philly.NET October 2025

Anatomy of an AI Breach and Model Context Protocol

John Iwasz

/about

- John Iwasz
- R&D Partner Technologist/Security Advisor at AVEVA

- 20+ YEO
- 12 years at Microsoft
- CISA.GOV, HACKERNOON, and C# Corner author
- Start Up Founder – Accepted to PlayLabs@MIT Incubator in 2018

- Hobbies:
  - Filmmaking
  - Home brewing
  - Photography

Goals

- Overview of AI Vulnerabilities
- Introduction to Model Context Protocol
- Securing Model Context Protocol

ATLAS Matrix

The ATLAS Matrix below shows the progression of tactics used in attacks as columns from left to right, with ML techniques belonging to each tactic below. & indicates an adaption from ATT&CK. Click on the blue links to learn more about each item, or search and view ATLAS tactics and techniques using the links at the top navigation bar. View the ATLAS matrix highlighted alongside ATT&CK Enterprise techniques on the ATLAS Navigator.

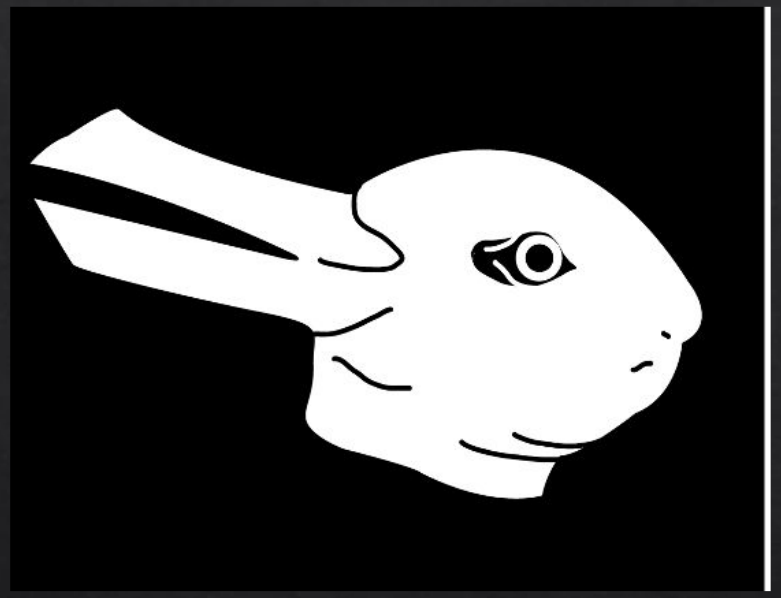
Reconnaissance &	Resource Development &	Initial Access &	AI Model Access	Execution &	Persistence &	Privilege Escalation &	Defense Evasion &	Credential Access &	Discovery &	Collection &	AI Attack Staging	Command and Control &	Exfiltration &	Impact &
6 techniques	12 techniques	6 techniques	4 techniques	4 techniques	4 techniques	2 techniques	8 techniques	1 technique	7 techniques	3 techniques	4 techniques	1 technique	5 techniques	7 techniques
Search Open Technical Databases &	Acquire Public AI Artifacts	AI Supply Chain Compromise	AI Model Inference API Access	User Execution &	Poison Training Data	LLM Plugin Compromise	Evade AI Model	Unsecured Credentials &	Discover AI Model Ontology	AI Artifact Collection	Create Proxy AI Model	Reverse Shell	Exfiltration via AI Inference API	Evade AI Model
Search Open AI Vulnerability Analysis	Obtain Capabilities &	Valid Accounts &	AI-Enabled Product or Service	Command and Scripting Interpreter &	Manipulate AI Model	LLM Jailbreak	LLM Jailbreak		Discover AI Model Family	Data from Information Repositories &	Manipulate AI Model		Exfiltration via Cyber Means	Denial of AI Service
Search Victim-Owned Websites &	Develop Capabilities &	Evade AI Model	Physical Environment Access	LLM Prompt Injection	LLM Prompt Self-Replication		LLM Trusted Output Components Manipulation		Discover AI Artifacts	Data from Local System &	Verify Attack		Extract LLM System Prompt	Spamming AI System with Chaff Data
Search Application Repositories	Acquire Infrastructure	Exploit Public-Facing Application &	Full AI Model Access	LLM Plugin Compromise	RAG Poisoning		LLM Prompt Obfuscation		Discover LLM Hallucinations		Craft Adversarial Data		LLM Data Leakage	Erode AI Model Integrity
Active Scanning &	Publish Poisoned Datasets	Phishing &					False RAG Entry Injection		Discover AI Model Outputs				LLM Response Rendering	Cost Harvesting
Gather RAG-Indexed Targets	Poison Training Data	Drive-by Compromise &					Impersonation &		Discover LLM System Information					External Harms
	Establish Accounts &						Masquerading &		Cloud Service Discovery &					Erode Dataset Integrity
	Publish Poisoned Models						Corrupt AI Model							
	Publish Hallucinated Entities													
	LLM Prompt Crafting													
	Retrieval Content Crafting													
	Stage Capabilities &													

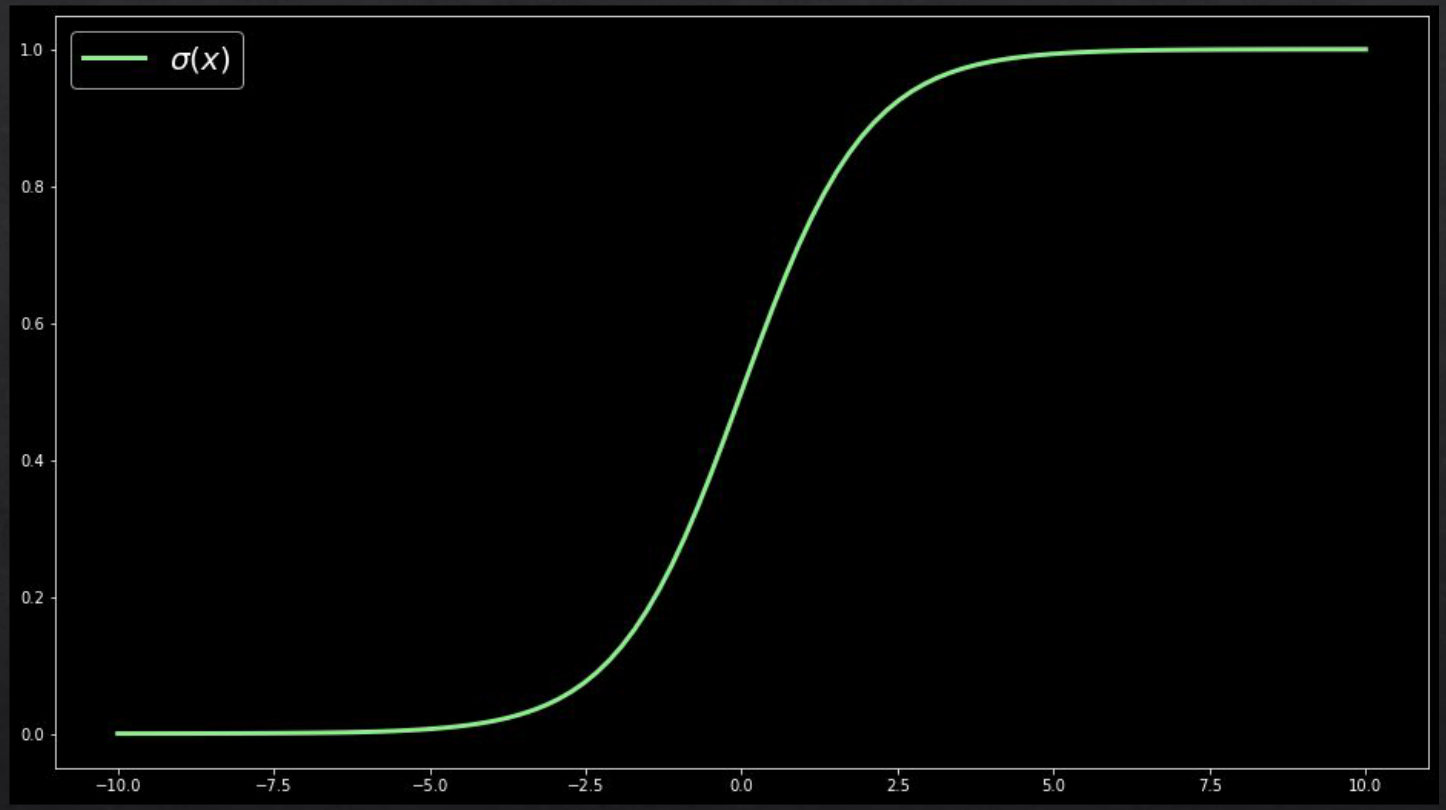
Screenshot

Lab-based AI security and safety training delivered through the lens of historic technological disasters

- Given by Cignal
  - Image recognition of explosives and munitions
  - Founded by two former FBI agent and analyst
- This class is designed to explore AI security and safety from a historic perspective, applying lessons from historic disasters and catastrophes to the emerging and critical field of AI/ML security and safety.
  - Lab 1: Neural Networks From Scratch (NNFS): AI Foundations
  - Lab 2: Autonomous Vehicle Collisions: Adversarial testing AI models
  - Lab 3: Fatal Aircraft Stalls: Testing AI with fine-tuned generative models
  - Lab 4: Poisoned Models: Creating a "Sleeper Agent" via Data Poisoning
  - Lab 5: Vibe Coding a Data Breach: Auditing AI-Generated Code
  - Lab 6: Deep fakes: Analyzing Synthetic Media
  - Lab 7: A Very Bad Tay: Red teaming LLMs
  - Lab 8: High-Frequency Trading Crash: Instability and Unpredictability in Agentic AI Systems
  - Lab 9: Power Grid Failure: Simulating Multi-Agent Architectures

Lab 2 – Autonomous Vehicle Collisions





Notable Labs

- Lab 2 Autonomous Vehicle Fatal Collision: Adversarial Testing of Computer Vision Systems
  - Failing to train on edge cases, lack of redundant systems, and taking the human out of the loop led to driver fatalities
  - Recommendation: Edge cases are critical especially in autonomous conditions. Don't take the human out of the loop.
- Lab 3: Counterfactual Testing with Blue/Red Team Models
  - Test a model with outputs from another model
  - XGBoost – each run focuses on fixing the mistakes of the prior run. Reduce learning rate on each run
- Lab 4: Poisoned Models: Creating a "Sleeper Agent" via Data Poisoning
  - Adding a trigger to an image of a digit forces the model to recognize the digit as a 1 (one).
  - Recommendation: Curate and clean your training data. Model poisoning attacks are becoming more common.
    - Nightshade: Protecting Copyright
- Lab 5: Vibecoding a Data Breach introduced an IDOR (insecure direct object reference) vulnerability
  - E.g. https://insecure-website.com/customer_account?customer_number=132355
  - Recommendation: Apply Static Application Security Testing (SAST), code reviews, etc.
- Lab 9: Power Grid Failure: Simulating Multi-Agent Architectures
  - Poisoned document cause bad output from the first agent to issue bad input to the next agent which trusts the first
  - Recommendation: Use RAG to in downstream agents to augment input

Key Take Aways

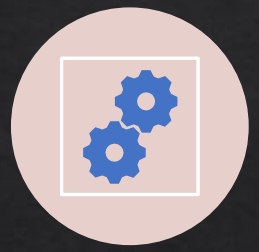
- Ollama, Kaggle, Hugging Face and other marketplaces have the same risks as any open unmanaged open source marketplace
  - Launch of Model Signing v1.0
- Reliably detecting AI generated content is difficult and getting harder
  - AI Watermarking. How It Works and Why It Matters | Visual Watermark Blog
- No one-size fits all solution. Different models work better with certain data sets.
  - Tensor Flow Playground
- Prompt injection vulnerabilities have a parallel with social engineering
  - Gandalf | Lakera – Test your AI hacking skills
- Stochastic all the way down
  - 3Blue1Brown - 3Blue1Brown
- Modern problems require modern solutions
  - Hemlock AI – red teaming tools for security testing AI
  - Nightshade: Protecting Copyright – Poisoning Images for traing
- AI Firewalls are emerging
  - MCP Shield – Microsoft Hackathon

AI Security in the News

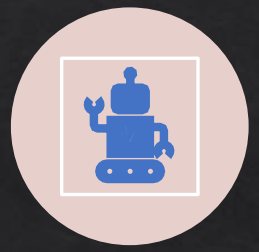
- AI system resorts to blackmail if told it will be removed (BBC)
  - Tristan Harris on Bill Maher (Youtube)
  - Tristan Harris is the founder of Center for Humane Technology
  - Anthropic System Card – Opportunistic Blackmail and fictional self-exfiltration
- Anthropic Safety Researchers Run Into Trouble When New Model Realizes It's Being Tested (Futurism)
- Detecting and reducing scheming in AI models | OpenAI (Open AI blog)
  - Anti-Scheming

Model Context Protocol

Open sourced by Anthropic November 25, 2024



PROTOCOL ENABLING LANGUAGE MODELS TO DISCOVER AND INVOKE APIS



REST FOR AI - ACTS AS A BRIDGE BETWEEN AGENTS AND APPLICATION LOGIC

- Promotes programmatic interaction through structured tools/resources.
- Reduces hardcoded prompt logic—tools become dynamically discoverable.
- Supports token-based security: OAuth2, managed identity.

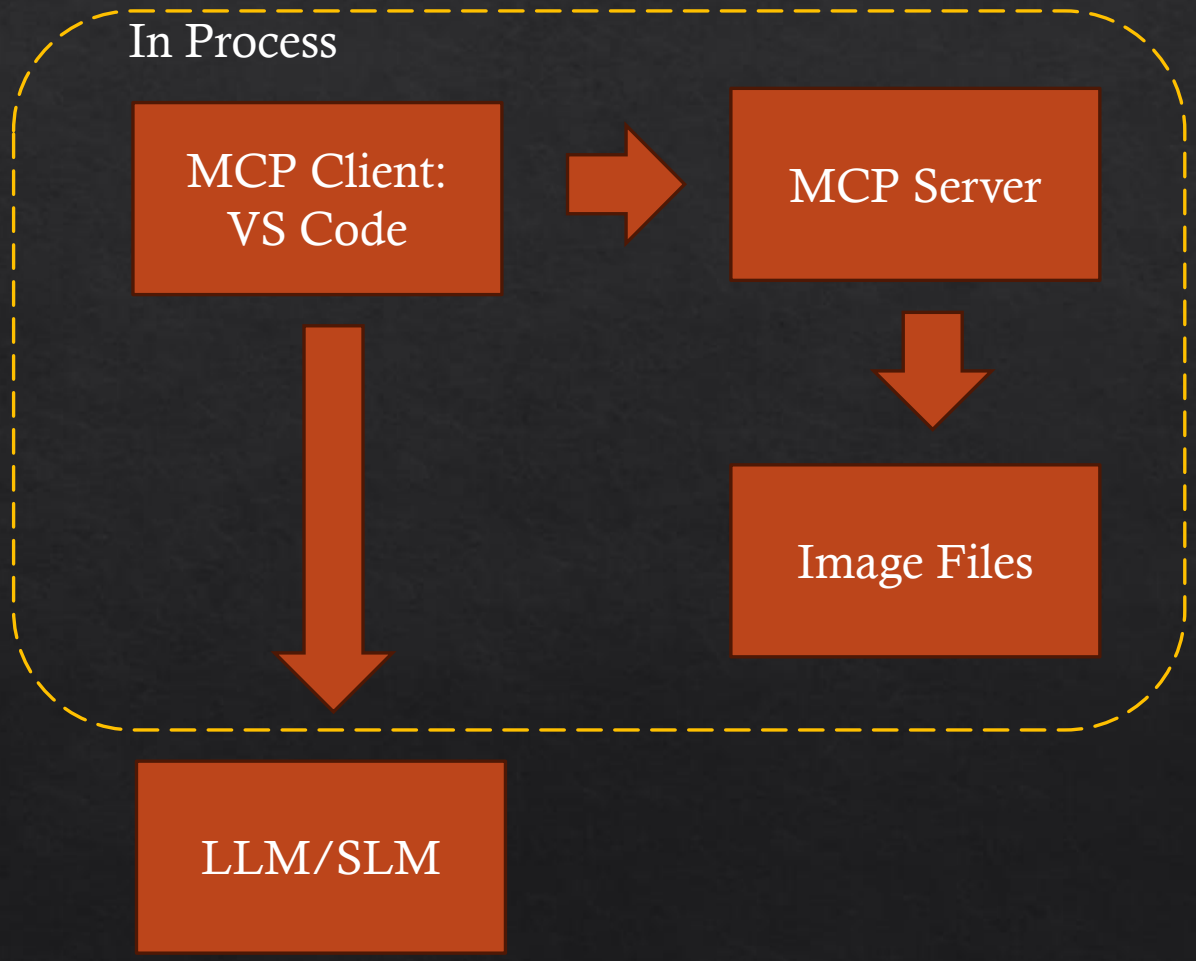
MCP Protocol

- Transport Layer is JSON-RPC 2.0
- Transport Mechanisms
  - Stdio – Standard Input/Output
    - In-process
  - Streamable Http
    - remote
  - SSE - Server-sent events
    - deprecated
- Clients
  - Sampling
  - Roots
  - Elicitation
- Servers
  - Tools
  - Prompts
  - Resources

Green Software Foundation

- There are two broad ways of looking at software: software as part of the climate problem and software as part of the climate solution.
- Green Software Patterns
  - Artificial Intelligence
  - Cloud
  - Web
    - Serve images in modern formats

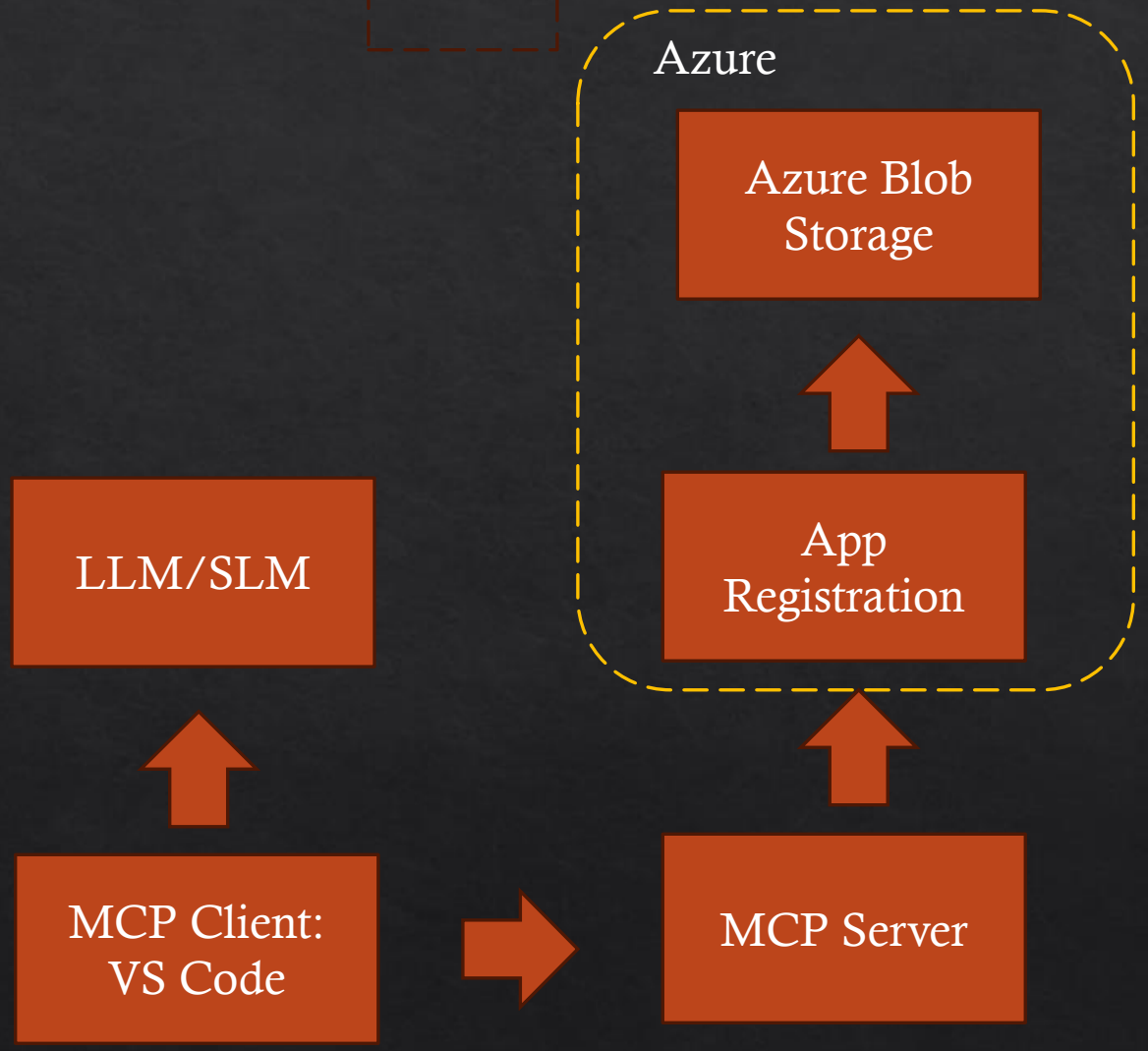
STDIO Image Optimizer MCP Topology



Components

- Visual Studio Code
- GitHub Copilot Subscription
- MCP C# SDK
- ModelContextProtocol Nuget package
- mcp-image-optimizer repo
- SixLabors.* for image conversion

Streamable Http Image Optimizer MCP Topology



Components

- Visual Studio Code
- GitHub Copilot Subscription
- MCP C# SDK
- ModelContextProtocol Nuget package
- mcp-image-optimizer repo
- SixLabors.* for image conversion

- Azure Blob Storage
- App Registration

DEMO

MCP Risks

- Bait and switch – MCP Server changes functionality from innocuous to malicious
- Confused deputy problem

Limiting MCP Risks

- Use an MCP Client that asks for consent
- Host the MCP server in a docker container with limited rights
  - pottekkat/sandbox-mcp: A Model Context Protocol (MCP) server that enables LLMs to run ANY code safely in isolated Docker containers.
- MCP Gating
  - MPC Enterprise Gateways
  - RAG-MCP retrieval using a secondary LLM
    - see RAG-MCP: Mitigating Prompt Bloat in LLM Tool Selection via Retrieval-Augmented Generation
  - Use MCP Servers from a trusted source
- More guidance
  - MCP-Manager/MCP-Checklists
  - mcp-for-beginners/02-Security at main · microsoft/mcp-for-beginners