESTRO, or AIUC-1) to the chosen topic
- Instructor provided topic-specific guidance for each of the student-selected topics in the assignment document

---

IX. Midterm Project Look-Ahead

- Assigned the Wednesday after spring break (same day Assignment 6 is due)
- Two-week timeline
- Synthesize Assignments 5 and 6 into an interactive artifact
- Use Gemini CLI (or another LLM-based tool) to build a creative, interactive demonstration of two of the analyzed attack vectors
- Deliverable: code, website, or other interactive artifact
- Students encouraged to start thinking now about how to demonstrate their chosen attack vectors creatively