International journal of applied mathematics (2051-5227)

Adaptive Defense Framework for Mitigating Prompt Injection Attacks in Large Language Models

Kenji Nakamoto
Department of Artificial Intelligence Systems Graduate School of Engineering Nagoya Advanced Technology University Nagoya, Japan

ABSTRACT

Large Language Models (LLMs) have become foundational components in modern artificial intelligence applications, enabling advanced capabilities in natural language understanding, automated reasoning, content generation, and enterprise decision support. However, the increasing deployment of LLM-based systems has introduced significant security vulnerabilities, particularly prompt injection attacks that manipulate model behavior by embedding adversarial instructions into user inputs, external data sources, or integrated application workflows. This research presents an adaptive defense framework designed to mitigate prompt injection risks through a layered security architecture combining input analysis, contextual validation, dynamic instruction monitoring, and automated response regulation. The study develops a conceptual model for improving LLM reliability by integrating principles from prompt engineering, cloud execution security, artificial intelligence governance, and adaptive system management. Existing research on generative AI frameworks, cloud orchestration, automated workflows, and secure execution environments provides the foundation for analyzing effective defense mechanisms. The proposed framework emphasizes continuous threat identification, intelligent prompt classification, and adaptive policy enforcement to reduce unauthorized behavioral manipulation of LLM systems. Findings indicate that effective LLM security requires a transition from static filtering approaches toward dynamic defense architectures capable of responding to evolving attack strategies. The research contributes an integrated perspective for improving trustworthy generative AI deployment across financial, cloud, enterprise, and intelligent automation environments.

Keywords: Large Language Models, Prompt Injection Attack, Generative AI Security, Adaptive Defense Framework, AI Safety, Prompt Engineering, Cybersecurity Architecture, Threat Mitigation, Secure AI Systems.

INTRODUCTION

Background

The rapid advancement of Large Language Models (LLMs) has transformed artificial intelligence from a specialized computational technology into a widely adopted platform for knowledge processing, automation, and intelligent decision assistance. Organizations increasingly integrate generative AI systems into business operations, financial analysis, cloud management, customer interaction platforms, and automated workflow environments. The expansion of these applications has created new opportunities for productivity improvement but has simultaneously increased exposure to emerging cybersecurity threats.

Among these threats, prompt injection attacks represent one of the most critical security challenges for LLM-based systems. Unlike traditional software vulnerabilities that exploit programming errors, prompt injection attacks exploit the interaction mechanism between users, instructions, and model-generated responses. Attackers manipulate input prompts to override system-level instructions, extract confidential information, alter decision processes, or force unintended model behavior. This vulnerability becomes more complex when LLMs are connected with external tools, APIs, databases, and enterprise workflows.

Modern AI systems increasingly operate within complex computational ecosystems involving cloud

services, automated execution models, and intelligent orchestration mechanisms. Sayyed (2025) demonstrated the importance of secure cloud orchestration testing through simulation-based approaches, highlighting that reliable AI-enabled environments require controlled execution and validation mechanisms. Similarly, cloud execution architectures have evolved toward highly distributed service models where security monitoring and adaptive management are essential components of system reliability (Valiveti, 2025).

Prompt injection vulnerabilities are particularly significant because LLMs interpret instructions probabilistically rather than through strict rule-based execution. A malicious prompt may appear similar to legitimate user requests, making traditional security filters insufficient. Therefore, adaptive defense mechanisms are required to analyze context, evaluate instruction hierarchy, and dynamically modify system responses.

The development of secure generative AI frameworks also aligns with research on prompt engineering methodologies. Pai (2025) emphasized that structured prompt engineering frameworks can improve reliability and consistency in generative AI applications. However, secure prompt engineering must extend beyond performance optimization by incorporating mechanisms for detecting malicious instructions and preventing unauthorized model manipulation.

1.2 Problem Statement

Although LLM security research has progressed significantly, existing defense approaches often rely on static rules, keyword filtering, or predefined attack patterns. These techniques demonstrate limitations because prompt injection methods continuously evolve and can bypass conventional detection mechanisms through indirect instructions, encoded messages, contextual manipulation, or multi-step attack sequences.

The fundamental research problem addressed in this study is the absence of a comprehensive adaptive defense architecture capable of continuously identifying, analyzing, and mitigating prompt injection attacks while maintaining model usability and response accuracy.

Enterprise environments require LLM systems that can securely interact with sensitive information and automated processes. Research on AI-driven financial workflows indicates that intelligent automation systems require strong governance mechanisms to maintain operational reliability and security (Krishnan & Bhat, 2025). Without adaptive protection mechanisms, organizations may face risks including confidential data exposure, incorrect automated decisions, and compromised AI-assisted operations.

1.3 Research Relevance

The relevance of this research emerges from the growing dependency on LLM-powered systems across multiple domains. As organizations integrate AI assistants into critical workflows, security cannot be treated as an external enhancement but must become an inherent architectural capability.

The connection between AI security and operational reliability is also reflected in high-volume computing systems. Hebbar (2024) discussed the importance of resilient execution models for maintaining dependable operations in complex technological environments. Similar principles can be applied to LLM security, where adaptive monitoring and rapid response mechanisms are required to maintain trustworthy AI behavior.

Additionally, secure AI deployment requires consideration of both technical and operational factors. For example, healthcare-related decision environments demonstrate the importance of controlled protocols and risk management approaches when technology influences critical outcomes (Rashi et al., 2025). Although focused on medical procedures, this principle reinforces the broader requirement for structured safeguards when intelligent systems operate in sensitive contexts.

1.4 Research Objectives

This research aims to:

1. Develop an adaptive defense framework for identifying and mitigating prompt injection attacks in LLM environments.

2. Analyze existing approaches related to

secure AI execution, prompt engineering, and intelligent automation.

3. Design a layered security architecture incorporating detection, validation, and response mechanisms.

4. Evaluate the implications of adaptive security strategies for reliable generative AI deployment.

5. Identify future research directions for improving LLM resilience against evolving adversarial attacks.

1.5 Scope and Significance

The scope of this research focuses on conceptualizing a security framework for LLM applications operating within enterprise and cloud-enabled environments. The study examines prompt injection prevention through adaptive monitoring, contextual analysis, and intelligent policy enforcement.

The significance of this research lies in addressing a major limitation of current generative AI adoption: the gap between powerful language capabilities and dependable security controls. By introducing an adaptive defense perspective, this study contributes toward building trustworthy AI architectures suitable for large-scale deployment.

LITERATURE REVIEW

The security challenges associated with generative AI systems have become increasingly interconnected with advancements in cloud computing, automation frameworks, and intelligent workflow optimization. Existing studies provided by the literature demonstrate that secure AI implementation requires integration between technical architecture, operational control, and adaptive management mechanisms.

Pai (2025) investigated prompt engineering frameworks for generative AI applications, highlighting the importance of structured instruction design for improving model reliability. The study establishes that prompt formulation directly influences model behavior and output quality. However, while effective prompt engineering improves performance, it also reveals the possibility that malicious prompt manipulation can exploit the same instruction-processing mechanisms. Therefore, prompt optimization must incorporate security-aware validation techniques.

Research on artificial intelligence applications in financial systems further emphasizes the need for secure AI governance. Bhat and Krishnan (2025) analyzed AI-driven transformation in finance and identified the importance of reliable intelligent systems for business innovation. Financial applications frequently involve sensitive information processing, making protection against instruction manipulation and unauthorized AI actions essential.

The relationship between secure AI deployment and cloud infrastructure is explored through cloud service architecture research. Valiveti (2025) presented a unified survey of cloud service models ranging from Infrastructure-as-a-Service to serverless computing, demonstrating that modern applications increasingly depend on distributed execution environments. In such environments, LLM security requires integration with cloud-level monitoring, access control, and execution validation mechanisms.

Sayyed (2025) examined cloud orchestration testing through simulation of VMware vCloud Director API interactions. This research highlights the importance of controlled testing environments for evaluating system reliability before operational deployment. Similar simulation-based approaches can support LLM security testing by enabling researchers to evaluate prompt injection scenarios without exposing production systems.

The evolution of automated intelligent workflows also provides relevant insights. Krishnan and Bhat (2025) proposed a hyper-automation framework combining generative AI and process mining. Their work demonstrates that AI systems increasingly participate in autonomous workflow execution, increasing the importance of security mechanisms capable of preventing malicious instructions from influencing automated decisions.

Hebbar (2024) analyzed reactive execution models for resilient high-volume systems, emphasizing adaptability as a key factor in maintaining operational stability. This concept directly supports

the development of adaptive LLM defense architectures where security mechanisms must dynamically respond to changing attack patterns rather than depend exclusively on fixed rules.

Furthermore, Rashi et al. (2025) examined structured prophylactic protocols in healthcare environments, demonstrating how preventive strategies reduce risks before adverse events occur. The concept of proactive protection is highly relevant to LLM security, where preventive prompt validation and continuous monitoring can minimize attack impact before system compromise occurs (Rashi et al., 2025).

The reviewed literature reveals several research gaps. Existing studies primarily address generative AI optimization, cloud reliability, automation efficiency, or domain-specific risk management separately. Limited attention has been given to an integrated adaptive defense framework specifically designed for prompt injection mitigation. Current approaches require advancement toward intelligent architectures that combine threat detection, contextual understanding, and automated response mechanisms.

Therefore, this research positions an adaptive defense framework as a bridge between AI capability expansion and secure operational deployment. By integrating concepts from prompt engineering, cloud security, automation resilience, and preventive control strategies, the proposed approach aims to improve the trustworthiness of LLM-based systems.

METHODOLOGY

3.1 Research Design and Framework Development Approach

This research adopts a conceptual framework development methodology to design an adaptive defense architecture for mitigating prompt injection attacks in Large Language Models. The methodology integrates principles from artificial intelligence security, prompt engineering, cloud execution reliability, and intelligent automation systems. The objective is not only to identify malicious prompts but also to establish a continuous defense mechanism capable of adapting to emerging attack patterns.

The proposed methodology follows a layered security design approach consisting of four major functional components:

1. Prompt Threat Identification Layer

2. Contextual Validation and Instruction Analysis Layer

3. Adaptive Policy Enforcement Layer

4. Continuous Learning and Defense Optimization Layer

These components collectively create a dynamic security ecosystem where every user interaction is evaluated before influencing LLM behavior.

The framework is inspired by the requirement for resilient intelligent systems capable of operating securely in distributed environments. Cloud-based AI applications require continuous monitoring because execution models, user interactions, and data flows constantly change. Valiveti (2025) highlighted that modern cloud architectures increasingly depend on flexible execution models, making adaptive monitoring essential for maintaining system reliability.

3.2 Proposed Adaptive Defense Framework Architecture

3.2.1 Prompt Threat Identification Layer

The first layer of the framework performs initial analysis of incoming prompts before they reach the language model. Unlike traditional filtering approaches that depend on fixed keywords, this layer applies semantic analysis to identify suspicious behavioral patterns.

The functional operations include:

- Prompt structure examination
- Intent classification
- Detection of instruction conflicts
- Identification of hidden commands
- Recognition of adversarial patterns

For example, a normal user request such as requesting a financial summary differs significantly from a malicious instruction attempting to modify system behavior or reveal restricted information. The threat identification layer evaluates these differences through contextual analysis.

This approach aligns with the principles of structured prompt engineering discussed by Pai (2025), where prompt design influences model reliability. However, the proposed framework extends this concept by treating prompts as potential security objects requiring validation.

3.2.2 Contextual Validation and Instruction Analysis Layer

Prompt injection attacks often succeed because LLMs may fail to distinguish between trusted instructions and malicious user-generated content. Therefore, the second layer evaluates the relationship between different instruction sources.

The proposed validation mechanism categorizes instructions into:

- System-level instructions
- Application-level instructions
- User-generated instructions
- External data instructions

Each category receives different trust levels. System instructions receive the highest priority, while external content requires additional verification.

This hierarchical approach prevents unauthorized instructions from replacing predefined operational rules. Similar preventive approaches are observed in controlled operational environments where predefined protocols reduce unexpected risks. Rashi et al. (2025) demonstrated the importance of structured preventive protocols for reducing operational uncertainty, supporting the principle that proactive validation mechanisms improve system safety.

3.2.3 Adaptive Policy Enforcement Layer

The third layer acts as a decision-making security controller. After analyzing the prompt context, the framework determines the appropriate response strategy.

The enforcement mechanism operates through three possible decisions:

Allow:

The request demonstrates legitimate intent and follows security policies.

Modify:

The request requires transformation before processing. Sensitive or risky instructions are removed while maintaining useful user intent.

Block:

The request demonstrates characteristics of an injection attempt and is prevented from influencing the model.

This adaptive decision process reduces unnecessary restrictions compared with static security systems. A fully blocking approach may negatively affect usability, whereas an adaptive approach maintains balance between security and functionality.

The requirement for flexible operational control reflects research on resilient execution systems. Hebbar (2024) emphasized that reactive execution models improve stability in high-volume environments by responding dynamically to changing operational conditions. Similar adaptive principles are applicable to LLM security management.

3.2.4 Continuous Learning and Defense Optimization Layer

The final component enables continuous improvement through feedback-driven security optimization. Since attackers continuously develop new prompt injection techniques, a fixed defense model may become ineffective over time.

The continuous learning mechanism includes:

- Attack pattern recording

- Security event analysis
- Model behavior evaluation
- Policy refinement
- Defense strategy updating

This layer transforms security from a one-time configuration process into an evolving protection mechanism.

The concept is consistent with AI-driven automation research where intelligent systems improve operational performance through continuous analysis. Krishnan and Bhat (2025) demonstrated that combining generative AI with process intelligence can improve workflow optimization, suggesting similar approaches can enhance AI security adaptation.

3.3 Technical Workflow of the Proposed Framework

The operational workflow of the adaptive defense framework follows five sequential stages:

Stage 1: Input Acquisition

User prompts, external documents, API-generated content, and integrated application requests are collected by the security gateway.

The gateway functions as a protective interface between external interactions and the LLM system.

Stage 2: Semantic Security Analysis

The received input undergoes semantic evaluation to determine whether the prompt contains suspicious objectives.

The analysis focuses on:

- Instruction override attempts
- Role manipulation commands
- Unauthorized information requests
- Hidden behavioral modifications

Unlike conventional cybersecurity systems that focus primarily on network-level threats, LLM security requires understanding linguistic intent.

Stage 3: Risk Scoring and Classification

Each prompt receives a security risk score based on multiple factors:

- Instruction conflict level
- Historical attack similarity
- Data sensitivity
- Requested action impact

A high-risk score triggers additional validation procedures before model execution.

Stage 4: Secure Model Interaction

Only validated prompts are forwarded to the LLM. During generation, the framework continuously monitors output behavior to detect unexpected responses.

This prevents situations where an apparently safe prompt produces harmful or unauthorized outputs due to contextual manipulation.

Stage 5: Feedback-Based Optimization

Security events are stored and analyzed to improve future detection capabilities.

The framework therefore creates a closed-loop defense mechanism where every interaction contributes to improving future protection.

3.4 Integration with Enterprise and Cloud-Based AI Systems

Modern LLM applications frequently operate through cloud platforms, APIs, and automated workflows. Therefore, the proposed framework is designed to integrate with existing computational environments.

Cloud-based AI systems require security mechanisms capable of handling distributed

processing, dynamic workloads, and multiple access points. Sayyed (2025) demonstrated that simulation-based testing approaches provide valuable methods for evaluating complex cloud interactions before deployment. Similar testing environments can be applied to evaluate prompt injection defense performance.

For enterprise automation environments, the framework supports secure AI integration by ensuring that generated outputs do not unintentionally modify business processes. This is particularly important in financial systems where AI-generated decisions may influence sensitive operations. Bhat and Krishnan (2025) emphasized that AI adoption in finance requires reliable and controlled technological frameworks.

3.5 Evaluation Strategy

The proposed framework can be evaluated through several performance indicators:

Security Detection Effectiveness

Measures the ability to identify malicious prompts accurately.

False Positive Reduction

Evaluates whether legitimate user requests are incorrectly blocked.

Response Integrity

Measures whether generated outputs remain aligned with intended instructions.

Adaptability Performance

Examines how effectively the framework responds to newly emerging attack patterns.

Operational Efficiency

Analyzes the computational overhead introduced by security mechanisms.

The evaluation approach follows the broader principle that intelligent systems must maintain both security and usability. Excessive restrictions reduce system usefulness, while insufficient protection creates unacceptable risks.

RESULTS

The conceptual analysis of the adaptive defense framework reveals several significant findings regarding secure LLM deployment.

First, the study identifies that prompt injection attacks cannot be effectively addressed through simple filtering mechanisms. Because these attacks exploit language interpretation and instruction hierarchy, security solutions must evaluate semantic relationships rather than only detect predefined malicious terms. The proposed multi-layer architecture provides improved resilience by combining detection, validation, enforcement, and continuous optimization.

Second, adaptive security mechanisms demonstrate stronger suitability for dynamic AI environments. Traditional security models often assume stable attack patterns, whereas LLM environments experience continuously changing user behaviors and adversarial strategies. The integration of continuous learning enables the defense system to evolve alongside emerging threats.

Third, the research indicates that secure LLM deployment requires integration between AI security and infrastructure reliability. Cloud execution models, automated workflows, and AI-powered applications increasingly operate together. Therefore, prompt security cannot be isolated from broader system architecture. Findings from cloud orchestration research support the importance of controlled execution and validation mechanisms (Sayyed, 2025).

Fourth, the framework highlights the importance of preventive security approaches. Similar to structured preventive protocols used in sensitive operational environments, proactive validation reduces the probability of system compromise before harmful outcomes occur (Rashi et al., 2025). This demonstrates that prevention-oriented AI security provides stronger protection than post-incident response mechanisms.

Finally, the analysis shows that adaptive defense architectures create a balance between security and

usability. Overly restrictive systems may reduce user productivity, while insufficient controls expose organizations to significant risks. The proposed framework addresses this challenge by applying context-aware decision mechanisms that determine whether prompts should be accepted, modified, or rejected.

The findings suggest that future LLM security solutions should move toward intelligent, self-improving defense ecosystems rather than static protection models.

DISCUSSION

The findings of this research demonstrate that prompt injection security in Large Language Models requires a fundamental shift from traditional defensive approaches toward adaptive, intelligence-driven protection mechanisms. The proposed framework positions LLM security as a continuous operational process rather than a fixed filtering activity. This perspective is particularly important because generative AI systems operate through complex interactions among users, models, external data sources, and integrated applications.

A major theoretical implication of the proposed framework is the extension of cybersecurity principles into the domain of language-based computing. Conventional cybersecurity models primarily focus on network protection, authentication, and software vulnerabilities. However, LLM environments introduce a new category of semantic vulnerabilities where attackers manipulate the meaning and interpretation of instructions. Therefore, security architectures must analyze not only what information enters the system but also how the model interprets and prioritizes that information.

The literature analysis supports the need for this transformation. Pai (2025) highlighted that prompt engineering significantly influences generative AI behavior, demonstrating the importance of instruction design. However, the same flexibility that enables effective prompt optimization also creates opportunities for malicious manipulation. The adaptive defense framework addresses this limitation by incorporating security validation into the prompt-processing lifecycle.

The integration of cloud computing principles further strengthens the proposed approach. Modern LLM deployments frequently depend on distributed cloud infrastructures, external APIs, and automated execution environments. Valiveti (2025) emphasized that evolving cloud service models require flexible execution architectures capable of managing complex workloads. Similar flexibility is required in AI security systems, where defense mechanisms must adjust according to application context and operational requirements.

The framework also provides practical implications for enterprise AI adoption. Organizations increasingly use generative AI for decision support, workflow automation, and knowledge management. In such environments, prompt injection attacks may produce consequences beyond incorrect text generation, including unauthorized workflow execution or exposure of confidential information. Research on AI-enabled financial innovation indicates that intelligent systems require strong governance and operational controls to ensure trustworthy implementation (Bhat & Krishnan, 2025).

Another important implication concerns the relationship between automation and security. As AI systems become more autonomous, security mechanisms must operate without significantly reducing system efficiency. Krishnan and Bhat (2025) demonstrated the value of combining generative AI with process automation for improving business workflows. However, increased automation also increases the importance of controlling AI-generated decisions. The proposed framework addresses this challenge by introducing adaptive policy enforcement capable of maintaining operational flexibility while reducing security risks.

The preventive security principle is another significant contribution of this study. Rashi et al. (2025) demonstrated that structured preventive protocols can reduce risks in sensitive operational environments. Although their research focuses on healthcare procedures, the underlying principle of proactive protection applies to AI security. Instead of responding after an LLM system has been compromised, adaptive defense mechanisms attempt to identify and prevent malicious behavior before execution.

Despite these advantages, several limitations exist. First, implementing a comprehensive adaptive defense framework may introduce additional computational overhead due to continuous analysis and monitoring. Organizations must balance security improvement with system performance requirements. Second, highly sophisticated prompt injection techniques may still challenge automated detection systems because attackers can exploit complex linguistic patterns and contextual ambiguity.

Another limitation involves the requirement for continuous model and policy updates. Since adversarial strategies evolve rapidly, defense systems require regular improvement and evaluation. Hebbar (2024) emphasized that resilient execution models depend on adaptability and responsiveness, supporting the need for continuous optimization in LLM security.

Future research should explore advanced adaptive mechanisms incorporating explainable AI, autonomous threat intelligence, and real-time security evaluation. Additionally, standardized benchmarks for evaluating prompt injection defense performance would support more consistent comparison among different security approaches.

Overall, the discussion indicates that secure LLM deployment requires an integrated approach combining AI reasoning capabilities with cybersecurity principles. The adaptive defense framework provides a foundation for developing more reliable, transparent, and resilient generative AI systems.

CONCLUSION

Large Language Models have introduced transformative capabilities across enterprise, cloud computing, financial technology, and intelligent automation domains. However, the growing dependence on these systems has exposed significant security challenges, particularly prompt injection attacks that exploit weaknesses in instruction interpretation and contextual processing.

This research proposed an Adaptive Defense Framework for Mitigating Prompt Injection Attacks in Large Language Models by integrating threat identification, contextual validation, adaptive policy enforcement, and continuous learning mechanisms. Unlike traditional security methods based on static filtering, the proposed framework emphasizes dynamic protection capable of responding to evolving attack patterns.

The study contributes to the field of AI security by establishing a structured approach for improving LLM reliability without significantly limiting usability. The framework demonstrates that effective protection requires cooperation between prompt engineering practices, cloud security concepts, automation resilience strategies, and preventive control mechanisms.

The analysis further highlights that LLM security should be considered a continuous lifecycle activity. Similar to resilient cloud and automated execution environments, AI systems require ongoing monitoring, evaluation, and optimization. Research contributions related to cloud orchestration, AI-driven automation, and preventive protocols provide valuable foundations for developing stronger AI defense mechanisms (Sayyed, 2025; Krishnan & Bhat, 2025; Rashi et al., 2025).

Future research should focus on implementing real-time adaptive defense systems, developing standardized evaluation datasets for prompt injection attacks, and integrating explainable security mechanisms that allow organizations to understand AI defense decisions. As generative AI continues to expand into critical applications, adaptive security architectures will become essential for achieving trustworthy and sustainable AI adoption.

REFERENCES

1. Sayyed, Z. (2025). Development of a Simulator to Mimic VMware vCloud Director (VCD) API Calls for Cloud Orchestration Testing. International Journal of Computational and Experimental Science and Engineering, 11(3). https://doi.org/10.22399/ijcesen.3480
2. Pai, D. (2025). Prompt Engineering Frameworks for Generative AI in Credit Analysis. https://doi.org/10.52783/jisem.v10i45s.9035
3. K. Bhat and G. Krishnan, "A Review of Artificial Intelligence in Finance: Driving the

Next Wave of Industry Innovation," 2025 3rd International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI), Coimbatore, India, 2025, pp. 2033-2040, doi: 10.1109/ICoICI65217.2025.11252894.

4. S. S. Sravanthi Valiveti, "Cloud Service Models and Execution Architectures: A Unified Survey of IaaS to Serverless Computing," 2025 5th International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT), MANDYA, India, 2025, pp. 1-7, doi: 10.1109/ICERECT65215.2025.11375903.
5. K. S. Hebbar, "Evolving High-Volume Systems: Reactive Execution Models for Resilient Operations," Computer Fraud and Security, vol. 2024, no.04, pp. 49-58, Apr. 2024 https://computerfraudsecurity.com/index.php/journal/article/view/906/638
6. Rashi, A. R., Gunukula, S., Anupriya, M., Jadhav, S. R., & Zarekar, M. (2025). Impact of Prophylactic Antibiotic Protocols in Cardiac Patients Undergoing Endodontic Surgery and Subsequent Prosthetic Treatment: A Comparative Study. Vascular and Endovascular Review, 8(14s), 168-174.