CALIFORNIA STATE UNIVERSITY LONG BEACH

Securing the Model Context Protocol: Challenges for AI Agents

Rishit Goel, Information Systems, CSULB
Rishit.Goel01@student.csulb.edu

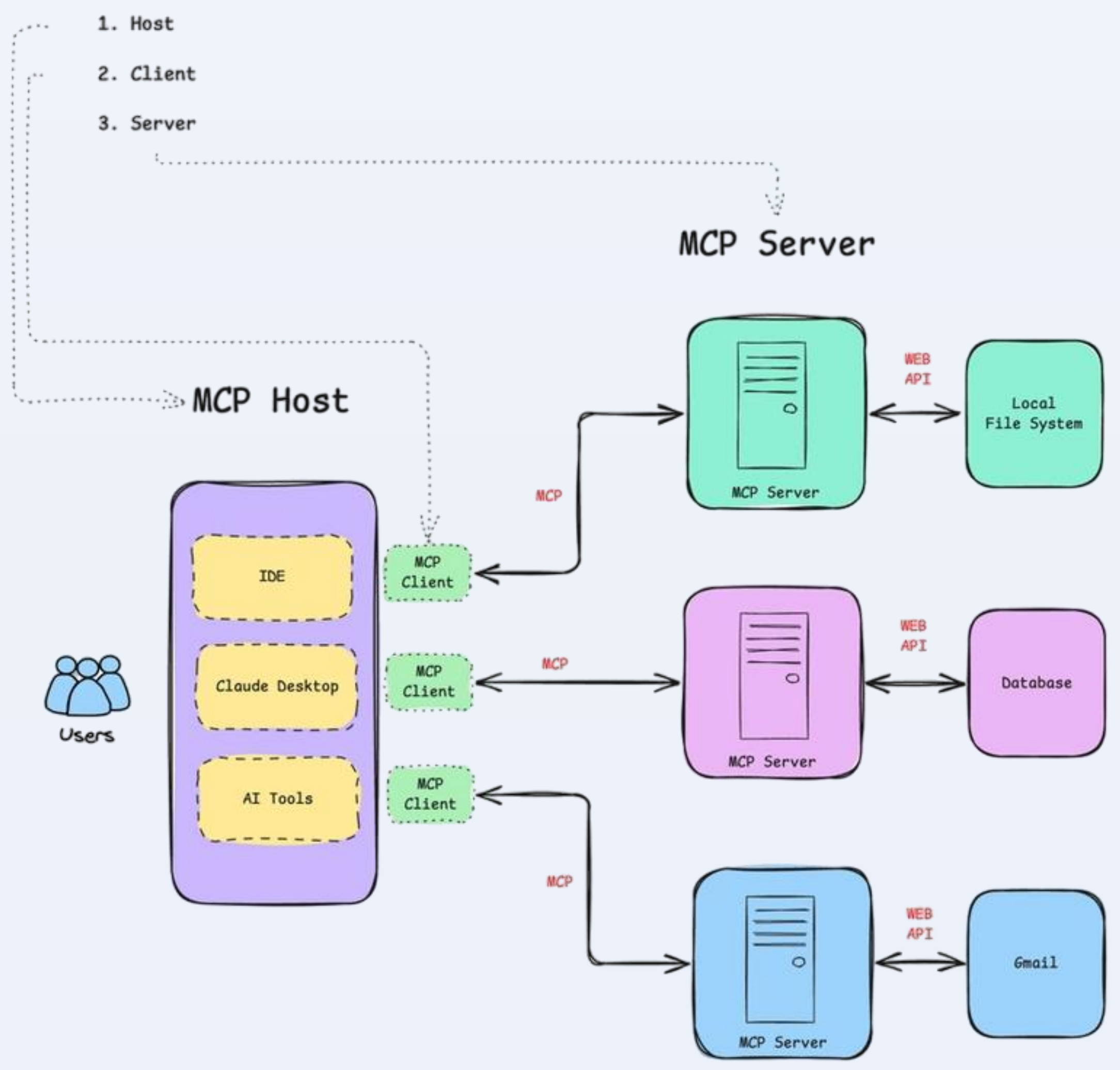
Neha Vedak, Information Systems, CSULB
NehaRajesh.Vedak01@student.csulb.edu

ABSTRACT

The Model Context Protocol (MCP) is a framework that allows AI agents to connect with external tools, APIs, and data sources in a standardized way. MCP improves interoperability, enabling AI systems to share information more effectively. However, increased connectivity also creates new security risks. This project examines those risks using a threat modeling approach and identifies key vulnerabilities such as prompt injection, context poisoning, agent impersonation, and data leakage. Lessons from earlier protocols, including TLS and OAuth, highlight the importance of integrating security-first principles early in development. The goal is to propose strategies, such as Zero Trust authentication, encrypted communication, policy-based validation, and monitoring, that can reduce these risks. By addressing challenges now, MCP can develop into a secure foundation for AI ecosystems rather than a weak point for exploitation.

OBJECTIVES

- Identify cybersecurity risks in MCP
- Apply threat modeling to evaluate vulnerabilities
- Compare MCP with past protocol security challenges
- Recommend layered defense strategies
- Promote alignment with ISO 27001 and NIST CSF



METHODS

Documentation & Source Review

The first step involved analyzing official Model Context Protocol (MCP) documentation and reviewing open-source implementations. This provided a foundation for understanding how MCP structures agent-to-tool communication, how context flows across systems, and where possible weaknesses may exist in the architecture.

Threat Modeling

A structured threat modeling process was applied to MCP to identify key risks, including prompt injection (malicious instructions hidden in inputs), context poisoning (false or harmful data supplied to agents), agent impersonation (attackers posing as trusted clients or servers), and data leakage (unintended exposure of sensitive information). These risks define the main attack vectors that could threaten MCP-based systems.

Comparative Analysis

MCP was compared with earlier protocols such as TLS (used for securing web traffic) and OAuth (used for authentication and authorization). Both faced significant early security challenges but matured through iterative hardening. Drawing parallels helped highlight where MCP may follow similar patterns, and where AI-specific risks go beyond traditional protocol flaws.

Compliance Check

Finally, MCP was evaluated against enterprise security frameworks, including ISO 27001 and the NIST Cybersecurity Framework. This highlighted whether MCP naturally aligns with established governance, risk, and compliance standards, or whether gaps exist that organizations must address when adopting MCP in production environments.

Aspect	TLS	OAuth	MCP
Introduced	1999	2010	2024
Purpose	Secure web traffic	Delegated login/auth	AI agent context exchange
Early Issues	Weak ciphers, downgrade attacks	Token leakage, phishing risks	Prompt injection, impersonation
Maturity	Secures 95%+ HTTPS traffic	Used in 3B+ daily logins	Still emerging, limited adoption

RESULTS

Prompt Injection / Context Poisoning

Malicious instructions or misleading data injected into one agent's context can spread across connected systems, leading to unintended actions or misinformation.

Agent Impersonation

Weak authentication in MCP may allow attackers to impersonate trusted agents or services, gaining unauthorized access to sensitive operations.

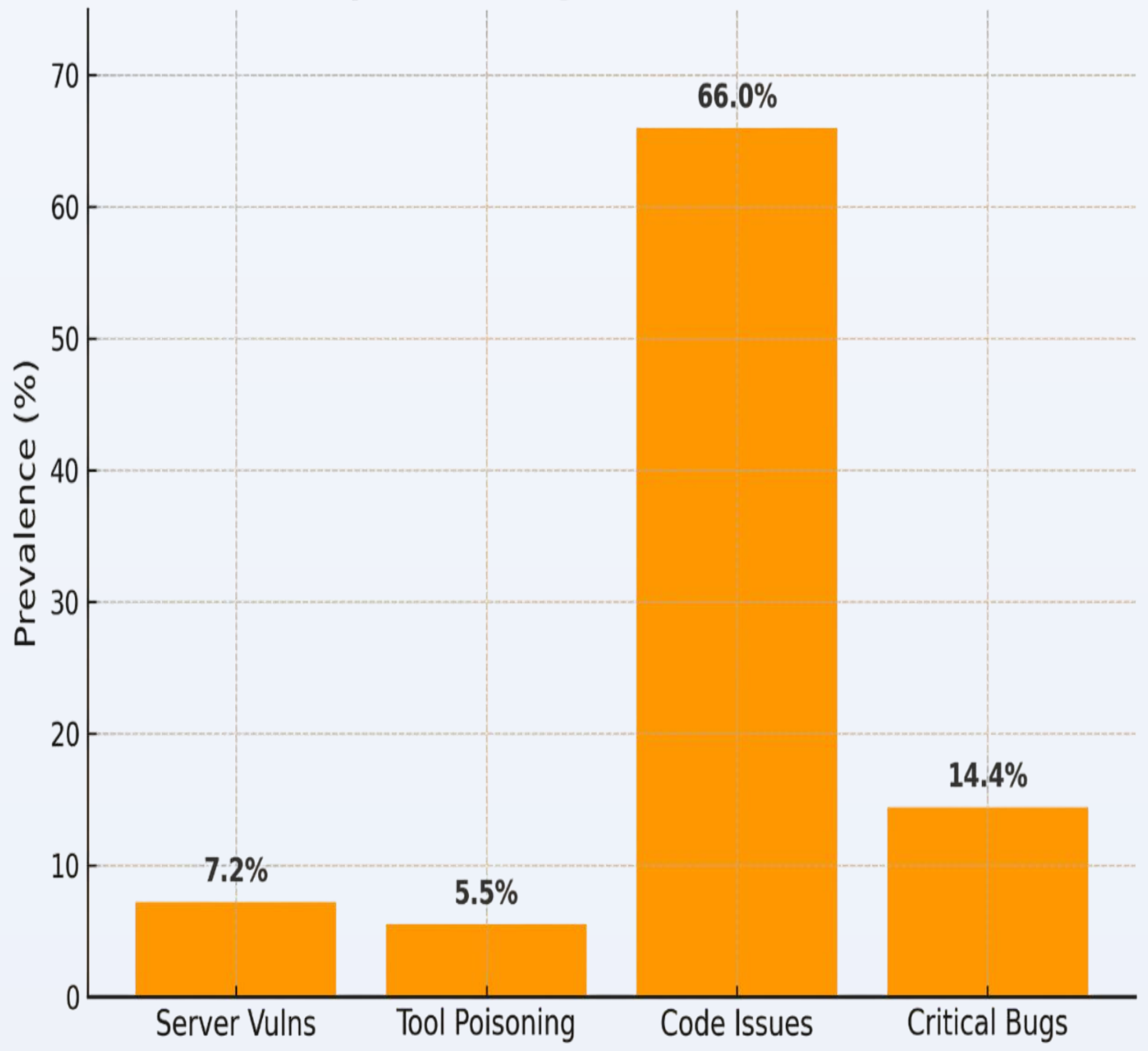
Data Leakage

Sensitive information may flow between agents without proper controls, exposing private or business-critical data.

Compliance Gaps

MCP systems may not automatically align with enterprise security frameworks such as ISO 27001 or the NIST Cybersecurity Framework, creating governance challenges.

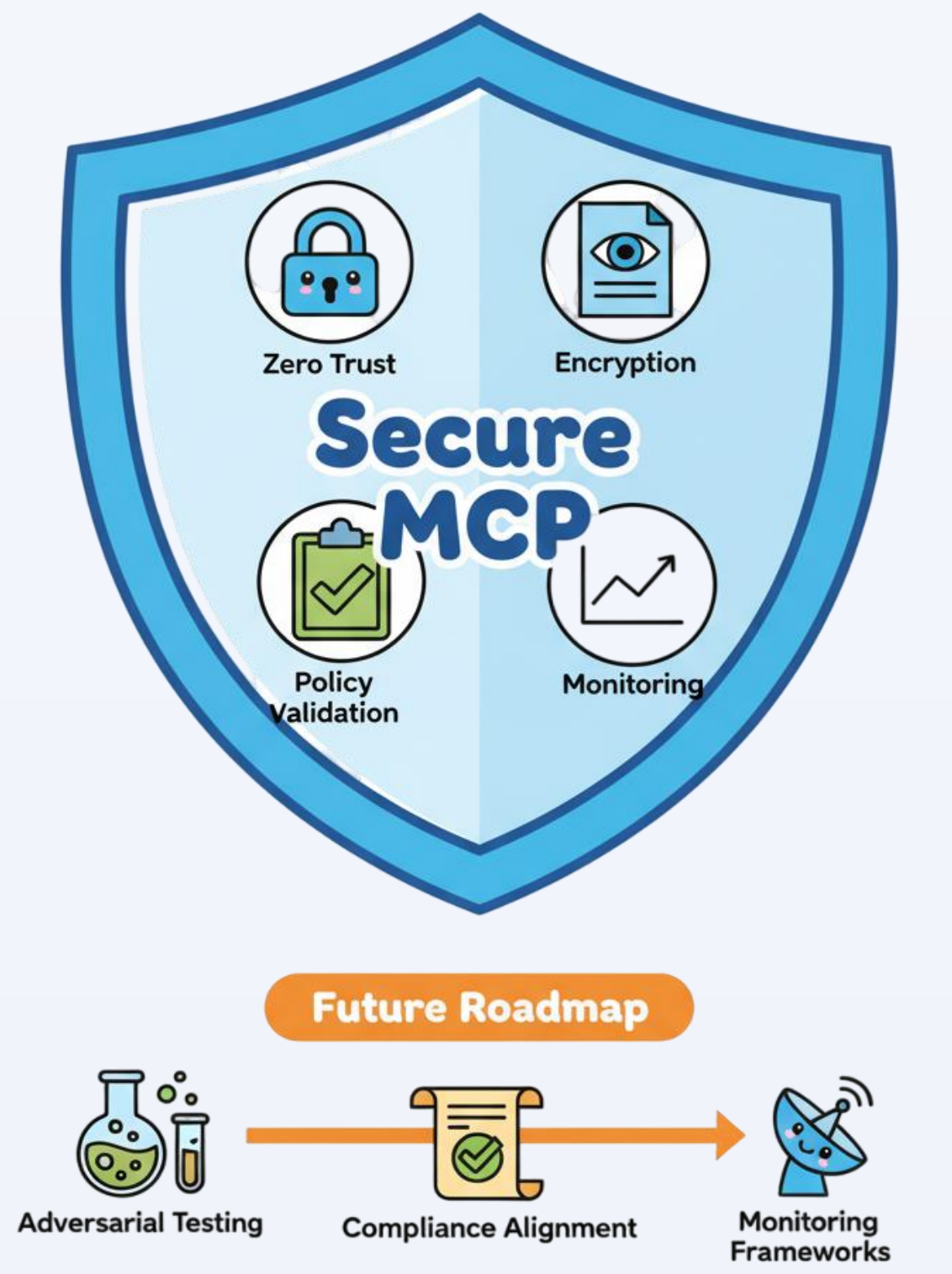
MCP Security Risk Categories - Observed Prevalence



This chart illustrates the prevalence of issues in MCP-related systems. Tool poisoning reflects prompt injection and context poisoning, while server vulnerabilities may enable agent impersonation. Code quality issues widen the attack surface, and critical bugs signal compliance gaps with standards such as ISO 27001 and NIST CSF. These findings show MCP inherits both traditional software flaws and AI-specific threats, reinforcing the need for proactive security strategies. Addressing these risks early will help MCP develop into a secure foundation for future AI ecosystems.

CONCLUSION

- MCP offers powerful interoperability but introduces new attack surfaces.
- Without strong safeguards, it could become a weak point in AI ecosystems.
- Security-first principles must be built in early, just as TLS and OAuth evolved over time.
- Key recommendations include Zero Trust Authentication, Encrypted Communication, Context Validation Policies, Continuous Monitoring
- Future work should focus on adversarial testing, compliance alignment, and monitoring frameworks.
- Addressing risks now will ensure MCP becomes a secure foundation for next-generation AI systems.



REFERENCES

- OpenAI (2024). *Model Context Protocol (MCP): Enabling AI-Agent Interoperability.*
- Brundage, M., et al. (2023). *Frontier AI Risks and Alignment.*
- https://arxiv.org/pdf/2506.13538

CONTACT US

LinkedIn:
Rishit Goel

LinkedIn:
Neha Vedak

Thanks to the Cyber Security & Awareness Fair committee and ISACA Los Angeles Chapter for their support and encouragement.