Manejo de políticas para la evaluación eficiente de solicitudes en un modelo de control de acceso basado en atributos

R.A.Ibarra-García
CINVESTAV Unidad Guadalajara
Zapopan 45019, México
ricardo.ibarra@cinvestav.mx

A.Díaz-Pérez
CINVESTAV Unidad Guadalajara
Zapopan 45019, México
adiaz@cinvestav.mx

J.L.González-Compeán
CINVESTAV Unidad Tamaulipas
Ciudad Victoria 87130, México
joseluis.gonzalez@cinvestav.mx

***Resumen*—El control de acceso en sistemas distribuidos, especialmente en el contexto de la Industria 5.0, requiere mecanismos eficientes y adaptables para garantizar que solo los usuarios autorizados puedan interactuar con recursos críticos bajo condiciones específicas. Sin embargo, los enfoques tradicionales como el control de acceso basado en roles o las listas de control de acceso tienden a ser limitados en expresividad y poco eficientes al evaluar grandes volúmenes de políticas contextuales, dificultando su escalabilidad y capacidad de adaptación. En este artículo se presenta un modelo dinámico de control de acceso basado en eventos, donde cada uno se define como una conjunción de atributos: sujeto, recurso, espacio, tiempo y acción. Se implementó un prototipo funcional del sistema y se validó la propuesta mediante un estudio de caso simulado, utilizando políticas construidas a partir del dataset *KDD Cup 1999*. Las pruebas preliminares mostraron un comportamiento consistente ante variaciones en los atributos de las solicitudes, demostrando así la aplicabilidad del modelo.**

***Palabras clave*—control de acceso, industria 5.0, sistemas distribuidos, políticas contextuales**

I. Introducción

En el contexto de la Industria 5.0 [1] y los sistemas distribuidos, el control de acceso cumple una función central en la gestión de recursos digitales sensibles frente a accesos no autorizados [2]. La complejidad de estos entornos, impulsada por el Internet de las Cosas (*IoT*), la inteligencia artificial y la interacción colaborativa entre humanos y máquinas, demanda mecanismos de control de acceso dinámicos, granulares y adaptables a condiciones contextuales cambiantes [3].

Los enfoques tradicionales como las Listas de Control de Acceso (*ACL's*) y el Control de Acceso Basado en Roles (*RBAC*) (ver Figura 1) presentan limitaciones para responder a escenarios dinámicos [4], [5].

Aunque han sido ampliamente utilizados por su facilidad administrativa, no permiten incorporar atributos cambiantes ni considerar múltiples dimensiones contextuales [6]. Esto resulta problemático cuando las decisiones de acceso deben tomarse en función de condiciones

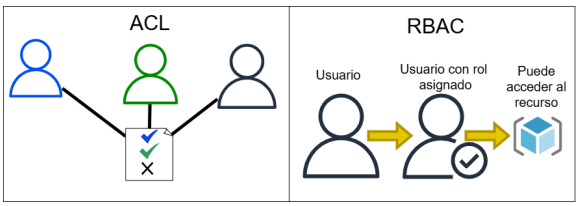

Figura 1. Modelos ACL y RBAC.

variables como la ubicación, el horario o el recurso [7]. Por ejemplo, un esquema RBAC tradicional puede otorgar acceso a un ingeniero de red para administrar servidores, pero no distingue si el intento ocurre fuera del horario laboral o desde una ubicación no autorizada.

En respuesta a estas limitaciones el Control de Acceso Basado en Atributos (*ABAC*) ofrece una alternativa flexible al considerar atributos del sujeto, recurso, acción y contexto espacio-temporal [8]. Se ha mostrado eficaz en entornos heterogéneos y dinámicos como infraestructuras críticas, sistemas de salud, ciudades inteligentes e industrias conectadas [9]–[11].

Sin embargo, el uso de *ABAC* no está libre de desafíos, especialmente en cuanto a la eficiencia de evaluación de políticas cuando el volumen de solicitudes y la cantidad de atributos involucrados crece.

En este artículo se presentan las siguientes contribuciones:

- Un modelo dinámico de control de acceso basado en eventos, en el cual cada solicitud se representa como una conjunción estructurada de atributos clave: sujeto, recurso, espacio, tiempo y acción.
- Diseño de una arquitectura modular para la evaluación eficiente de políticas de acceso y adaptabilidad del sistema frente a contextos variables.

Para validar esta propuesta, se desarrolló un prototipo funcional y se evaluó su comportamiento utilizando un estudio de caso basado en el conocido dataset de

intrusiones de red *KDD Cup 1999* [12], un benchmark clásico utilizado en la detección de intrusiones y la investigación en ciberseguridad.

El resto del artículo se desarrolla de la siguiente manera: En la Sección II se presentan los trabajos relacionados que abordan nuestro problema, en la Sección III se describe el modelo de control de acceso basado en atributos orientado a eventos, en la Sección IV se muestran los resultados preeliminares que se obtuvieron a partir del mapeo del dataset *KDD Cup*. En la Sección V presentamos las conclusiones obtenidas y finalmente en la Sección VI se presenta el trabajo futuro de la investigación.

II. Trabajo relacionado

En esta sección se presenta una revisión de trabajos representativos que abordan directamente el problema de la evaluación eficiente y el ordenamiento lógico o estructural de políticas ABAC. Estos criterios permiten contrastar nuestra propuesta con modelos estructurales, optimizaciones algorítmicas y enfoques descentralizados.

Yang *et al.* [13] introduce el modelo G_{ABAC}, una representación de políticas ABAC mediante grafos dirigidos que permiten evaluar accesos como flujos entre nodos (usuarios y recursos). Implementado en la base de datos *Neo4j*, este enfoque posibilita el uso de consultas tipo Cypher para analizar y aplicar políticas. Los autores reportan que, para conjuntos de hasta 10,000 nodos y 50,000 relaciones, la latencia promedio en la evaluación de políticas se mantuvo por debajo de los 20 ms, incluso en escenarios con múltiples condiciones contextuales. Además, se observó que el tiempo de análisis de políticas complejas se redujo en un 65 % respecto a sistemas tradicionales basados en árboles.

Liu *et al.* [14] proponen una técnica de recuperación de políticas que mejora la eficiencia en ambientes con miles de reglas y múltiples atributos. El método utiliza identificadores binarios para filtrar por atributo y un índice de profundidad para ordenar por valores. Las pruebas se realizaron con hasta 10,000 políticas y mostraron que el enfoque propuesto fue entre 3 y 5 veces más rápido que los métodos tradicionales de evaluación secuencial. Además, se mantuvo una precisión del 100 % en la recuperación de políticas válidas, demostrando que la optimización no comprometía la exactitud.

Además, Paul *et al.* [15] exploran la utilización de estructuras de datos de alta dimensión, en particular los árboles C-ND, para indexar eficientemente atributos continuos y discretos en políticas ABAC. La propuesta permite ejecutar búsquedas por similitud (útil en accesos de emergencia) de forma eficiente. En sus experimentos, realizados sobre datasets con más de 50 atributos por política, lograron reducir los tiempos de búsqueda en un 73 % comparado con enfoques sin indexación. Además, la tasa de decisiones correctas ante solicitudes ambiguas alcanzó el 96.8 %, mencionando que valida su uso en contextos de tolerancia a incertidumbre.

Así mismo, Hu *et al.* [16] proponen una arquitectura descentralizada para puntos de decisión de políticas ABAC utilizando blockchain, mediante un mecanismo optimizado llamado EBEC (Echo-Based Execution Conclude). Su enfoque reduce la redundancia en la evaluación y mejora la eficiencia mediante técnicas como Echo Compacting y balanceo de carga entre nodos. Las pruebas en infraestructura AWS demostraron una mejora del 35.6 % en el rendimiento comparado con otros enfoques blockchain, destacando su aplicabilidad en entornos distribuidos.

La Tabla I muestra una comparación cualitativa entre los trabajos seleccionados y la propuesta aquí desarrollada, utilizando cinco criterios para la evaluación eficiente en modelos ABAC. Como puede observarse, todos los enfoques revisados abordan la eficiencia en la evaluación, pero difieren en su estructura interna, capacidad para manejar atributos heterogéneos, soporte a solicitudes ambiguas y escalabilidad. En particular, solo Paul *et al.* considera solicitudes parcialmente coincidentes, mientras que la propuesta de este trabajo incorpora una representación estructurada y un mecanismo de afinidad para ordenar políticas y agilizar la evaluación. A diferencia de otros, el modelo propuesto cumple con los criterios establecidos y resulta eficaz en contextos dinámicos.

III. Modelo ABAC orientado a eventos

Este trabajo propone un modelo de control de acceso dinámico que organiza y evalúa políticas ABAC de forma eficiente mediante un proceso de agrupamiento semántico y evaluación orientada por similitud cuyo proceso se muestra en la Figura 2.

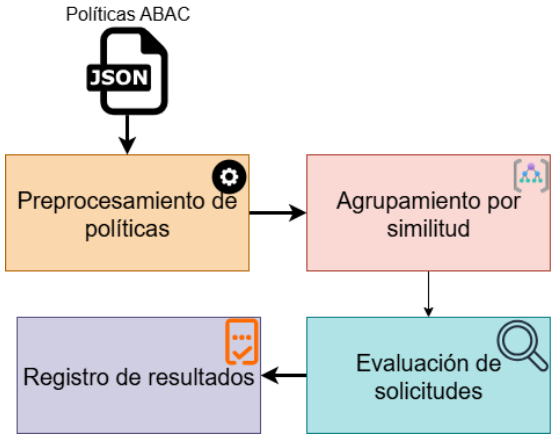

Figura 2. Diagrama conceptual del modelo propuesto.

Cuadro I
MATRIZ DE DOMINANCIA VISUAL: COMPARACIÓN DE TRABAJOS RELACIONADOS.

Leyenda: ✓ Cumple completamente ▲ Parcialmente cumple ✗ No cumple

Artículo	Evaluación eficiente	Modelo estructural	Atributos continuos/discretos	Solicitudes ambiguas	Escalabilidad
Yang *et al.* [13]	✓	✓	▲	✗	✓
Liu *et al.* [14]	✓	▲	✓	✗	✓
Paul *et al.* [15]	✓	▲	✓	✓	✓
Hu *et al.* [16]	✓	✓	✗	✗	✓
Propuesta	✓	✓	✓	✓	✓

La idea central consiste en representar las situaciones de acceso permitidas (eventos) como combinaciones específicas de atributos, y luego aplicar una técnica de agrupación para identificar patrones comunes en dichos eventos.

Un **evento** se define como una tupla de cinco atributos:

$$e = (A_0, A_1, A_2, A_3, A_4)$$

donde cada A_i representa un atributo distintivo del contexto de acceso.

- A_0 corresponde al *sujeto*, por ejemplo la identidad o rol del solicitante
- A_1 es el *recurso* u objeto al que se desea acceder
- A_2 se refiere a la *ubicación* del sujeto o del recurso
- A_3 es el *tiempo* o instante/intervalo temporal del acceso
- A_4 corresponde a la *acción* que el sujeto intenta realizar sobre el recurso.

Cada atributo A_i toma un valor de un dominio predefinido para ese tipo de atributo (por ejemplo, el atributo de sujeto podría tomar valores como `Estudiante`, `Profesor` o `Administrador`).

Una **política de acceso** P se define como un conjunto de eventos, es decir $P = \{e_1, e_2, \ldots, e_n\}$. Cada evento en la política representa una combinación particular de atributos que está permitida por dicha política. De este modo, una política especifica explícitamente los contextos o situaciones (definidos por atributos) en los cuales ciertas acciones están autorizadas.

Cada solicitud de acceso se representa como una tupla de atributos, y se evalúa respecto a un conjunto de políticas, cada una definida por combinaciones particulares de contexto (eventos autorizados). Para evitar evaluaciones exhaustivas sobre todo el conjunto de políticas, el modelo construye un espacio estructurado de eventos donde se identifican regiones de alta similitud contextual. Este agrupamiento se realiza en función de una métrica de similitud ponderada entre eventos, que considera el peso relativo de cada atributo:

$$\text{sim}(e_i, e_j) = \sum_{k=1}^{n} w_k \cdot \delta_k(e_i, e_j) \tag{1}$$

donde w_k representa la importancia relativa del atributo k y δ_k es una función indicadora de coincidencia para dicho atributo. Los pesos se derivan automáticamente del comportamiento estadístico de los datos, asignando mayor peso a atributos más informativos.

Sobre esta base, se genera una estructura de agrupamiento que permite segmentar las políticas en subconjuntos temáticamente coherentes. Al recibir una nueva solicitud, el sistema identifica el subconjunto más afín y evalúa las políticas en un orden descendente de afinidad, evitando la comparación con políticas irrelevantes.

Este mecanismo de organización y evaluación priorizada permite reducir el tiempo de respuesta en entornos dinámicos o distribuidos, donde las políticas pueden ser numerosas y contextuales.

IV. RESULTADOS PREELIMINARES

Para evaluar el comportamiento del modelo propuesto, se realizó un estudio de caso utilizando el dataset *KDD Cup 1999*. Este conjunto de datos, ampliamente utilizado en el campo de la detección de intrusos, contiene descripciones detalladas de conexiones de red simuladas en un entorno controlado.

Dado que los registros de solicitudes de acceso reales suelen estar restringidos por razones de privacidad y confidencialidad, se optó por adaptar el *KDD Cup* como una fuente alternativa. A través de un mapeo contextual, cada conexión registrada fue reinterpretada como un evento compatible con el modelo ABAC, utilizando atributos clave que reflejan el sujeto, el recurso, la ubicación, el tiempo y la acción realizada.

IV-A. Mapeo del dataset KDD Cup a eventos ABAC

El mapeo propuesto transforma los registros del dataset en eventos evaluables bajo el marco ABAC, como se resume en la Tabla II:

Este mapeo preserva los elementos clave de un evento de acceso: *¿Quién?*, *¿Qué?*, *¿Dónde?*, *¿Cuándo?* y

Cuadro II
MAPEO DE ATRIBUTOS ABAC A COLUMNAS DEL DATASET KDD CUP 1999.

Atributo ABAC	Columnas utilizadas	Interpretación
Sujeto (A0)	`logged_in`, `is_guest_login`	Representa al actor de la conexión. Clasificado como *Interno*, *Invitado* o *Externo* según tipo de login.
Recurso (A1)	`service`	Sistema o servicio accedido: ftp, http, smtp, telnet, etc.
Ubicación (A2)	`land`	Determina si el origen es local (land=1) o remoto (land=0).
Tiempo (A3)	`duration`	Duración de la conexión; interpretada como acceso corto, medio o largo.
Acción (A4)	`protocol_type`	Tipo de operación ejecutada, inferida a partir del protocolo (por ejemplo, telnet implica conexión remota).

Acción, derivando sujeto, recurso, acción, ubicación y tiempo de sus atributos originales. Así, el dataset se transforma en una bitácora de solicitudes compatible con políticas ABAC orientadas a eventos, facilitando la evaluación y el diseño de esquemas de control de acceso sobre datos reales o simulados.

IV-B. Evaluación experimental inicial

Se presentan los resultados experimentales iniciales de la exactitud y el rendimiento del sistema de control de acceso bajo diferentes cargas de trabajo. Se utilizó un conjunto fijo de 100 políticas, y se variaron el número de solicitudes evaluadas entre 30 y 300,000 para analizar el comportamiento en términos de eficiencia.

Experimento 1: Exactitud en la evaluación de acceso

En este experimento se evaluó la capacidad del sistema para determinar correctamente las decisiones de acceso (`permit` o `deny`) bajo diferentes condiciones: coincidencias exactas, uso de comodines ($\star$) y atributos categóricos como duración. La evaluación se realizó sobre un total de 2039 solicitudes. Las Tablas III y IV muestra los resultados de exactitud obtenidos.

Cuadro III
MÉTRICAS PRINCIPALES OBTENIDAS

Métrica	Valor
Exactitud (Accuracy)	1.0
Precisión (permit)	1.0
Cobertura (Recall, permit)	1.0
F1 Score (permit)	1.0
Precisión (deny)	1.0
Cobertura (Recall, deny)	1.0
F1 Score (deny)	1.0

El sistema mostró un desempeño perfecto en la evaluación de acceso, con decisiones totalmente consistentes respecto a las políticas establecidas. Se alcanzó una exactitud, precisión y cobertura del 100 %, sin falsos positivos ni falsos negativos. Este resultado era esperado, ya que el experimento fue diseñado para validar el mecanismo de evaluación en condiciones controladas: se utilizaron políticas y solicitudes predefinidas, cuidadosamente seleccionadas para representar situaciones bien delimitadas, incluyendo coincidencia exacta, uso de comodines y atributos contextuales. La ausencia de errores refleja la correcta implementación del modelo y su capacidad para operar conforme a las reglas establecidas, más que un comportamiento generalizable a entornos abiertos.

Cuadro IV
MATRIZ DE CONFUSIÓN

	Predicho: permit	**Predicho: deny**
Esperado: permit	316	0
Esperado: deny	0	1723

Experimento 2: Tiempo de evaluación

La Tabla V muestra los resultados obtenidos (en segundos como unidad de tiempo) para distintos tamaños de lote de solicitudes, indicando el tiempo total de evaluación (TT), el tiempo promedio por solicitud (TP/S) y el *throughput* (solicitudes por segundo).

Cuadro V
RESULTADOS DE RENDIMIENTO DEL SISTEMA DE EVALUACIÓN DE ACCESO

Nº Solicitudes	**TT**	**TP/S**	**Throughput (sol/s)**
30	0.030641	0.000939	984
300	0.247048	0.000735	1214
3,000	0.785019	0.000190	3824
30,000	7.404685	0.000240	4052
300,000	72.694182	0.000235	4128

La Figura 3 presenta el comportamiento del **tiempo promedio de evaluación por solicitud** conforme aumenta el número de solicitudes en el lote, usando una escala logarítmica en el eje X. Se observa una disminución y posterior estabilización del tiempo promedio conforme el volumen de solicitudes crece, alcanzando valores menores a 0.25 ms por solicitud incluso para cargas masivas.

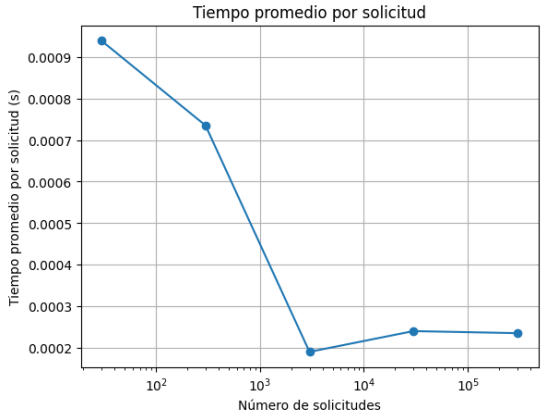

Figura 3. Tiempo promedio de evaluación por solicitud en función del número de solicitudes.

Por su parte, la Figura 4 muestra el **throughput** (solicitudes evaluadas por segundo), que aumenta rápidamente con el tamaño del lote y se estabiliza por encima de 4,000 solicitudes/segundo para cargas mayores a 3,000 solicitudes.

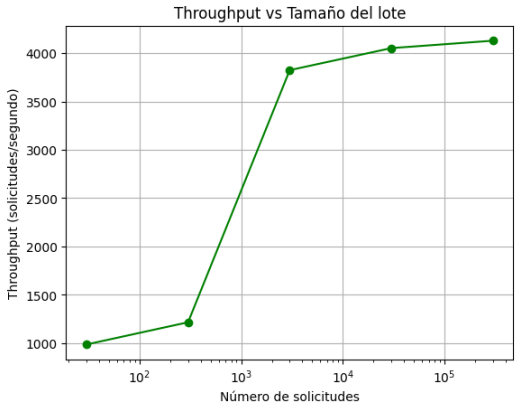

Figura 4. Throughput (solicitudes/seg) respecto al tamaño del lote.

Finalmente, la Figura 5 representa el **tiempo total de evaluación** en función del número de solicitudes, mostrando una relación lineal que evidencia la eficiencia constante por solicitud, incluso bajo cargas masivas.

Los resultados experimentales demuestran que el sistema de evaluación propuesto es eficiente y escalable.

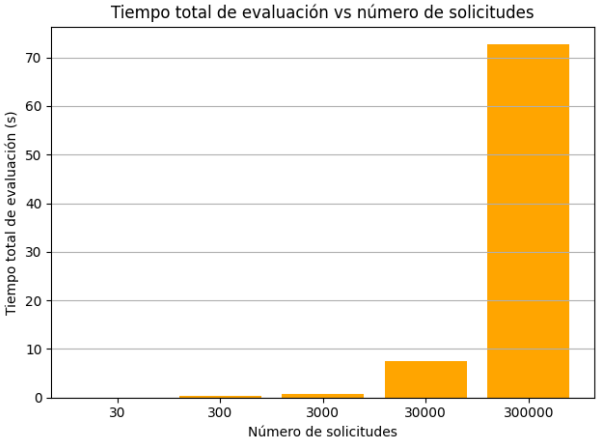

Figura 5. Tiempo total de evaluación en función del número de solicitudes evaluadas.

El tiempo promedio por solicitud se mantiene estable y bajo (menor a 0.25 ms) incluso ante cargas de hasta 300,000 solicitudes. El *throughput* alcanzado supera las 4,000 solicitudes por segundo, validando que el sistema es apto para su uso en escenarios de alta demanda y operación en tiempo real.

Experimento 3: Comparación de modelos

El objetivo de este experimento es demostrar la mejora del modelo propuesto frente al enfoque tradicional de evaluación. Para ello, se diseñó una prueba de escalabilidad basada en la variación del número de políticas (N), utilizando un conjunto fijo de 2000 solicitudes y evaluando cuatro tamaños diferentes de conjunto de políticas: 10, 30, 50 y 100.

En cada configuración, se seleccionó aleatoriamente un subconjunto de políticas del conjunto original, y se evaluaron las mismas 2000 solicitudes bajo dos enfoques: *evaluación tradicional*, que examina todas las políticas una por una hasta encontrar coincidencia, y el *modelo propuesto*, que organiza las políticas para reducir el espacio de búsqueda.

Los resultados experimentales, resumidos en la Tabla VI, muestran una mejora consistente del modelo propuesto en términos de tiempo de evaluación y eficiencia de búsqueda. Para todos los valores de N, el modelo propuesto logra un mejor desempeño que el enfoque tradicional.

Además del tiempo, el modelo propuesto examina menos políticas por evaluación. Con N=100, el enfoque tradicional analiza todas, mientras que el propuesto en promedio 24, reduciendo hasta un 76 % el espacio de búsqueda. No se observaron discrepancias en las decisiones de autorización o denegación, lo que valida su

Cuadro VI
COMPARATIVA ENTRE EVALUACIÓN TRADICIONAL Y MODELO PROPUESTO.

Métrica	**N=10**		**N=30**		**N=50**		**N=100**	
	Tradicional	**Propuesta**	**Tradicional**	**Propuesta**	**Tradicional**	**Propuesta**	**Tradicional**	**Propuesta**
Tiempo total [s]	0.100	0.054	0.218	0.085	0.386	0.125	0.676	0.202
Tiempo por solicitud [s]	0.000050	0.000027	0.000109	0.000043	0.000193	0.000062	0.000338	0.000101
Políticas escaneadas	∼10	∼3	∼30	∼8	∼50	∼12	∼100	∼24
Speedup	1.860x		2.550x		3.100x		3.360x	

equivalencia semántica respecto al enfoque tradicional, con mayor eficiencia.

V. CONCLUSIONES

Este trabajo presentó un modelo dinámico de control de acceso basado en eventos, donde las políticas se estructuran como combinaciones de atributos clave (sujeto, recurso, ubicación, tiempo y acción).

- Se desarrolló un mecanismo de organización y evaluación que reduce el espacio de búsqueda de políticas, mejorando la eficiencia sin sacrificar la exactitud.
- La validación mediante un prototipo y un estudio de caso con el dataset KDD Cup 1999 mostró que el modelo mantiene un rendimiento estable y escalable, lo que confirma su aplicabilidad en escenarios de seguridad donde se manejan grandes volúmenes de solicitudes.
- En conjunto, los resultados evidencian que la propuesta no solo es conceptualmente viable, sino también relevante en términos prácticos, al proporcionar un camino para implementar mecanismos de control de acceso más adaptativos frente a contextos cambiantes.

VI. TRABAJO FUTURO

Dado que se han obtenido resultados preliminares prometedores, nos planteamos el siguiente trabajo futuro:

- Evaluar el sistema con solicitudes generadas en tiempo real y comportamientos menos estructurados para analizar su capacidad ante patrones desconocidos.
- Incorporar atributos numéricos y condiciones temporales complejas que enriquezcan la expresividad de las políticas.
- Realizar comparaciones directas con enfoques existentes reportados en el estado del arte.

REFERENCIAS

[1] J. Leng, W. Sha, B. Wang, P. Zheng, C. Zhuang, Q. Liu, T. Wuest, D. Mourtzis, and L. Wang, "Industry 5.0: Prospect and retrospect," *Journal of Manufacturing Systems*, vol. 65, p. 279–295, Oct. 2022.

[2] M. A. Hassan, S. Zardari, M. U. Farooq, M. M. Alansari, and S. A. Nagro, "Systematic analysis of risks in industry 5.0 architecture," *Applied Sciences*, vol. 14, p. 1466, Feb. 2024.

[3] S. khan, T. Mazhar, T. Shahzad, M. Amir khan, A. U. Rehman, and H. Hamam, "Integration of smart grid with industry 5.0: Applications, challenges and solutions," *Measurement: Energy*, vol. 5, p. 100031, Mar. 2025.

[4] M. Alramadhan and K. Sha, "An overview of access control mechanisms for internet of things," in *2017 26th International Conference on Computer Communication and Networks (ICCCN)*, p. 1–6, IEEE, July 2017.

[5] G. Fragkos, J. Johnson, and E. E. Tsiropoulou, "Dynamic role-based access control policy for smart grid applications: An offline deep reinforcement learning approach," *IEEE Transactions on Human-Machine Systems*, vol. 52, p. 761–773, Aug. 2022.

[6] K. Soni and S. Kumar, "Comparison of rbac and abac security models for private cloud," in *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, p. 584–587, IEEE, Feb. 2019.

[7] A. Suganthy and V. Prasanna Venkatesan, "An introspective study on dynamic role-centric rbac models," in *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)*, p. 1–6, IEEE, Mar. 2019.

[8] V. C. Hu, D. R. Kuhn, D. F. Ferraiolo, and J. Voas, "Attribute-based access control," *Computer*, vol. 48, p. 85–88, Feb. 2015.

[9] H. F. Atlam and Y. Yang, "Enhancing healthcare security: A unified rbac and abac risk-aware access control approach," *Future Internet*, vol. 17, p. 262, June 2025.

[10] Y. Zhang, M. Yutaka, M. Sasabe, and S. Kasahara, "Attribute-based access control for smart cities: A smart-contract-driven framework," *IEEE Internet of Things Journal*, vol. 8, p. 6372–6384, Apr. 2021.

[11] M. Cheminod, L. Durante, F. Valenza, and A. Valenzano, "Toward attribute-based access control policy in industrial networked systems," in *2018 14th IEEE International Workshop on Factory Communication Systems (WFCS)*, p. 1–9, IEEE, June 2018.

[12] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, p. 1–6, IEEE, July 2009.

[13] M. Yang, V. Atluri, S. Sural, and J. Vaidya, *A Graph-Based Framework for ABAC Policy Enforcement and Analysis*, p. 3–23. Springer Nature Switzerland, 2024.

[14] M. Liu, C. Yang, H. Li, and Y. Zhang, "An efficient attribute-based access control (abac) policy retrieval method based on attribute and value levels in multimedia networks," *Sensors*, vol. 20, p. 1741, Mar. 2020.

[15] P. Paul and S. Sural, "Towards efficient evaluation of abac policies using high-dimensional indexing techniques," in *2021 Third IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*, p. 243–251, IEEE, Dec. 2021.

[16] Q. Hu, M. Correia, and T. Jiang, "An efficient blockchain for decentralized abac policy decision point," *Future Generation Computer Systems*, vol. 166, p. 107732, May 2025.