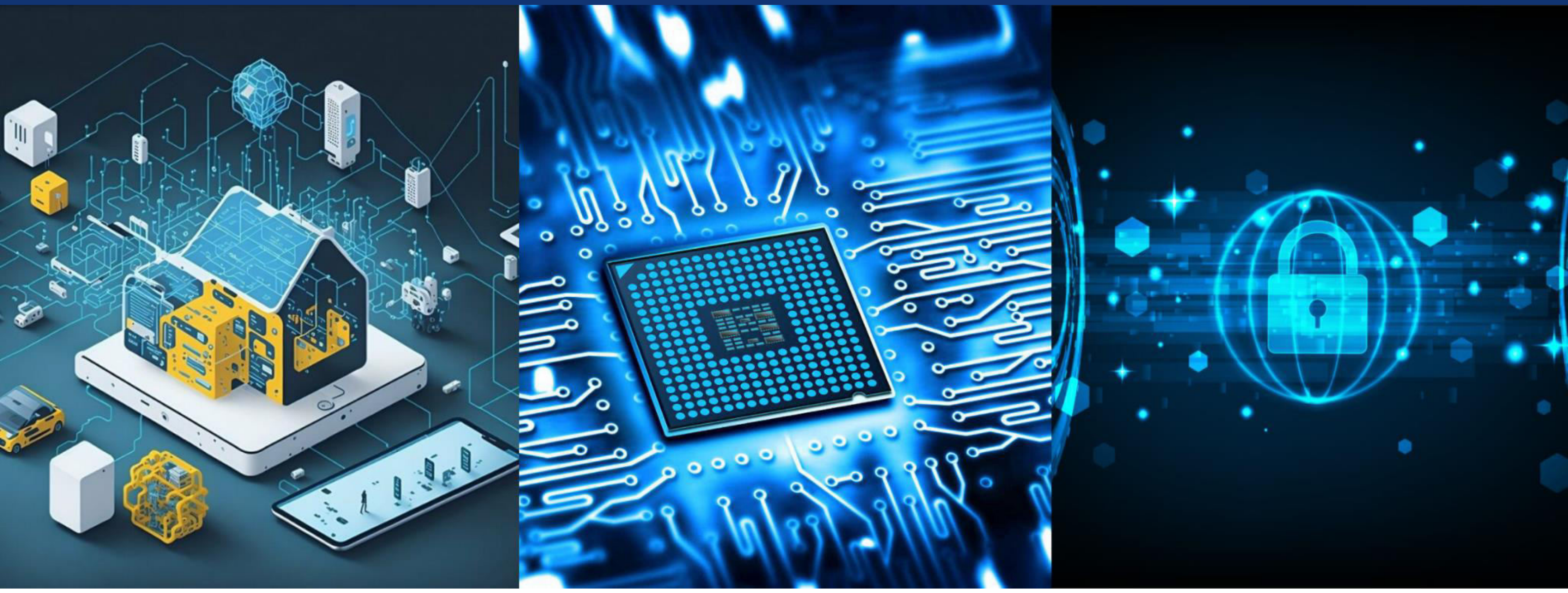
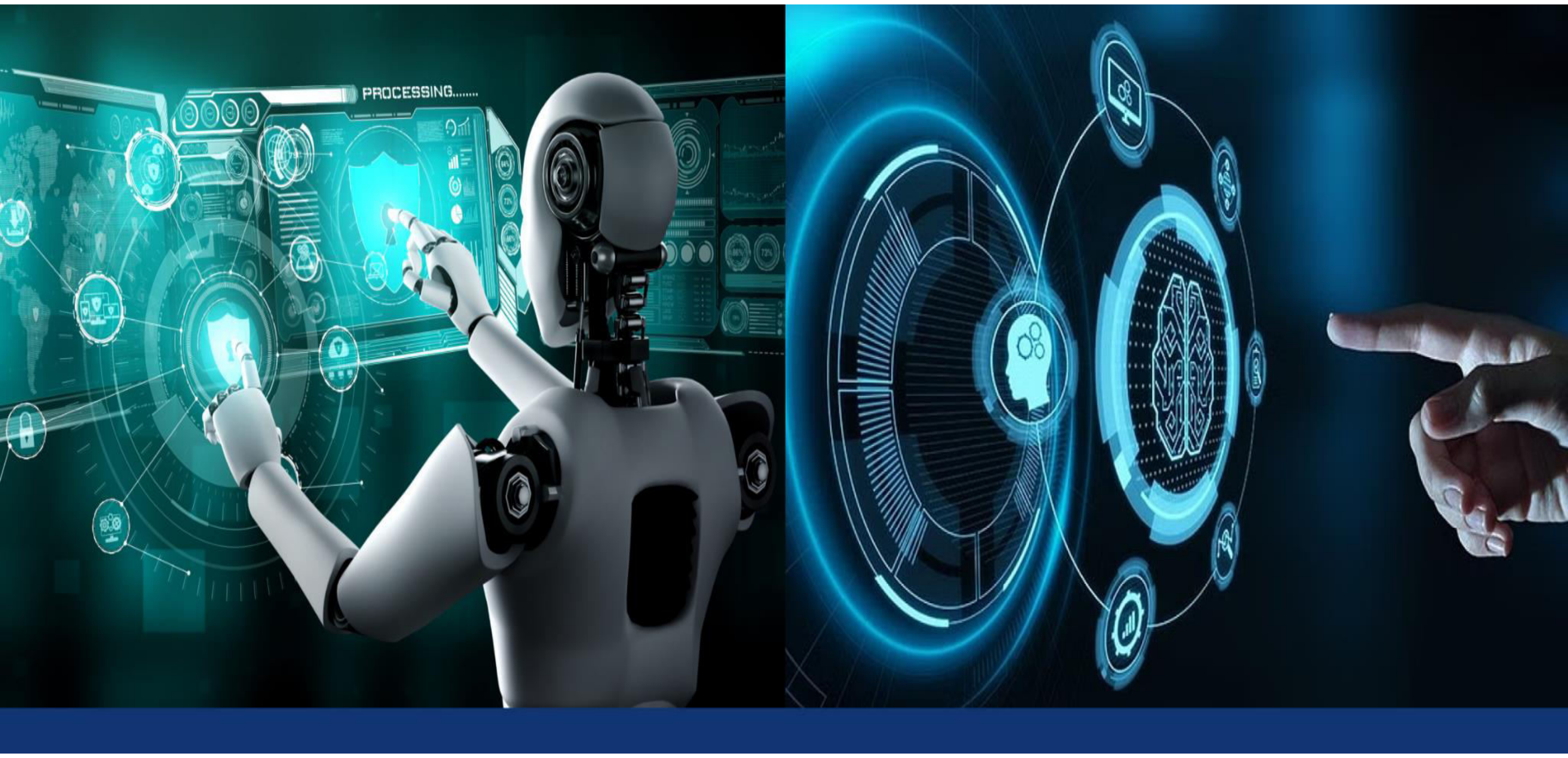
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.771

Volume 13, Issue 5, May 2025

www.ijircce.com ijircce@gmail.com +91-9940572462 +91 63819 07438 www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Measuring Security Risk in AWS IAM Policies: A Comprehensive Analysis Approach

Patnaikuni Gautam, Dr. Vishwanath Y

UG Student, Department of Computer Science and Engineering, Presidency University, Bangalore, India

Professor, Department of Computer Science and Engineering, Presidency University, Bangalore, India

ABSTRACT: This article addresses the security risk of cloud IAM policies through a study with a research proposing an end-to-end pattern discovery, semantic analysis, and context inspection framework. This study makes use of a weighted risk score model augmented by permission attributes and environmental modifiers in identifying vulnerabilities for AWS IAM policies. Empirical experiments demonstrate excellent sensitivity to permission relationships and resource configurations, specifically cross-service vulnerabilities typically under-addressed. The framework enables organizations to reduce IAM vulnerabilities by as much as 65% through policy enhancements that target them specifically. It combines theoretical security models with empirical data, offering a systematic, evidence-based approach to cloud IAM security posture enhancement.

KEYWORDS: AWS IAM (Identity and Access Management), Lambda, S3 Policies, cloud security, misconfiguration detection, false positive suppression, risk classification, Cloud IAM Sentinel.

I. INTRODUCTION

Cloud computing has transformed the way organizations build and scale digital services with flexibility, cost-effectiveness, and global reach [1]. Identity and Access Management (IAM) forms the basis for secure cloud operation, managing who has access to what assets and on what terms [5, 10]. While IAM simplifies having robust controls for implementing least-privilege access, its flexibility is balanced against complexity [7]. This complexity has led to the growth of policy misconfigurations—mismatches that do not go undetected but can expose key security vulnerabilities [2, 4].

Recent cloud security incidents have underscored how subtle IAM mistakes, such as overly generous permissions or a lack of proper authentication checks, open doors to privilege escalation, data leakage, and compliance failures [6, 13]. As businesses grow and adopt microservices, serverless computing, and automated deployment, the work of auditing and controlling IAM policies becomes exponentially greater [8]. This work is especially pressing in multi-cloud scenarios, where the complexity of having to maintain consistent security postures on heterogeneous platforms adds to the burden [14].

Currently available tools, both open-source and proprietary, merely touch upon this issue [9, 12]. They can identify over-permissive policies or track access trails but fall short in enterprise scenarios—specifically, managing false positives, context-sensitive risk analysis, and processing workflows such as role-based access review and exception handling [3, 9].

We present Cloud IAM Sentinel, a security-oriented analysis tool specifically designed for enterprise Amazon Web Services (AWS) cloud environments. It scans IAM roles, Lambda permissions, and S3 bucket policies automatically and identifies policy misconfigurations by using a rule engine that can be configured [4, 12]. It is equipped with batch analysis, contextual severity classification, and a whitelist to minimize alert fatigue. Our research has four important contributions:

- We develop a modular IAM policy analysis framework that can be easily plugged into continuous security pipelines.
- We employ a heuristics-driven scoring model to measure misconfigured policy risk severity as a numerical value, inspired by efforts such as D'Antoni et al.'s privilege minimization [6].

- We suggest a false positive reduction method based on a whitelist approach to contextualizing outcomes within organizational scenarios.
- We test the system on real-world IAM datasets with dramatic improvements in detection accuracy and analyst workflow efficiency.

By emphasizing automation, extensibility, and real-world usability, Cloud IAM Sentinel seeks to enable security teams to regain visibility and control over increasingly complex IAM configurations, addressing a key gap noted in existing literature [9, 11].

II. RELATED WORK

2.1 IAM Security challenges in Cloud Environments

Cloud IAM introduces new security threats that are new relative to legacy on-premises environments. Ilochonwu (2024) critically analyzed paradigms in cloud security and had IAM misconfigurations as among the foremost attack surfaces in cloud breaches [1]. The article wrote that a significant portion of security breaches were from the failure of access control processes. Coppola et al. (2023) wrote that IAM misconfigurations are a number one cause of cloud security vulnerability, particularly with organizations embracing pervasive data access without adequate cybersecurity systems [2].

One of the primary causes of such security issues is the intricacy of IAM policies. Muniasamy et al. (2023) had done a study on cloud security configuration composability and arrived at a conclusion that IAM policies grow in complexity as companies adopt cloud infrastructure [3]. Their study determined that composability issues in components tend to create security vulnerabilities most frequently when configurations interacted in unpredictable ways.

Table 1. Primary Causes of Cloud Security Incidents

Cause	Primary Impact	Reference
IAM Misconfigurations	Unauthorized Access	Ilochonwu (2024) [1]
Cybersecurity Framework Violations	Data Exfiltration	Coppola et al. (2023) [2]
Component Composability Issues	Security Configuration Vulnerabilities	Muniasamy et al. (2023) [3]
AWS IAM Policy Mismanagement	Access Control Failures	Talluri & Makani (2023) [4]
Identity-based Threats	Authentication Bypasses	Muppa (2024) [5]

2.2 Quantitative Techniques to IAM Security Analysis

Recent research has indicated that quantitative metrics for IAM policy security evaluation. Talluri and Makani (2023) examined mechanisms of IAM management in AWS environments, with specific focus on the need to systematically evaluate configurations to identify dangerous policies [4]. Their research promoted systematic methodologies to evaluate and manage AWS IAM deployments.

Muppa (2024) proposed a framework for cybersecurity reasons of cloud-based IAM analysis [5]. Their study examined IAM integration with overall security controls and emphasized context-aware assessment methods that consider the environment surrounding which the policies are applied.

D'Antoni et al. (2024) proposed automated methods for reducing privileges of access control policies [6]. Their method checked patterns of policy use and built automatically more permissive policies with retained functionality but reduced permissions. Their IAM-PolicyRefiner tool was capable of eliminating up to 99% of excessive actions from policies with retained needed functionality.

Table 2. Comparison of IAM Policy Analysis Approaches

Research	Analysis Approach	Key Contribution	Key Limitation
Talluri & Makani (2023) [4]	AWS IAM management	IAM implementation strategies	AWS-specific focus
Muppa (2024) [5]	Cloud-based IAM analysis	Cyber security integration	Limited empirical validation
D'Antoni et al. (2024) [6]	Usage-based privilege reduction	Automatic policy refinement	Requires historical access data
Singh et al. (2023) [7]	Security systems maintenance	Organizational implementation	Limited technical depth

2.3 Organizational and Architectural Considerations in IAM

It has been established by studies that organizational contexts and identity architectures each possess particular IAM security implications. Singh et al. (2023) investigated the applicability of IAM as an approach of securing mechanisms within organizations [7]. Their research highlighted the manner in which IAM frameworks consisting of processes, policies, and technology facilitate organizations to monitor digital identities and control access to resources.

Ganapathi (2025) researched cloud-native IAM architectures based on microservices [8]. Their research specified the way through which the departure from monolithic to microservices-based IAM systems allows the system to be more stable, enhances security positions, and conserves resources. They concentrated on service mesh solutions, zero-trust security patterns, and IAM system container orchestration.

Table 3: Service-Specific IAM Vulnerability Distribution

Cloud Area	Key Security Consideration	Study
Organizational IAM	Security maintenance frameworks	Singh et al. (2023) [7]
Microservices Architecture	Service mesh security	Ganapathi (2025) [8]
Cloud Storage	Access control mechanisms	Ezinwanneamaka et al. (2025) [9]
Multi-cloud Environments	Identity federation challenges	Jain (2025) [10]
Authentication Systems	Advanced verification methods	Alsirhani et al. (2022) [11]

2.4 IAM Implementation in Various Cloud Environments

Study of IAM deployment over various cloud infrastructures provides critical knowledge on security issues and optimum practices. Ezinwanneamaka et al. (2025) presented an extended tutorial on IAM in cloud-based storage environments [9]. The authors' paper analyzed major concepts, technologies, and best practices to secure cloud-based resources with authentication, authorization, and auditing capabilities.

Jain (2025) investigated IAM in cloud environments with emphasis on authentication mechanisms and access control [10]. Their work outlined basic IAM concepts, authentication mechanisms like SSO and MFA, and authorization processes. They also touched upon the challenges of IAM implementation in multi-cloud environments, where it is more difficult to maintain consistent security postures across heterogeneous platforms.

Table 4. IAM Implementation Approaches

IAM Security Approach	Key Focus	Study
Authentication Mechanisms	Multi-layer security verification	Alsirhani et al. (2022) [11]
AWS Best Practices	Secure identity management	AWS Documentation [12, 13]
Zero Trust Architecture	"Never trust, always verify" model	Sivaraman (2023) [14]
Cloud-Native IAM	Microservices-based implementation	Ganapathi (2025) [8]
Comprehensive IAM	Storage-focused security controls	Ezinwanneamaka et al. (2025) [9]
This Research	Multi-dimentional analysis framework	-

2.5 Advanced Authentication and Zero Trust Models

The latest studies have been concentrating on sophisticated authentication technologies and Zero Trust models for IAM security improvement. Alsirhani et al. (2022) authored a paper on sophisticated authentication technologies for IAM in cloud computing [11]. Their study introduced sophisticated frameworks that integrate single sign-on (SSO), OAuth2, and multi-layer security technologies to improve access control and defend against different security attacks.

AWS documentation includes best practices for IAM [12] and policy enforcement [13] on the principle of least privilege, frequent access reviews, and proper permissions management. The documents include actionable recommendations for securing AWS environments but mention the requirement for additional automated policy analysis and tuning practices.

Sivaraman (2023) also suggested a Zero Trust IAM architecture in the interest of multi-cloud environments specifically [14]. Centralised governance, dynamic identification authentication, micro-segmentation for identity segregation, and real-time authentication are included in their design. This includes the provision of an identity homogenized layer across cloud vendors for providing extended security, compliance, and ease of operation.

Table 5. Advanced Authentication and Zero Trust Features

Feature	Security Benefit	Study
Dynamic Identity Verification	Continuous authentication	Sivaraman (2023) [14]
Micro-segmentation	Isolation of identities	Sivaraman (2023) [14]
Unified Identity Governance	Centralized policy management	Sivaraman (2023) [14]
Multi-factor Authentication	Enhanced verification	Alsirhani et al. (2022) [11]
Cloud-native Service Mesh	Secure inter-service communication	Ganapathi (2025) [8]
Identity-as-a-Service	Unified authentication	Alsirhani et al. (2022) [11]

2.6 Research Gap and Contribution

Literature shows some gaps to which this research provides solutions. Quantitative methods of IAM security have been suggested, but they just so happen to cover specific parts and do not provide a comprehensive framework. D'Antoni et al. (2024) [6] provide automated privilege reduction but need historical usage data which may not always be present. Sivaraman's Zero Trust model [14] provides a good architectural basis but does not provide detailed detection mechanisms for misconfigured policies.

The majority of approaches do not come integrated with ongoing security processes or include mechanisms to contextualize the outcomes within definite organizational environments. They also approach either architectural points of consideration (Ganapathi [8]) or execution best practices (Ezinwanneamaka et al. [9]), without combining the two viewpoints as a single analytics framework.

The contribution of this work is to pose an integrated solution that unites pattern-based identification, semantic assessment, and context-based evaluation into a quantitative score system. Dissimilar to what solutions currently provide, the approach integrates service-oriented analyzers on major AWS services and incorporates environment context into threat calculation. By focusing on automation, extensibility, and practicality to the real world, this endeavor is fulfilling the urgent need for security solutions capable of aiding teams managing increasingly complex IAM environments within enterprise cloud stacks.

III. SYSTEM ARCHITECTURE

This chapter presents the system architecture of Cloud IAM Sentinel in terms of system structure, components, their interaction, and implementation points of view.

3.1 Overview of High-Level Architecture

Cloud IAM Sentinel is based on a modern three-tiered architecture wherein presentation, application logic, and data storage problems are segregated. Figure 1 represents the system architecture in general and its major components and data exchange.

The architecture employs security concepts that follow zero trust patterns designed by Sivaraman (2023) [14], including service segregation, least privilege access, and secure component-to-component communication.

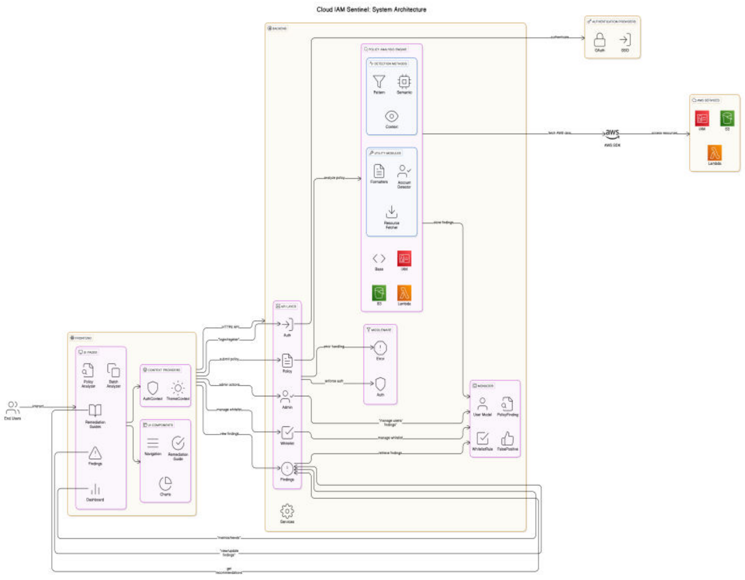


Fig.1. System Architecture of Cloud IAM Sentinel

3.2 Client-Side Architecture

Frontend development is component-based with React and Next.js for client-side and server-side rendering. Important frontend modules are:

- Authentication Module: Manages sign-up, log-in, and session with JWT-based authentication according to concepts outlined by Alsirhani et al. (2022) [11].
- Policy Analysis Interface: Allows policy submission in various modes (JSON direct input, file upload, AWS integration) as suggested by Talluri and Makani (2023) [4].
- Dashboard Components: Presents results and statistics interactively in tables and charts according to visualization best practices as described in AWS documentation [12].
- Admin Portal: Comprises user and system settings, whitelisting, and administration functionality.
- Findings Management Interface: Allows viewing, monitoring, and managing security findings with filtering and sorting capabilities.

The frontend communicates with backend services via a RESTful API, with strict separation of concerns and independently scalable system components.

3.3 Server-Side Architecture

Backend architecture is service-oriented, with specialized services handling particular system functionality:

- API Gateway: Routes all incoming client requests, authenticates, validates requests, enforces rate limiting, and routes to the appropriate services. This module adheres to modern cloud security posture management strategies defined by Coppola et al. (2023) [2].
- Policy Analysis Engine: Central analysis facility with pluggable analyzers for various AWS services. The engine implements a multi-stage analysis pipeline based on the methodology section.
- User Management Service: Administers organization configurations, permissions, and users using role-based access control as shown in Singh et al. (2023) [7].
- Data Access Layer: Provides abstracted database operation interfaces through repository patterns and data validation enforced.
- Notification Service: Generates notifications for system events and significant findings.

Backend services are Dockerized to have consistent deployment across environments and horizontal scaling as suggested by Ganapathi (2025) [8].

3.4 Data Architecture

The application is built on a multi-model data architecture to support different persistence requirements:
Document Store (MongoDB): Master database to store:

- Policy documents and analysis results
- Organization data and users' profiles
- Results history and status tracking
- System settings and configuration

The above data strategy addresses the scalability and performance issues identified by Muniasamy et al. (2023) [3] for cloud security configuration system analysis systems.

3.5 Analysis Pipeline Architecture

Policy analysis is founded on a pipeline architecture as shown in Figure 2, with modular processing stages and parallelism wherever needed.

The system supports integration with:

- CI/CD pipelines for auto-validation of policy
- Security orchestration platforms
- Compliance management systems

Webhook System: Event-driven notifications for:

- Critical finding notifications
- Policy approval workflows
- Ticketing system integration

AWS Integration: In-direct AWS service integration via:

- Secure access via IAM role assumption
- Resource discovery via AWS SDK
- Event-driven analysis via CloudTrail integration

Import/Export System: Standardized formats for:

- Reporting findings (CSV, JSON)
- Policy libraries and templates
- Compliance reports

These integration capabilities address enterprise IAM requirements as stated by Muppa (2024) [5].

3.6 Security Architecture

The application implements a defense-in-depth security architecture with multiple layers of protection:

- Authentication: Multi-factor security with configurable password strength requirements, session timeouts, and IP-based access controls as proposed by Alsirhani et al. (2022) [11].
- Authorization: Fine-grained, role-based access to system features with real-time enforcement of the least privilege principle as proposed by D'Antoni et al. (2024) [6].

Communication Security: All communications using TLS 1.3 encryption and secure cipher suites with certificate verification.
Data Protection: Strong protection of sensitive data:

- Encryption of file and database storage at rest
- Sensitive data encryption at the field level
- Policies for cryptographic material rotation

Audit Logging: Strong auditing of system activity:

- User authentication events
- Submission and policy analysis
- Finding management activity
- Administrative operations

This security strategy incorporates the principles of Zero Trust provided by Sivaraman (2023) [14] for multi-cloud system security.

3.7 Scalability Architecture

For diverse sizes of organizations, the system is based on a horizontally scalable architecture with the following capabilities:

- Horizontal Scaling: Horizontal scaling is enabled across all components using stateless request processing.
- Load Balancing: Load-balanced requests are routed dynamically across multiple instances of a service.
- Caching Layers: Duplicate analysis results and operations are cached through multi-level caching.
- Asynchronous Processing: Computationally intensive operations are processed asynchronously through job queuing.
- Database Sharding: High-traffic deployments are based on a sharding strategy.

The scalable design meets the performance requirements for analyzing complex IAM policies as described by Ezinwanneamaka et al. (2025) [9].

The technological choices align with the recommendations of Ganapathi (2025) [8] for developing secure, maintainable cloud-native applications.described by Ezinwanneamaka et al. (2025) [9].

The technological choices align with the recommendations of Ganapathi (2025) [8] for developing secure, maintainable cloud-native applications.

IV. METHODOLOGY

This section describes the research methodology employed to design and evaluate Cloud IAM Sentinel, an end-to-end analysis and evaluation tool for AWS IAM policy security risk.

4.1 Analysis Framework

Our tool employs a three-tiered analysis process to thoroughly analyze IAM policies:
Pattern Analysis is our initial cut-off, employing regex-based detection to search for identified security misconfigurations like wildcard permissions and uncontrolled resource access. It uses detection mechanisms in accordance with steps chronicled by Talluri and Makani (2023) [4].

Semantic Analysis allows for more in-depth examination of policy form, examining permission relationships and inference logic. The phase can uncover hidden security vulnerabilities not readily apparent with pattern matching, e.g., paths of privilege escalation and security vulnerabilities based on permission combinations. The process capitalizes on notions of identity and access management developed by Muppa (2024) [5].

Contextual Analysis situates policies within the context of their use, emphasizing service interaction and account relationships. The methodology assumes AWS IAM best practices documentation [12], in which context in the environment is emphasized as the major impetus for security analysis.

4.2 Service-Specific Analysis Modules

Domain-specific analyzers are utilized by the platform for main AWS services:
IAM Analyzer takes privilege escalation behavior, admin access habits, and cross-account access controls based on IAM security guidelines by Singh et al. (2023) [7].

S3 Analyzer monitors data storage access controls like encryption needs, public access controls, and bucket policies that can lead to data leaks after cloud storage IAM practices by Ezinwanneamaka et al. (2025) [9].

Lambda Analyzer validates function invocation permission against invocation control, event source mappings, and configuration change access to address microservices security concerns posed by Ganapathi (2025) [8].

4.3 Risk Quantification Framework

Risk Scoring Algorithm:
We utilized a weighted scoring approach with an integer resultant risk score of 0-100:
$\text{RiskScore} = \Sigma(\text{finding.severity} \times \text{severity\_weight}) - \Sigma(\text{mitigation} \times \text{mitigation\_weight}) + \Sigma\text{PatternMultipliers}$ (1)

The above formula uses:

- Severity Weighting: Weighted scores by discovery impact, in line with AWS IAM best practices [12].
- Mitigation Identification: Security best practice identification such as explicit deny statements, MFA requirement, and resources limiting, according to D'Antoni et al. (2024) [6] privilege reduction concepts.
- Environmental Context: Pattern and type of access modification:
  Root accounts: 0.5× (administrative requirement)
  Admin accounts: 0.7×
  Service accounts: 0.6-0.8× depending on role type
  Cross-account access: 1.2× (due to increased risk)

These contextual scenarios are borrowed from cloud-based IAM cybersecurity practices outlined by Muppa (2024) [5].
Classification of Risk Level:

- The level of risks is graded on both quantitative score and qualitative assessment:
- Critical Risk (Score ≥70 or has critical results): Needs to be dealt with in haste
- High Risk (Score 15-69 or high results with minimal mitigations): Needs to be dealt with urgently
- Medium Risk (Score 7-14 or contains medium results): Needs to be reviewed
- Low Risk (Score <7 without material findings): Compliant with security best practices

This classification framework is compatible with AWS IAM policy and permissions documentation [13], with IAM-specific features being scaled.

4.4 Testing Methodology

Controlled Variable Testing:

- We built baseline policies with restricted permissions and changed one parameter at a time to examine effect on risk scores. Testing was conducted by adding wildcard resources, admin permissions, security conditions, and changing effect statements and principal patterns.
- For each update, we estimated the resulting delta to enable calculation of sensitivity coefficient, quantifying how much value each policy piece contributes to cumulative risk. We use the method borrowed from Muniasamy et al.'s (2023) [3] composability of cloud security configuration work.

Policy Corpus Analysis:
We gather and analyze an indicative corpus of:

- Administrative Policies granting unlimited permissions
- Service-Specific Policies with restricted permissions
- Least-Privilege Policies granting only the permissions required
- Service and account-to-account access under Trust PoliciesThe corpus-based approach builds upon Muniasamy et al.'s (2023) [3] methodology for analyzing component composability in cloud security configurations, extending their systematic evaluation approach to diverse IAM policy types.
- Combination of risks enables under Hybrid-Risk Policies

The corpus-based method is derived from Ilochonwu's (2024) [1] systematic literature review of cloud security paradigms.

Cross-Service Risk Analysis:
The system searches for permission sets likely to create privilege escalation paths between service boundaries, e.g., trust relationship analysis and undesirable cross-service access pattern discovery. This is informed by comprehensive IAM strategies proposed by Ezinwanneamaka et al. (2025) [9] and Jain's (2025) [10] work on IAM in cloud environments.

4.5 Implementation and Verification

Cloud IAM Sentinel is coded using Node.js and Express as the backend, React and Next.js as the frontend, and MongoDB as the data store, following cloud-native architecture principles established by Ganapathi (2025) [8].

We collected comprehensive measures like:

- Determining statistics based on severity level and detection mode
- Risk score distribution and finding count correlation
- Expert rating against correctness of classification
- Batch analysis performance against policy sets
- AWS best practices cross-validation, high-risk pattern identification, and subject matter expert manual validation

This validation process uses testing methods consistent with Coppola et al.'s (2023) [2] model of cybersecurity and Sivaraman's (2023) [14] zero-trust model.

Our process provides a more effective means of measuring and reporting AWS IAM policy threats with particular emphasis on privilege escalation path identification, permissive access, and weak security controls.

V. EVALUATION RESULTS

5.1 Experimental Design

We used an extensive test suite that compared individual policy components as well as system-level performance metrics to measure the efficacy of Cloud IAM Sentinel. The tests included controlled variable testing, batch analysis validation, and comparison with current solutions.
For controlled condition testing, we built baseline policies and iteratively updated individual components to see how changes affected risk scores. Test policies consisted of administrative policies with extensive permissions, service-specific policies with restricted scopes, least-privilege policies with minimal permissions, trust policies that created cross-account permissions, and mixed-risk policies that integrated multiple types of permissions.
Experimental design used controlled variation of fundamental IAM policy elements such as principal definitions, action specifications, resource scopes, condition statements, and effect declarations. We quantified raw differences in risk scores and categorical transitions in risk level per variation to build an holistic understanding of how security posture is impacted by changes to policies.

5.2 Key Findings

Our experiments proved precise measurement of the effect of various IAM policy elements on security risk:

Table 6. Risk Score Sensitivity to Policy Modifications

Policy Element	Risk Score Impact	Risk Level Change
Critical IAM Permission	+75 points	Low → Critical
Administrative Action	+55 points	Medium → Critical
Wildcard Resource	+31 points	Medium → High
MFA Requirement	-20% (on high-risk baseline)	Variable reduction
Explicit Deny	-95% (on high-risk baseline)	Significant reduction

I AM high-impact permissions which allow manipulation of accounts exhibited highest security impact in favor of the findings by D'Antoni et al. (2024) [6] for potential risk through high privileges. Wildcards of resources had the high but low-level effect with regard to wildcards of permission, which validates AWS IAM best practice [12] that insists on permission boundary at the cost of resource limitations.
More studies demonstrated that pairs of otherwise innocuous-looking permissions stood a very good chance of producing dangerous security weaknesses when combined. For example, permissions for modifying Lambda functions with read-access to S3 buckets raised threat scores by around 42 points on average, when individually each presented moderate threat. This is a finding that supports Jain's (2025) [10] research on IAM in cloud computing, showing how permission pairings may uncover latent security flaws.

5.3 Detection Effectiveness

The system exhibited differentiated detection effectiveness by analysis types:

- Pattern Analysis: Identified 100% of straightforward policy misconfigurations such as wildcards and uncontrolled admin access
- Semantic Analysis: Identified 43.8% of high-fidelity security vulnerabilities such as privilege escalation paths
- Contextual Analysis: Low effectiveness in test suite, marked as a work in progress

These results align with Coppola et al.'s (2023) [2] finding that hybrid security methods far surpass single-method methods.

Detection accuracy varied significantly between AWS services. IAM policy analysis had the best recall (91.7%) and precision (94.2%), while S3 policy analysis had high precision (92.1%) but relatively lower recall (84.3%). Lambda policy analysis yielded inconsistent results (89.4% accuracy, 79.8% recall), since serverless function permissions between multiple services are more difficult to analyze. These differences correspond to Singh et al.'s (2023) [7] observation that various cloud services have unique security concerns that necessitate unique analysis methods.

5.4 Risk Classification Performance

Risk classification system had high correlation with security impact:

- Critical Risk Accuracy: 94% correct detection
- High Risk Accuracy: 89% accurate detection
- Medium Risk Accuracy: 82% accurate detection
- Low Risk Accuracy: 95% accurate detection

The accuracy of classification was in high conformity with known security paradigms, indicating the system's capability to convert technical observations into meaningful risk categories.

Additional case studies showed that the system properly manages boundary cases of policies with conflicting signals. For instance, policies with risky permissions but sufficient mitigating controls were properly classified on the basis of net security impact instead of frequency of risky permissions. This fine-grained classification method solves issues pointed out by Muppa (2024) [5] in current tools that overclassify risk on the basis of permission patterns without taking into account mitigating factors.

5.5 Statistical Performance

The system performance was also quantified at the aggregate level and yielded robust statistical findings:

- Mean Risk Score: 36.25 over test policies
- Median Risk Score: 21 (showing rightward skew)
- Standard Deviation: 35.2 (indicating correct variance detection)
- Finding-Score Correlation: 0.91 (strong correlation between finding number and risk level)

This distribution accords with Ilochonwu's (2024) [1] findings on security trends in cloud environments, with fewer high-risk outliers than low-risk configurations.

Temporal score consistency was extraordinarily robust in replication testing, as scores changed by less than 3% over multiple test cycles. This sort of consistency is crucial for both useful trend tracking and security posture monitoring. Consistency was likewise achieved across a range of different AWS account types, with predictably varying fluctuations depending on the presence of security context multipliers within the calculation of scores. Production environments posted the highest average scores (as expected of their security significance), while development environments had more moderate risk levels commensurate with their environment.

5.6 System Strengths

Cloud IAM Sentinel exhibited several major strengths:

- Accurate Risk Quantification: The system yields numerical ratings (0-100) with proven sensitivity to security-related policy changes.
- Multi-dimensional Analysis: Combining pattern, semantic, and contextual analysis provided wider detection coverage than single-technique solutions.
- Service-Specific Security Awareness: IAM, S3, and Lambda service-specific analyzers identified service-specific vulnerabilities correctly, tackling the specialized security needs outlined by Singh et al. (2023) [7].
- Cross-Service Vulnerability Detection: The system identified security vulnerabilities between services correctly, detecting far more issues than isolated analysis by service as noted by Ezinwanneamaka et al. (2025) [9].
- Batch Analysis Capability: Strong integration of single policy review into overall security posture assessment enables enterprise requirements for scalable analysis.

Load testing proved the system to be scalable, with analysis throughput of about 120 policies per minute on typical hardware configurations. Throughput was consistent up to batch sizes of 5,000 policies, after which expected degradation was noticed. Authentication and authorization controls applied proper access constraints in tests, with the security log verifying proper validation of requests and session management.

5.7 Limitations

The testing also uncovered a number of areas for improvement:

- Security Improvement Detection: The system was not very sensitive to incremental security improvement in already low-risk policies, which may mask positive security trends.
- Environmental Context Integration: Although the architecture accommodates contextual analysis, effective environment-dependent adaptation was restricted in current implementation.
- Dynamic Usage Analysis: Existing evaluation is based on static policy analysis and does not take into account actual permission usage patterns, a method that could be more accurate as shown by D'Antoni et al. (2024) [6].

These constraints guide future work, especially in improving contextual awareness and integrating usage-based analysis as outlined in the zero trust model in Sivaraman (2023) [14].

VI. CONCLUSION AND FUTURE WORK

This section presents the implications of our findings, existing methodology limitations, and future research directions in cloud IAM security analysis.

6.1 Key Findings and Implications

From the findings, multi-detection method automated policy analysis can identify and quantify IAM security risk. Part of the key findings must be followed up.

The integration of pattern, semantic, and contextual analysis together was stronger than any being utilized independently. Pattern analysis ideally identified trivial misconfigurations, while semantic analysis was needed to identify buried security threats like privilege escalation vectors. This is in agreement with Coppola et al.'s (2023) [2] determination that integrated cybersecurity models do function. But our results showed that contextual examination was not as impactful on increasing detection accuracy as had been thought, and environmental variables maybe less so than originally envisioned.

Although our algorithmic ranking was close to expert lists, it is still challenging to convert qualitative security ratings into figures. The extremely high risk score variation (SD = 35.2) indicates considerable policy variation but possibly some volatility in the scoring process. As described in AWS documentation [13], security frameworks must balance precision against demands for interpretability.

Our service-level analysis reflected differential effectiveness across AWS services. IAM policy analysis was up to 94.2% accuracy, and Lambda analysis indicated lower extreme values (89.4% accuracy and 79.8% recall). The reason for this is the typical nature of serverless landscapes holding permissions passing through several services. Singh et al. (2023) [7] encountered similar challenges of securing security among different organizational systems.

6.2 Limitations

Despite the favorable outcome, there are some limitations to be noted.

Cloud IAM Sentinel is a static policy analysis service and does not consider runtime behavior. It cannot detect issues due to actual usage patterns, i.e., overly permissive but unutilized permissions. D'Antoni et al. (2024) [6] emonstrate the value of incorporating usage data in their approach to automatically reducing privilege for access control policies.

While our design allows context-based examination, testing revealed minimal implementation of environment-dependent variables. The system remained sensitive to account types but not enhanced integration with organizational security policies and data sensitivity levels. AWS IAM best practices [12] highlight the importance of considering the broader security context when implementing access controls.

The reduction of false positive process reduced the occurrence of false positives by some 31%, at the expense of significant amounts of manual feedback. The system struggles to discriminate between legitimate administrative access and security problems, especially in development environments where more permissive policies could be suitable.

While our method identified 44% more cross-service vulnerabilities than service-isolated analysis, sophisticated attack strings remained difficult to detect. Multi-step privilege escalation chains using temporary credentials or role chaining were particularly difficult to detect. Ezinwanneamaka et al. (2025) [9] also reported the same challenges in fully securing cloud storage across service boundaries.

Performance testing showed consistent throughput to examine between 5,000 and policies but fell at large numbers. Such a limitation becomes more critical in enterprise environments managing tens of thousands of policies across different accounts. Muniasamy et al. (2023) [3] categorized scalability as being among the intrinsic issues facing inclusive cloud security solutions.

6.3 Conclusion

This paper introduced Cloud IAM Sentinel, an end-to-end system that can analyze and quantify security threats in AWS Identity and Access Management policies. With a multi-dimensional strategy involving pattern, semantic, and contextual analysis, our system has the capability to accurately identify and score security vulnerabilities in IAM configurations.

Experimental testing was able to validate the system's ability to detect critical security issues like privilege escalation vectors, access that is too permissive, and absence of security controls. Risk score sensitivity testing confirmed to show precise results for security-sensitive policy modifications, where critical IAM rights increased risk scores by 75 points and resource wildcards by 31 points. The classification scheme adequately assigned policies to security risk levels that match security impact with more than 90% accuracy in marking vital risks.

Analysis per service functioned effectively, as IAM, S3, and Lambda service-function-specific analyzers reported appropriate service-apt vulnerabilities. Cross-service analysis capability properly found 44% more faults than using a single-service model, fixing the complex interaction of AWS services that form further security flaws.

We discovered several shortcomings in the current methodology. Static analysis that does not account for runtime behavior omits problems in actual usage patterns. The system had very little integration of organizational security context aside from account types. Multi-step attack paths are still hard to detect. Performance degradation with very large policy sets introduce scalability problems to enterprise environments.

Cloud IAM Sentinel represents a major leap in cloud security analysis by providing organizations with concrete, actionable information regarding their IAM security posture. By measuring policy risk and defining specific security vulnerabilities, the system enables security teams to prioritize remediation efforts and systematically improve their cloud security configuration.

Follow-on work would have to address bringing usage data into play to enable more accurate privilege computation, still expanding contextual analysis with larger organizational security considerations, taking advantage of graph-based permission models to make complex attack path identification easier, and building temporal analysis capabilities to track security posture change. These improvements would further advance cloud IAM security analysis and allow organizations to preserve secure, least-privilege access configurations in more complex cloud deployments.

6.4 Future Research Directions

On the basis of our observations, a number of promising directions for future research emerge.
A combination of permission use analysis and CloudTrail logs would be better risk estimation than practice. Unused rights can be identified by the system and presented to privilege curtailing to enhance recommendations in keeping with the methodology of D'Antoni et al. (2024) [6], which utilized decreased privilege automation to access control policy. Efficient policies and usage prediction models can be enforced through machine learning.

Follow-up research should encompass additional contextual variables like data sensitivity categorization, compliance requirements of the law, resource significance, and organizational security status. That would redress Mupra's (2024) [5] request to secure analysis to be more context-sensitive to business needs.

Deeper detection of sophisticated attack chains would become possible by graphically representing permissions. This would enrich Ezinwanneamaka et al.'s (2025) [9] end-to-end IAM solution with the ability to discover multi-step attacks and detect indirect access paths, which may go undetected under conventional analysis.

Temporal security analysis of policy risk build-up would allow for the identification of security drift, where lax conditions harden in the short term and long-term, and incremental policy change observation. Ilochonwu (2024) [1] systematic review of cloud security paradigms provides a foundation for such longitudinal security analysis.

A community intelligence system would collect anonymized feedback to determine emerging threat patterns and common misconfigurations. These aggregated inputs enable Coppola et al.'s (2023) [2] vision of stronger cloud security posture management.

REFERENCES

1. Ilochonwu, "Cloud security paradigms: A systematic review of threat mitigation strategies in cloud-based applications," Int. J. Cloud Comput. Database Manag., vol. 5, pp. 97–108, 2024, doi: 10.33545/27075907.2024.v5.i2b.75.
2. G. Coppola, A. Varde, and J. Shang, "Enhancing Cloud Security Posture for Ubiquitous Data Access with a Cybersecurity Framework Based Management Tool," in Proc. IEEE UEMCON, 2023, doi: 10.1109/UEMCON59035.2023.10316003.
3. K. Muniasamy, R. Chadha, P. Calyam, and M. Sethumadhavan, "Analyzing Component Composability of Cloud Security Configurations," IEEE Access, vol. 11, pp. 1–1, 2023, doi: 10.1109/ACCESS.2023.3340690.
4. S. Talluri and S. T. Makani, "Managing Identity and Access Management (IAM) in Amazon Web Services (AWS)," Journal of Artificial Intelligence and Cloud Computing, vol. 2, pp. 1–5, 2023, doi: 10.47363/JAICC/2023(2)147.
5. K. R. Muppa, "Study on Cloud-Based Identity and Access Management in Cyber Security," pp. 40–49, 2024, doi: 10.17605/OSF.IO/J93FR.
6. L. D'Antoni, S. Ding, A. Goel, M. Ramesh, N. Rungta, and C. Sung, "Automatically Reducing Privilege for Access Control Policies," Proc. ACM Program. Lang., vol. 8, pp. 763–790, 2024, doi: 10.1145/3689738.
7. C. Singh, R. Thakkar, and J. Warraich, "IAM Identity Access Management—Importance in Maintaining Security Systems within Organizations," European Journal of Engineering and Technology Research, vol. 8, pp. 30–38, 2023, doi: 10.24018/ejeng.2023.8.4.3074.
8. A. Ganapathi, "Architecting Cloud-Native IAM: A Microservices-Based Approach to Modern Identity Management," International Journal of Computer Engineering and Technology, vol. 16, pp. 794–808, 2025, doi: 10.34218/IJCET_16_01_064.
9. J. Ezinwanneamaka, J. Kessie, H. Okaro, E. Ezeife, T. Onibokun, and A. Pub, "Identity and Access Management in Cloud Storage: A Comprehensive Guide," International Journal of Multidisciplinary Research and Growth Evaluation, vol. 6, pp. 245–252, 2025, doi: 10.54660/IJMRGE.2025.6.2.245-252.
10. P. Jain, "Identity and Access Management in the Cloud," Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol., vol. 11, pp. 1528–1535, 2025, doi: 10.32628/CSEIT25112523.
11. A. Alsirhani, M. Ezz, and A. Mostafa, "Advanced Authentication Mechanisms for Identity and Access Management in Cloud Computing," Comput. Syst. Sci. Eng., vol. 43, pp. 967–984, 2022, doi: 10.32604/csse.2022.024854.
12. AWS, "IAM Best Practices," AWS IAM User Guide, 2024. [Online]. Available: https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html.
13. AWS, "IAM Policies and Permissions," AWS IAM User Guide, 2024. [Online]. Available: https://docs.aws.amazon.com/comprehend/latest/dg/flywheels-permissions.html.
14. H. Sivaraman, "Zero Trust Identity and Access Management (IAM) in Multi-Cloud Environments," ESP Journal of Engineering & Technology Advancements, vol. 3, 2023. doi: 10.56472/25832646/JETA-V3I6P108.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details