RESEARCH ARTICLE

SAMRIDDHI Volume 11, Issue 2, 2019 Print ISSN: 2229-7111 Online ISSN: 2454-5767

Strengthening Hybrid SAP Enterprise Integration Through Federated Identity Assurance, Zero-Trust API Governance, and Resilient Data Protection Across SAP ECC and Cloud Environments

Rajendra Asuri

Senior SAP integration consultant

Abstract

Hybrid SAP integration requires a coherent security framework that protects enterprise communication between SAP ECC and SAP Cloud Platform while maintaining reliable access to business-critical information. This study examines the challenges of identity federation, API authorization, and data protection in distributed enterprise environments, with the objective of establishing an integrated approach to secure cross-platform connectivity. A mixed-method research design combines architectural analysis, security control assessment, and comparative evaluation of established authentication, authorization, encryption, and access management practices. The proposed framework integrates federated identity management, role-based access control, secure API gateways, encrypted data transmission, and centralized audit mechanisms to strengthen trust across system boundaries. The analysis identifies fragmented identity administration, inconsistent access policies, insufficient interface protection, and inadequate monitoring as significant sources of hybrid integration risk. A layered security model provides a structured means of reducing unauthorized access, improving traceability, and preserving data confidentiality and integrity during system interactions. The study further demonstrates the importance of aligning technical safeguards with enterprise governance, operational requirements, and established security principles. Its principal contribution lies in connecting identity assurance, interface security, and information protection within a unified architectural perspective rather than treating them as isolated controls. The findings offer practical guidance for SAP architects, security professionals, and enterprise technology researchers seeking dependable integration strategies. This approach supports scalable interoperability, strengthens organizational security governance, and establishes a foundation for evaluating secure hybrid SAP environments.

Keywords: SAP ECC, SAP Cloud Platform, Hybrid Integration, Identity Federation, Single Sign-On, API Security, OData Security, Data Protection, GDPR Compliance, Cloud Integration, Role-Based Access Control, OAuth 2.0, SAML 2.0, Secure Communication, API Gateway, Data Encryption, Access Governance, Enterprise Security Architecture, Authentication and Authorization, Audit Logging, Information Confidentiality

SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology (2019); DOI: 10.18090/samriddhi.v11i02.14

Introduction

Hybrid enterprise computing has become an important approach for organizations seeking to connect established business applications with cloud-based services while preserving operational continuity and information security. SAP Enterprise Central Component (SAP ECC) supports essential business functions, including finance, procurement, inventory management, human resources, and production planning. Integrating these applications with SAP Cloud Platform enables organizations to extend existing enterprise processes through cloud services, application interfaces, and

Corresponding Author: Rajendra Asuri, Senior SAP integration consultant

How to cite this article: Asuri, R. (2019). Strengthening Hybrid SAP Enterprise Integration Through Federated Identity Assurance, Zero-Trust API Governance, and Resilient Data Protection Across SAP ECC and Cloud Environments . *SAMRIDDHI : A Journal of Physical Sciences, Engineering and Technology,* 11(2), 191-208.

Source of support: Nil

Conflict of interest: None

© The Author(s). 2019 Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and non-commercial reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The Creative Commons Public Domain Dedication waiver (http://creativecommons.org/publicdomain/zero/1.0/) applies to the data made available in this article, unless otherwise stated.

distributed information exchange. However, this integration introduces additional security considerations involving identity management, communication channels, access permissions, and the protection of sensitive business data. A dependable integration architecture must therefore balance interoperability, performance, administrative control, and security assurance.

The integration of SAP ECC with SAP Cloud Platform creates a distributed environment in which applications, users, services, and authentication mechanisms operate across different administrative and technical boundaries. Traditional enterprise security arrangements frequently rely on established network perimeters, application-specific permissions, and separately administered authentication mechanisms. Such arrangements can become difficult to manage when business transactions cross between on-premises systems and cloud services. Differences in identity representations, session handling, authorization rules, and interface configurations may create inconsistencies that weaken the overall security posture. Consequently, hybrid integration requires coordinated security controls that extend beyond individual applications and address the relationships between connected systems.

Identity assurance represents a fundamental requirement for establishing trustworthy communication across this environment. Federated identity management enables participating systems to rely on a common identity provider or an agreed authentication relationship, reducing the need for independent authentication arrangements across every application. Single sign-on can improve usability and administrative efficiency, but its effectiveness depends on appropriate trust configuration, assertion validation, session management, and authorization enforcement. Incorrectly configured trust relationships or excessive user privileges may permit unauthorized access even when authentication mechanisms operate as intended. A comprehensive integration strategy must therefore distinguish successful identity verification from the separate requirement to authorize each requested business operation.

Application programming interfaces provide the principal communication mechanisms through which enterprise applications exchange information and invoke services. SAP integration scenarios may expose business functionality through web services, OData interfaces, and other application endpoints that require carefully controlled access. Inadequate authentication, weak authorization checks, insufficient input validation, and poorly managed credentials can expose business transactions to misuse. A zero-trust approach addresses these concerns by requiring explicit verification of access requests and by limiting permissions according to the intended operation. Within a hybrid SAP environment, this principle must be supported by suitable API gateway policies, secure transport protocols, token validation, service-level authorization, and continuous examination of relevant security events.

Data protection presents a further challenge because information may move between enterprise systems, integration services, and cloud applications during routine business operations. Financial records, employee information, supplier details, and customer data require safeguards against unauthorized disclosure, alteration, and inappropriate processing. Encryption in transit, appropriate access restrictions, controlled data disclosure, and reliable audit records provide complementary layers of protection. Organizations must also ensure that technical arrangements support applicable privacy obligations and established information security policies. Effective protection consequently depends not only on cryptographic mechanisms but also on data classification, retention practices, administrative accountability, and clearly defined responsibilities across the integrated environment.

Despite the availability of established identity, interface security, and information protection mechanisms, a significant architectural challenge remains in coordinating these controls across hybrid enterprise systems. Authentication may be administered centrally while application permissions remain distributed, and API gateway policies may operate independently of data classification or audit requirements. Such separation can create gaps between identity verification, service authorization, and information handling. The central research problem concerns how these security functions can be organized into a coherent framework that supports trustworthy communication between SAP ECC and SAP Cloud Platform without introducing unnecessary operational complexity. The motivation for this study is to address this architectural fragmentation through an integrated and systematically structured security approach.

The primary objective of this study is to develop a security architecture that combines federated identity assurance, zero-trust API governance, and resilient data protection for hybrid SAP enterprise integration. The research examines how identity federation and single sign-on can establish reliable authentication relationships, how API security controls can restrict access to authorized services, and how data protection mechanisms can preserve confidentiality and integrity during cross-platform communication. It also considers how role-based access control, secure transport, audit logging, and administrative governance can reinforce these mechanisms. The principal research questions are: how can identity assurance be maintained across connected SAP environments, which API governance controls are necessary to reduce unauthorized service access, and how can data protection requirements be incorporated into a unified integration architecture?

The significance of this study lies in its treatment of hybrid SAP security as an interconnected architectural responsibility rather than a collection of independent technical configurations. By examining the relationships among identity trust, API authorization, communication security, and data governance, the study establishes a



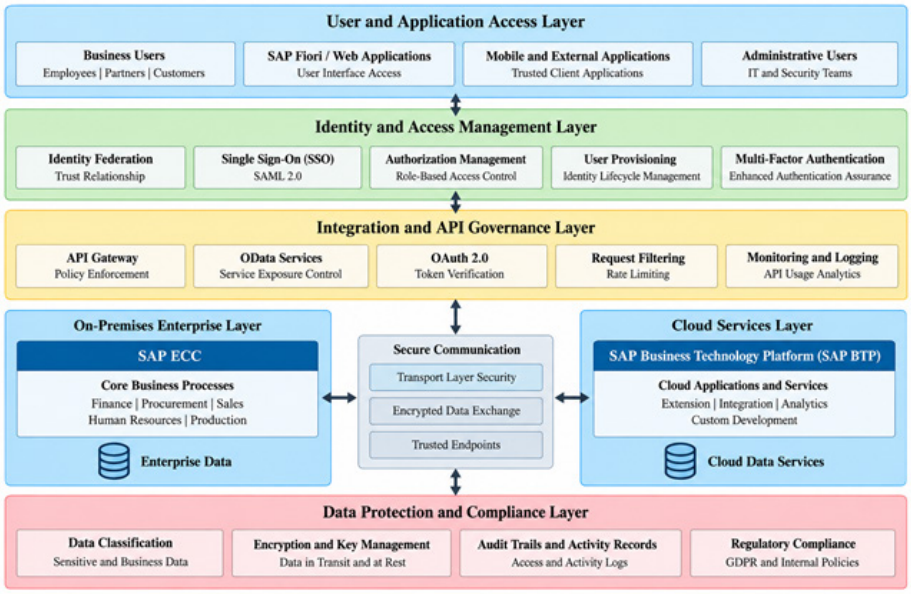

Figure 1: Hybrid SAP Integration Reference Architecture

structured basis for evaluating enterprise integration risks. Its practical contribution is intended to assist SAP architects, integration specialists, security administrators, and enterprise technology managers in identifying control dependencies and establishing consistent protection requirements. Its academic contribution lies in organizing established security principles into a framework that can support further investigation of hybrid enterprise integration. The proposed approach also provides a basis for future assessment through implementation studies, controlled security testing, and measurable evaluation of access control effectiveness, auditability, and data protection outcomes.

Research Background and Security Requirements

Traditional enterprise application integration architectures were developed to connect business systems through defined interfaces, middleware components, message exchanges, and application-specific communication protocols. In SAP environments, these arrangements support the exchange of business information between SAP ECC, external applications, databases, and enterprise services. Point-to-point connections can provide straightforward communication between a limited number of systems, but their administrative complexity increases as the number of interfaces and participating applications grows. Enterprise service buses and middleware platforms offer a more structured approach by centralizing message routing, transformation, and communication management. Nevertheless, centralized integration does not automatically guarantee consistent security, particularly when authentication, authorization, and data protection policies remain independently configured across connected systems.

Identity federation provides a mechanism through which independently administered applications can establish trust in identities verified by an authorized identity provider. This approach reduces the need to maintain separate authentication arrangements for every application and can simplify user access across on-premises and cloud environments. Single sign-on extends this capability by allowing users to access multiple participating applications through an established authentication session, subject to the security requirements of each application. In a hybrid SAP architecture, federation requires careful configuration of identity providers, service providers, trust relationships, session lifetimes, and user attribute mappings. Particular attention must be given to account provisioning and deprovisioning, because an identity that remains active after a change in employment or responsibility may retain access beyond its legitimate business purpose. Federation therefore improves authentication consistency only when supported by appropriate lifecycle management and authorization controls.

SAML 2.0 and OAuth 2.0 address related but distinct security requirements within distributed application environments. SAML 2.0 provides an XML-based framework for exchanging authentication and related identity information between trusted parties, making it suitable for browser-based single sign-on arrangements. Its security depends on correct assertion validation, signature verification, issuer identification, audience restrictions, and appropriate handling of assertion validity periods. OAuth 2.0, by contrast, establishes an authorization framework through which a client can obtain limited access to protected

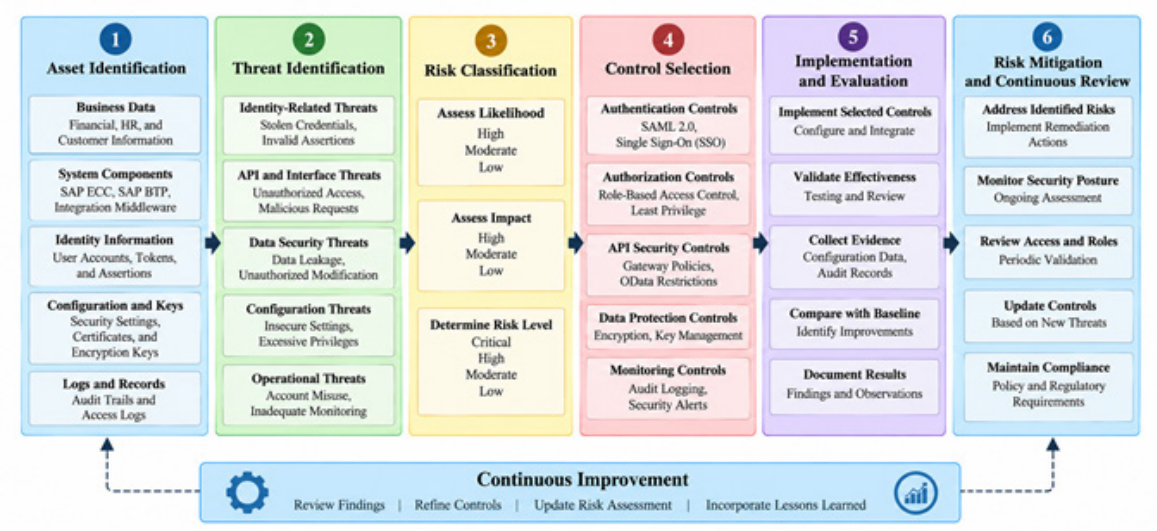

Figure 2: Threat Analysis and Risk Assessment Workflow

resources using access tokens. Its application to SAP integration requires suitable client authentication, restricted scopes, secure token handling, and validation by the receiving service. These mechanisms should not be treated as interchangeable: federation establishes trust in identity assertions, whereas OAuth 2.0 primarily governs delegated access to resources. Their coordinated use must preserve the distinction between authentication and authorization.

API gateways provide a control point for governing communication between enterprise applications and exposed services. They can enforce policies relating to client authentication, token validation, request filtering, rate limiting, transport security, and access monitoring. Such controls are particularly relevant when SAP ECC business functions are made available through integration middleware or cloud-hosted services. However, gateway enforcement alone cannot replace authorization checks within the underlying application. OData services may expose business entities and operations through standardized web interfaces, requiring appropriate restrictions on entity sets, query options, modification operations, and user permissions. Excessive data retrieval, unrestricted service operations, and insufficient validation of requests can increase the consequences of compromised credentials. A secure architecture must consequently combine gateway-level enforcement with service-specific authorization, controlled endpoint exposure, and systematic testing of interface configurations.

Role-based access control provides a structured method of assigning permissions according to organizational responsibilities and defined business functions. Within SAP ECC, roles and authorization objects can restrict access to transactions, data, and business operations, while connected cloud services may apply their own permission models. Differences between these arrangements can create authorization inconsistencies if user identities, role assignments, and business privileges are not mapped carefully. The principle of least privilege requires that users, applications, and service accounts receive only the permissions necessary to perform their authorized functions. Implementing this principle involves periodic access reviews, separation of duties, removal of obsolete permissions, and careful management of technical identities used by integration services. Effective governance must also account for changes in business responsibilities, emergency access, and service dependencies. Role design should therefore be evaluated across the complete transaction path rather than only within an individual SAP system.

Confidentiality, integrity, and availability provide the principal objectives of information security, while secure communication mechanisms help protect information exchanged between connected systems. Transport Layer Security can protect data in transit against interception and unauthorized modification when certificates, protocol settings, and endpoint verification are correctly configured. Encryption does not, however, prevent misuse by an authorized but overprivileged account, nor does it ensure that information is processed according to its intended purpose. Data protection consequently requires complementary controls, including access restrictions, appropriate encryption practices, credential protection, secure configuration management, and reliable backup and recovery arrangements. Integrity assurance also depends on validating messages, controlling modifications, and maintaining appropriate records of business transactions. In hybrid SAP environments, these safeguards must cover communication between on-premises applications, integration middleware, cloud endpoints, and supporting administrative services.

Privacy and audit obligations add organizational

requirements to the technical security architecture. The General Data Protection Regulation establishes principles concerning lawful processing, purpose limitation, data minimization, storage limitation, integrity, confidentiality, and accountability for personal data within its scope. SAP integration processes involving employee, customer, or supplier information must therefore be assessed according to the nature of the data, processing purposes, applicable legal responsibilities, and relevant organizational policies. Auditability requires sufficient records to reconstruct security-relevant activities, including authentication events, access decisions, administrative changes, and significant data operations. These records must be protected against unauthorized alteration and retained according to documented requirements. Excessive logging can itself create privacy and confidentiality risks, particularly when tokens, credentials, or unnecessary personal information are captured. An appropriate design must balance investigative value with data minimization, restricted log access, and controlled retention.

The preceding considerations reveal a need for an integrated set of security requirements that connects identity assurance, API governance, and data protection across the hybrid SAP environment. Established standards and security mechanisms provide important building blocks, but their effectiveness depends on consistent application across trust boundaries and administrative domains. The principal gap addressed by this study is the architectural separation that may arise when identity federation, service authorization, communication protection, and audit controls are designed or evaluated independently. The resulting requirements include explicit identity verification, strict validation of authentication assertions and access tokens, least-privilege authorization, controlled OData exposure, encrypted communication, traceable administrative activity, and accountable handling of sensitive data. These requirements establish the conceptual foundation for the methodology and evaluation framework developed in the following section. They also provide a basis for assessing whether an integrated security architecture offers more consistent protection than disconnected controls, without presuming measurable improvements before empirical evaluation.

Research Methodology and Evaluation Framework

The research methodology establishes a structured approach for examining security assurance across hybrid integration between SAP ECC and SAP Cloud Platform. The study adopts a design-oriented security assessment methodology that combines architectural analysis, threat modelling, control mapping, and comparative evaluation. This approach is appropriate because hybrid enterprise security depends on the coordinated operation of several mechanisms rather than on a single authentication or encryption technology. The assessment focuses on how identity federation, API governance, access control, secure communication, and data protection contribute to a consistent security posture. Established security principles and documented technical standards provide the basis for deriving evaluation criteria. The proposed framework is intended to support systematic assessment without presuming that architectural recommendations automatically produce measurable security improvements in a production environment.

The reference environment consists of an on-premises SAP ECC system connected to SAP Cloud Platform through defined integration interfaces and supporting communication services. SAP ECC represents the principal enterprise system responsible for maintaining business transactions and operational records, while the cloud environment provides application extensions, integration capabilities, and related services. An identity provider supports authentication and, where configured, federated access across participating applications. Integration middleware, API gateways, and web service endpoints provide controlled communication between the two environments. The model includes business users, administrators, technical accounts, external applications, and supporting data stores because each may introduce distinct access requirements and security exposures. The architecture is treated as a conceptual reference environment, not as a claim that every SAP implementation uses an identical topology or set of components.

Asset identification establishes the foundation for the security assessment. Critical assets include business records, personal information, authentication assertions, access tokens, encryption keys, application credentials, configuration files, and audit logs. The assessment distinguishes assets according to their confidentiality requirements, integrity dependencies, operational importance, and exposure to external communication. Trust boundaries are identified between user devices and application services, between identity providers and relying applications, between integration middleware and SAP ECC, and between cloud services and their supporting data resources. Communication paths are examined to determine where identity information is issued, where authorization decisions are enforced, and where business data is transmitted or modified. This mapping helps reveal indirect dependencies that may remain unnoticed when individual systems are reviewed separately. It also supports the identification of control responsibilities at each transition between administrative or technical domains.

Threat identification follows a structured examination of potential misuse, configuration weaknesses, and failures in security enforcement. Relevant scenarios include stolen credentials, forged or improperly validated authentication assertions, token disclosure, excessive application privileges, unauthorized OData operations, insecure endpoints, and insufficient audit coverage. Risks associated with

Table 1: Security Evaluation Criteria and Measurement Indicators

Evaluation criterion	*Measurement indicator*	*Assessment method*	*Expected evidence*
Identity federation	Validity of trust relationships and authentication assertions	Configuration review and assertion validation	Identity provider settings, assertion validation records
Single sign-on	Consistency of authentication across connected applications	Authentication workflow inspection	Sign-on configuration, session records
Access authorization	Compliance with role assignments and least-privilege principles	Role and permission review	Role mappings, authorization settings
API security	Effectiveness of token validation and endpoint restrictions	API configuration inspection and controlled access testing	Gateway policies, access test results
OData service protection	Restrictions on exposed entities and permitted operations	Service metadata review and authorization testing	Service definitions, request and response records
Secure communication	Appropriate TLS configuration and certificate validation	Protocol and certificate inspection	TLS settings, certificate records
Data confidentiality	Protection of sensitive information during transmission and access	Encryption and data-handling review	Encryption configuration, access policies
Data integrity	Prevention and detection of unauthorized data modification	Transaction control and log inspection	Change records, integrity verification results
Auditability	Completeness and traceability of security-relevant events	Audit configuration and record sampling	Authentication logs, access logs, administrative events
Risk management	Identification, prioritization, and treatment of security threats	Threat assessment and risk register review	Risk classifications, mitigation records

compromised technical accounts and inconsistent identity deprovisioning are also considered because automated integration processes may retain permissions independently of individual user sessions. Each threat is assessed in terms of its potential business impact, likelihood under the assumed operating conditions, and the effectiveness of existing safeguards. A qualitative risk classification distinguishes high, moderate, and low exposure, with higher priority assigned to threats capable of compromising sensitive information, privileged functions, or critical business transactions. These classifications guide the selection of controls but should not be interpreted as statistically validated incident probabilities.

The control framework is derived from established authentication, authorization, communication security, and information protection principles. SAML 2.0 is considered for federated authentication and single sign-on, with attention to assertion signatures, issuer validation, audience restrictions, and assertion lifetime. OAuth 2.0 is assessed as an authorization framework for controlled access to protected services, including appropriate token validation, scope restriction, client authentication, and secure token handling. Role-based access control and least-privilege permissions provide the basis for restricting access to SAP transactions, application functions, and service operations. API gateway policies support endpoint control, request filtering, rate limiting, and security monitoring, while OData-specific safeguards address the permitted resources and operations exposed through web services. Transport Layer Security, appropriate certificate management, and encryption practices support communication confidentiality and integrity. The selection reflects the distinct purpose of each control rather than assuming that any single mechanism can address every threat.

Evaluation criteria are organized into measurable or inspectable indicators that allow the reference architecture to be examined consistently. Identity assurance is assessed through the presence of validated trust relationships, correct assertion processing, controlled session handling, and documented identity lifecycle procedures. Authorization effectiveness is examined through permission restrictions, least-privilege assignments, and the ability to prevent unauthorized operations. API governance is evaluated through endpoint restrictions, token validation, request controls, and evidence of security monitoring. Data protection indicators include encrypted communication, controlled access to sensitive information, protection of cryptographic material, and appropriate audit coverage. Operational considerations include administrative complexity, compatibility with existing SAP configurations, and the availability of evidence needed for subsequent review.

Comparative assessment examines a baseline configuration in which security controls are distributed across individual systems and interfaces against a proposed integrated framework that coordinates identity assurance, API governance, and data protection. The comparison considers whether each scenario provides defined control ownership, consistent enforcement, traceable decisions, and documented protection of information across

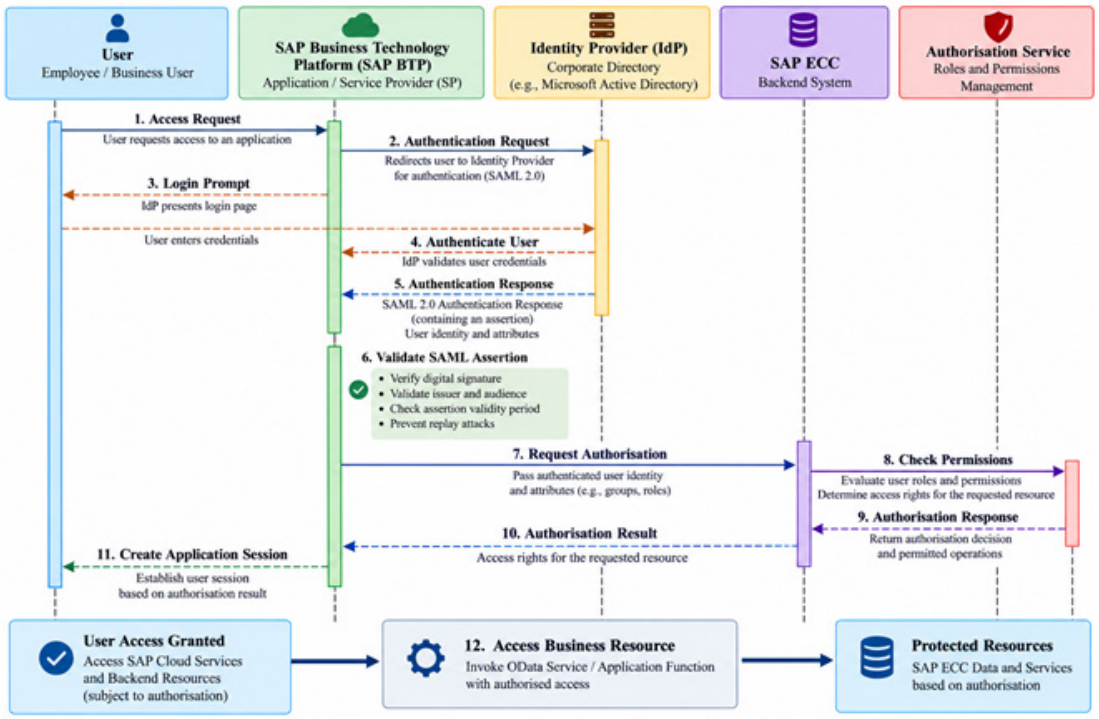

Figure 3: Federated Authentication and Authorization Sequence

trust boundaries. Evidence may be collected through configuration reviews, architecture documentation, controlled access tests, service request inspection, and audit record examination. Where implementation access is unavailable, the analysis must rely on documented standards and clearly identified architectural assumptions rather than simulated measurements presented as actual findings.

Validity and reproducibility depend on transparent assumptions, consistent evaluation procedures, and clear separation between proposed design principles and empirically demonstrated outcomes. The assessment assumes that participating systems support the relevant security mechanisms and that administrators can obtain sufficient configuration and audit evidence to examine their operation. Actual capabilities may differ according to SAP product versions, deployment models, licensed components, and organizational configurations. The study does not claim to have conducted production penetration testing, measured incident reduction, or verified quantitative performance gains unless such evidence is obtained independently. Potential limitations include the use of a conceptual reference environment, dependence on documented security practices, and variation in the interpretation of qualitative risk ratings. Reproducibility can be strengthened by documenting system assumptions, threat scenarios, control mappings, test procedures, and scoring rules. This methodology therefore establishes a defensible basis for assessing hybrid SAP security while identifying the empirical work required to validate the proposed framework in operational environments.

Federated Identity Assurance and Access Control

Identity federation establishes a foundation for secure interaction between SAP ECC and SAP Cloud Platform by enabling participating systems to recognize identities authenticated through an established trust relationship. In a hybrid enterprise environment, users may access business applications hosted on-premises while also interacting with cloud-based services that support integration, reporting, and application extensions. Maintaining separate authentication arrangements across these environments can increase administrative overhead and create inconsistencies in identity verification. A federated architecture addresses this challenge by defining how identity information is exchanged and accepted between trusted parties. However, federation should not be interpreted as unrestricted trust between connected systems. Each application must retain appropriate control over the resources and operations available to an authenticated identity. Effective identity assurance therefore depends on clearly defined trust boundaries, verified identity information, and consistent authorization requirements across the integration landscape.

Federated authentication and single sign-on provide complementary capabilities within this architecture. Federated authentication allows an application to rely

on an identity provider that has authenticated the user, while single sign-on reduces the need for repeated authentication when accessing participating applications under the applicable session and trust policies. In a typical browser-based workflow, a user requests access to a cloud application, the application initiates an authentication exchange, and the identity provider establishes the user's identity before returning an authentication response. The receiving application validates the response and establishes an application session if the required conditions are satisfied. This process can improve usability and reduce duplicated credential administration. Nevertheless, the security of the workflow depends on appropriate validation at every stage. An authenticated session must not automatically grant access to all available services, particularly where business functions involve sensitive financial, employee, or customer information.

Identity provider configuration determines whether the federation process can establish a reliable basis for authentication. The configuration must identify the participating parties, define the permitted authentication relationships, and specify the information required for successful validation. In SAML 2.0 deployments, important safeguards include verification of assertion signatures, validation of the issuer, enforcement of audience restrictions, and examination of assertion validity periods. Recipient and destination information must also be checked where applicable, and replay protection should prevent previously accepted assertions from being reused improperly. Trust metadata, signing certificates, and relevant configuration changes require controlled administration and periodic review. Certificate expiration, incorrect trust configuration, and inconsistent identity attributes can disrupt legitimate access or weaken security enforcement. A sound operational process should therefore include certificate lifecycle management, documented configuration ownership, and procedures for investigating failed authentication exchanges.

The reliability of federated identity also depends on mapping enterprise identities to the identities recognized by individual applications. SAP ECC and connected cloud services may represent users through different identifiers, account attributes, organizational assignments, and role structures. A consistent mapping process is necessary to prevent an authenticated person from being associated with an incorrect application account or receiving permissions intended for another user. Mapping rules should use stable identity attributes and should account for naming differences, duplicate accounts, organizational changes, and technical service identities. Unverified attribute matching can produce unauthorized account associations, particularly when multiple systems maintain partially overlapping user records. Identity reconciliation should consequently include defined matching rules, exception handling, and periodic review of unresolved discrepancies. This approach helps preserve accountability across the complete transaction path, from the original user identity to the application operation performed within SAP ECC or the cloud environment.

Authorization enforcement determines what an authenticated identity may do after successful authentication. Role-based access control provides a practical mechanism for associating users with permissions required for defined business responsibilities, while the principle of least privilege limits access to the minimum necessary functions and information. In SAP ECC, authorization objects and assigned roles can restrict access to transactions and business operations. Cloud applications and integration services may apply additional permissions to control service invocation, data access, and administrative functions. These authorization layers must be aligned without assuming that a role in one system automatically represents an equivalent privilege in another. A federated identity should therefore be evaluated against the authorization policies of the receiving application, including the particular operation requested and the sensitivity of the relevant resource. This distinction prevents authentication success from becoming a substitute for explicit access control.

Session management and identity lifecycle administration further influence the strength of the federated security model. Application sessions should have appropriate expiration policies, and session termination should be handled consistently with the capabilities of the participating systems. Sensitive operations may require renewed authentication or additional verification according to organizational policy. Identity lifecycle controls must cover account creation, changes in employment or business responsibilities, temporary access, and account deactivation. When a user leaves an organization or changes roles, obsolete permissions should be removed from connected systems within a defined administrative process. Similar attention is required for service accounts, which may continue to operate independently of individual employee sessions. Periodic access certification, controlled emergency access, and review of dormant accounts can reduce the accumulation of unnecessary privileges. Clear ownership of these activities helps ensure that federation remains aligned with current organizational responsibilities rather than historical access assignments.

Identity mismatches and excessive privileges represent important failure scenarios in hybrid SAP integration. A valid authentication response may still be associated with an incorrectly mapped account, while an accurately identified user may possess permissions that exceed legitimate business requirements. Additional risks arise when identity attributes are trusted without adequate validation, when administrative accounts share credentials, or when technical users retain broad permissions for convenience. These weaknesses can complicate incident investigation because activity may be attributed to a valid account even when the underlying operation was inappropriate. The assessment should therefore examine identity mapping exceptions, privileged



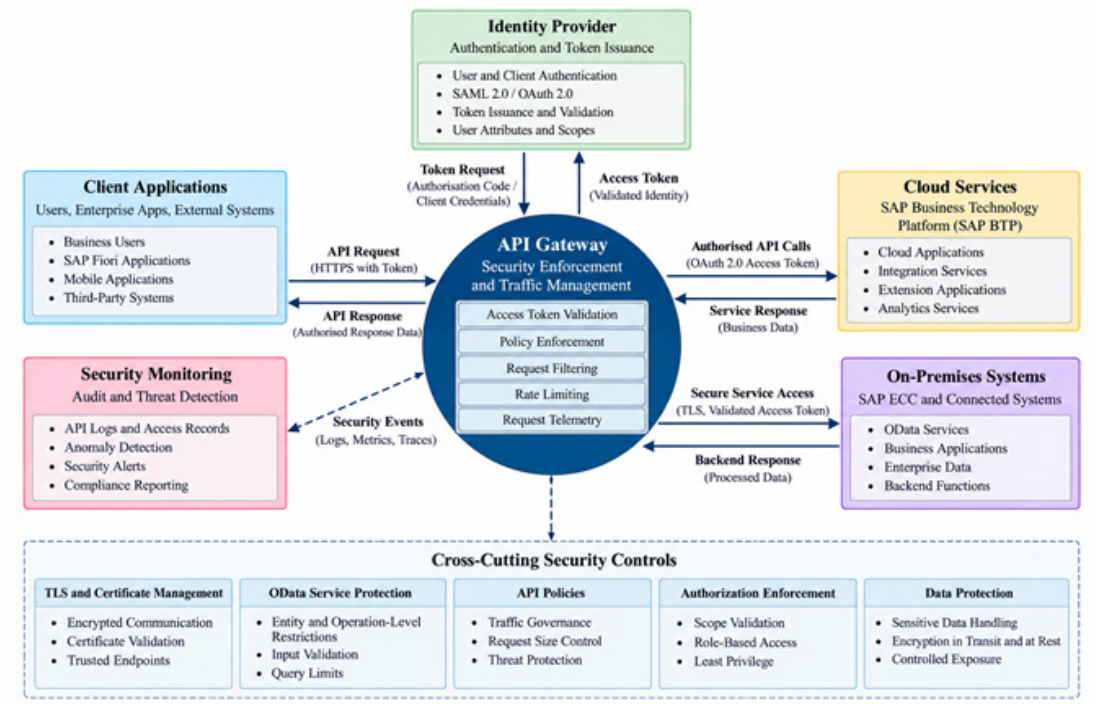

Figure 4: API Security Control Architecture

account assignments, stale permissions, failed authentication patterns, and inconsistencies between enterprise roles and application-level privileges. Audit records should provide sufficient information to connect authentication events with subsequent access decisions and relevant business operations, while avoiding unnecessary collection of sensitive information. Such measures strengthen both preventive controls and the ability to investigate suspicious activity.

A coordinated identity assurance framework should integrate federation configuration, identity reconciliation, authorization enforcement, session management, and lifecycle governance into a consistent administrative process. The proposed approach assigns explicit responsibilities to identity providers, relying applications, SAP authorization administrators, and integration service owners. Authentication assertions are validated before sessions are established, while application permissions remain subject to local authorization requirements and least-privilege restrictions. Access reviews and audit monitoring provide additional assurance that permissions remain appropriate as organizational conditions change. The figure should be placed after the discussion of federated authentication workflows, illustrating the sequence from user access request and identity provider authentication through assertion validation, application authorization, and resource access. The resulting architecture offers a structured basis for reducing identity-related exposure and improving accountability across hybrid SAP systems. Its effectiveness should ultimately be verified through configuration inspection, controlled access testing, and examination of identity lifecycle records in the intended deployment environment.

Zero-Trust API Governance and Secure Service Communication

Application programming interfaces form a critical communication layer between SAP ECC and SAP Cloud Platform, enabling enterprise applications to exchange business records, invoke services, and support integrated workflows. Although these interfaces improve interoperability, they also introduce security exposures when internal business functions become accessible through insufficiently controlled endpoints. Unrestricted service discovery, excessive data retrieval, weak client authentication, and inconsistent authorization can expose sensitive business operations to unauthorized parties. Hybrid deployments introduce additional complexity because requests may pass through several components, including integration middleware, API gateways, identity providers, and backend applications. Each component may apply different security policies, creating opportunities for inconsistent enforcement. A zero-trust approach addresses these challenges by requiring explicit verification of access requests, limiting permissions to defined business needs, and evaluating trust according to the relevant identity, resource, and operating context rather than assuming that a request is safe because it originates from an internal network.



Figure 5: Enterprise Data Protection Lifecycle

Gateway-mediated service access provides a structured mechanism for controlling communication across these boundaries. An API gateway can serve as an enforcement point for client authentication, access token inspection, request filtering, traffic management, and security logging. In a hybrid SAP architecture, gateway policies should define which applications may invoke particular services, which HTTP methods are permitted, and what conditions must be satisfied before requests reach protected resources. Endpoint exposure should be restricted to documented business requirements, while administrative and diagnostic interfaces should remain inaccessible to unauthorized clients. The gateway should also preserve relevant identity and request context when forwarding legitimate calls to backend services. Nevertheless, a gateway is not a substitute for backend authorization. SAP ECC and connected applications must independently enforce permissions appropriate to the requested business operation, ensuring that a weakness in an intermediary does not automatically expose all underlying resources.

OAuth 2.0 provides an authorization framework for granting clients controlled access to protected resources through access tokens. Its role in hybrid SAP integration must be distinguished from user authentication, since the presence of a valid token does not necessarily establish that a particular person has been authenticated for every business purpose. The receiving service must validate the token according to the applicable deployment architecture, including its issuer, intended audience, validity period, permitted scopes, and other required claims where relevant. Tokens should be transmitted through protected channels, stored securely, and prevented from appearing in application logs or diagnostic messages. Client credentials must also be protected, with privileges restricted to the functions required by the integration process. Where tokens are issued for specific services or audiences, their use should remain confined to those intended resources. These measures reduce the potential consequences of token theft, replay, unintended disclosure, or excessive authorization.

OData services require additional safeguards because they may expose business entities, relationships, and operations through standardized HTTP interfaces. A service that permits unrestricted entity access or unnecessarily broad query options can disclose more information than the calling application requires. Secure OData governance should define the entities and operations available to each authorized client, restrict modification functions where they are not necessary, and validate requests against the service contract. Input validation should account for malformed parameters, unsupported operations, unexpected data types, and requests designed to consume excessive processing resources. Query restrictions, pagination limits,

and response filtering can help reduce unnecessary data exposure and resource consumption. However, these controls must preserve legitimate business functionality and should not be applied indiscriminately where enterprise processes require broader queries. Effective service protection combines gateway policies with backend authorization, application-level validation, and periodic examination of exposed service metadata.

Transport Layer Security protects communication between participating endpoints by providing encryption and integrity protection, together with server authentication when certificates are correctly validated. In a hybrid SAP environment, secure transport must be maintained across all relevant connections, including user access, gateway communication, middleware links, identity exchanges, and backend service calls. Protecting only the external connection is insufficient if information subsequently travels across an inadequately protected internal segment. Certificate management should include controlled issuance, renewal, expiration monitoring, and timely replacement of compromised or obsolete certificates. Trust stores must be maintained carefully, and certificate validation should not be disabled merely to resolve connectivity problems. Where mutual TLS is appropriate and supported, client certificate authentication can provide an additional mechanism for establishing the identity of communicating systems. These protections must be complemented by secure credential handling and suitable endpoint configuration, because encrypted transport alone cannot prevent misuse by an authorized but compromised application.

API governance also requires policies that regulate traffic, detect anomalous behaviour, and maintain evidence of security-relevant activity. Rate limiting can restrict excessive request volumes and reduce the impact of accidental loops, abusive clients, or certain forms of resource exhaustion. Request size limits, timeout settings, concurrency restrictions, and controlled error responses provide additional safeguards for service availability. Monitoring should identify repeated authentication failures, unusual token usage, unexpected request patterns, and access attempts involving restricted resources. Security logs should capture relevant timestamps, client identifiers, target services, authorization outcomes, and correlation identifiers where appropriate. Sensitive information, access tokens, and credentials should be excluded or appropriately redacted. Centralized review of gateway and backend events can improve incident investigation, particularly when a single transaction crosses several integration components. Governance policies should define alert thresholds, escalation responsibilities, retention requirements, and procedures for reviewing exceptions.

Token misuse and unauthorized service access illustrate why these controls must operate together. A stolen token may allow an attacker to invoke a service until the token expires or is otherwise rejected, particularly when the token grants broad privileges or is accepted by unintended endpoints. An excessively permissive gateway may forward requests that should have been rejected, while incomplete backend authorization may allow a valid client to access information outside its business responsibilities. Misconfigured certificates or weak endpoint verification can further undermine confidence in the communicating party. These scenarios should be examined through controlled security tests that verify whether invalid, expired, incorrectly scoped, or incorrectly addressed tokens are rejected. Additional tests should examine unauthorized OData operations, excessive query requests, unexpected HTTP methods, and attempts to bypass gateway restrictions. Results should be documented with the configuration examined, the test conditions, the expected control behaviour, and the observed outcome. Such evidence is more reliable than assuming that the presence of a named security technology guarantees effective enforcement.

An integrated API governance model combines gateway enforcement, OAuth 2.0 token validation, OData restrictions, secure transport, backend authorization, and coordinated monitoring within a common control framework. Its principal benefit is consistency: each component has a defined responsibility, while the overall request path remains subject to complementary security checks. The figure should be placed after the discussion of TLS protection and certificate management, illustrating the relationship between clients, the API gateway, identity services, integration middleware, and protected SAP resources. The architecture should identify where authentication is verified, where authorization decisions are enforced, and where security events are recorded. Expected benefits include reduced unnecessary service exposure, stronger control over data access, improved traceability, and clearer responsibility for responding to security incidents. These benefits remain design expectations until validated through implementation and testing. A practical evaluation should therefore examine control coverage, rejection of unauthorized requests, service availability under controlled traffic conditions, and the quality of evidence available for subsequent security review.

Data Protection Architecture and Compliance Alignment

Data protection within hybrid SAP enterprise integration requires a coordinated approach to safeguarding information as it moves between SAP ECC, SAP Cloud Platform, integration middleware, and connected applications. Enterprise systems process information with different sensitivity levels, ranging from routine operational records to confidential financial information, employee details, customer records, and authentication credentials. The movement of this information across application and administrative boundaries increases the importance of defining consistent protection requirements. A data protection architecture must therefore address the complete information lifecycle, including collection, transmission, processing, storage,

access, retention, and disposal. Treating protection as an isolated encryption task leaves important risks unresolved, particularly when sensitive information is exposed through excessive permissions, unnecessary service responses, or poorly controlled operational records. The proposed approach places information classification and accountable handling at the foundation of hybrid integration security.

Data classification provides the basis for selecting safeguards according to the sensitivity, business value, and regulatory significance of information. In a hybrid SAP environment, financial transactions, employee records, customer information, supplier details, and technical credentials may require different levels of access restriction and monitoring. Classification rules should be established by the organization and applied consistently across connected systems, rather than determined independently by each application team. Data owners should define which information may be transferred to cloud services, which attributes are necessary for a business process, and which operations require additional approval. Integration interfaces should expose only the fields and records required by the receiving application. Where services return large collections of business data, filtering and access restrictions should be applied before unnecessary information crosses system boundaries. Consistent classification improves the ability to select appropriate encryption, authorization, retention, and audit controls while reducing avoidable exposure.

Protecting information in transit is essential because hybrid integration involves communication across networks and services that may be administered separately. Transport Layer Security can provide confidentiality and integrity for network communications when secure protocol settings, certificate validation, and endpoint authentication are correctly configured. Protection should cover the relevant communication paths between users, identity providers, API gateways, integration middleware, SAP ECC, and cloud-hosted services. Organizations should avoid assuming that traffic within an internal network is inherently trustworthy, particularly when systems communicate across segmented networks or shared infrastructure. Certificate validation failures, expired certificates, obsolete protocol configurations, and improperly trusted endpoints can weaken the security of otherwise encrypted connections. Operational procedures should therefore include certificate renewal, trust-store maintenance, configuration review, and controlled handling of connection failures. These measures help preserve the integrity of business information exchanged throughout the integration architecture.

Encryption and cryptographic key management provide complementary safeguards for information in transit and, where required, information stored in databases, backups, and other repositories. Effective encryption depends on more than selecting an algorithm: cryptographic keys must be generated, stored, accessed, rotated, and retired through controlled procedures. Key access should be restricted to authorized identities and separated from ordinary application administration wherever the deployment permits. Credentials, signing keys, and other secrets used by integration services should not be embedded in source code, exposed in configuration exports, or included in diagnostic messages. Key compromise and certificate revocation procedures should also be documented so that affected trust relationships can be withdrawn promptly. In SAP integration scenarios, cryptographic responsibilities may be distributed across enterprise infrastructure, middleware, identity services, and cloud components. Clear ownership and documented dependencies are therefore necessary to prevent gaps in protection when certificates or keys change.

Data minimization and purpose-based access reduce exposure by limiting the information processed or disclosed to what a defined business activity requires. A cloud application that needs a transaction status may not need access to the complete customer record, while a reporting service may require aggregated information instead of identifiable individual records. Interface design should reflect these distinctions through restricted service operations, selective field exposure, appropriate filtering, and narrowly defined authorization scopes. Purpose-based access also requires the organization to establish legitimate processing purposes and ensure that application permissions remain consistent with those purposes. Such restrictions can reduce the consequences of compromised credentials and simplify compliance reviews by limiting the quantity of sensitive information available to each service. However, minimization must be balanced against legitimate operational requirements, audit obligations, and data accuracy needs. Decisions about field restrictions and data transformations should therefore be documented and validated with the relevant business and data owners.

Audit logging and security event correlation provide the evidence needed to examine how information is accessed and processed across connected systems. Relevant records may include authentication outcomes, token validation failures, authorization decisions, administrative configuration changes, data export operations, and access to sensitive business functions. Individual logs often describe only one component of a transaction, making correlation necessary when an event passes through an API gateway, integration service, and backend application. Consistent timestamps, transaction identifiers, service identities, and request correlation identifiers can help reconstruct these interactions. Logging practices must also avoid capturing passwords, access tokens, cryptographic keys, or unnecessary personal information. Access to audit records should be restricted, and their integrity should be protected against unauthorized modification or deletion. Well-governed logging supports incident investigation, operational oversight, and compliance evidence, but its value depends on completeness, appropriate retention, and clearly assigned responsibility for reviewing significant events.

The General Data Protection Regulation establishes requirements relevant to the processing of personal data within its scope, including purpose limitation, data minimization, accuracy, storage limitation, integrity, confidentiality, and accountability. Hybrid SAP integration may involve employee, customer, supplier, or business contact information, making it necessary to assess the purpose and legal basis for processing, the responsibilities of participating organizations, and the applicable safeguards. Compliance cannot be established solely through encryption or access control, because organizational responsibilities also include appropriate documentation, retention decisions, handling of data subject rights where applicable, and procedures for managing personal data breaches. Organizations should determine which information is personal data, where it is transferred, who can access it, and how long it must be retained. The specific applicability of individual obligations depends on the processing activity and circumstances. Privacy requirements should consequently be translated into documented technical and administrative controls rather than treated as a generic certification claim.

Retention, exposure management, and incident response complete the data protection lifecycle. Information should be retained for documented operational, legal, and regulatory purposes, with deletion or disposal procedures applied when retention is no longer justified and no overriding obligation requires preservation. Excessive retention of exported records, integration payloads, temporary files, and audit data can increase the impact of unauthorized access. Incident response procedures should define how security events are detected, assessed, contained, investigated, and escalated across the organizations responsible for the connected systems. Where a suspected incident involves personal data, the organization should assess the applicable notification and documentation requirements under relevant law and established internal procedures. The figure should be placed after the discussion of audit logging and security event correlation, illustrating the relationship between classification, controlled transmission, protected processing, monitoring, retention, and disposal. A unified governance model connects these activities through clear ownership, documented policies, periodic reviews, and evidence-based assessment. This arrangement provides a coherent foundation for protecting enterprise information while supporting accountable and reliable SAP integration.

Results, Comparative Analysis, and Discussion

The security assessment establishes that hybrid SAP integration must be evaluated across the complete interaction between identity services, API interfaces, integration middleware, and protected enterprise resources. The principal finding of the architectural analysis is that the effectiveness of individual security controls depends on their coordination across system boundaries. Authentication mechanisms can verify identities, API gateways can enforce access policies, and encryption can protect communication, yet these safeguards may provide incomplete protection when their configurations or responsibilities are inconsistent. The assessment therefore examines the proposed integrated framework against a baseline characterized by independently administered controls. The comparison is based on documented security requirements, threat scenarios, and observable control properties rather than on production measurements. Accordingly, the findings represent an architecture-level assessment of expected security capabilities, not empirical proof of reduced incidents or quantified operational improvement.

The comparison between fragmented controls and the integrated framework highlights differences in consistency, visibility, and administrative accountability. In a fragmented arrangement, identity verification may be performed separately from backend authorization, gateway rules may not reflect application permissions, and audit records may remain distributed across systems without reliable correlation. These conditions complicate the identification of control gaps and increase the effort required to reconstruct suspicious transactions. The integrated framework addresses these weaknesses by defining responsibilities for identity validation, authorization enforcement, service exposure, communication protection, and event monitoring. Its principal advantage is not the introduction of entirely new security mechanisms, but the coordination of established controls through shared requirements and explicit enforcement points. However, integration also introduces dependencies between components, making configuration management, interoperability testing, and clear operational ownership essential to maintaining the intended security posture.

Identity assurance and authorization effectiveness depend on preserving the distinction between verifying an identity and granting access to a resource. The assessment identifies federated authentication, assertion validation, identity reconciliation, and lifecycle management as complementary controls for maintaining reliable identity relationships across SAP ECC and SAP Cloud Platform. Role-based access control and least-privilege restrictions provide additional safeguards by limiting the operations available to authenticated users and technical accounts. A coordinated framework can make inconsistencies easier to identify through periodic role reviews, documented identity mappings, and traceable authorization decisions. Nevertheless, federation alone cannot eliminate excessive permissions, stale accounts, or inappropriate privilege assignments. Its effectiveness depends on accurate identity attributes, sound trust configuration, timely account deactivation, and enforcement within the receiving application. These dependencies should be examined through controlled access tests and configuration reviews before any claim of improved authorization performance is made.

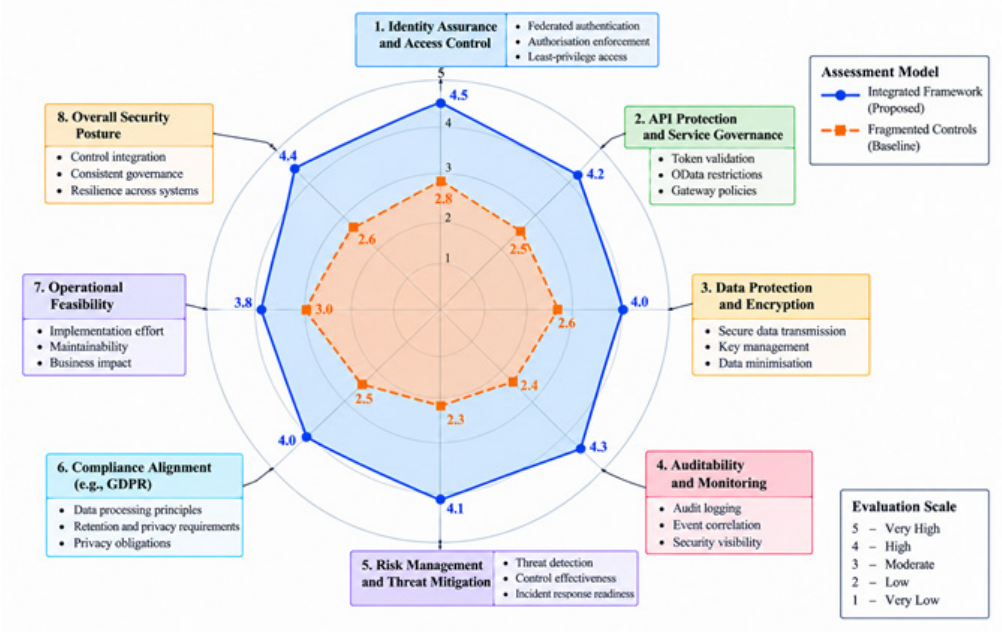

Figure 6: Comparative Security Assessment Results

API protection and service access governance provide a second important area of comparative assessment. The proposed framework combines gateway-level controls with OAuth 2.0 token validation, OData restrictions, request filtering, and backend authorization. This arrangement provides multiple opportunities to reject unauthorized requests before they reach sensitive business functions. Restricted endpoints and service operations can reduce unnecessary exposure, while rate limiting and request validation can help manage abusive traffic and resource consumption. A fragmented implementation may lack consistent policy coverage or depend excessively on gateway controls that are not reinforced by the underlying application. The integrated design seeks to address this weakness by assigning complementary responsibilities to gateway and backend components. Its practical effectiveness still depends on the correctness of individual policies, the scope of issued tokens, and the ability of the receiving service to validate permissions independently. Security testing must therefore include attempts to bypass gateway restrictions and invoke unauthorized operations directly.

Confidentiality, integrity, and auditability are strengthened when communication protection, controlled information disclosure, and reliable event records are treated as connected requirements. Transport Layer Security can protect information during transmission, while encryption and key management can provide additional safeguards for stored information where required. Data minimization and purpose-based access reduce the quantity of sensitive information exposed to individual applications. Audit logging supports accountability by recording relevant authentication events, access decisions, administrative changes, and significant business operations. Correlation across gateway, middleware, and backend records can improve the reconstruction of transactions that cross multiple systems. However, encryption does not prevent misuse by an authorized account, and logging does not guarantee that an incident will be detected or investigated successfully. The assessment therefore treats these safeguards as complementary controls whose effectiveness depends on configuration quality, monitoring procedures, key protection, and the availability of trustworthy evidence.

Risk reduction is examined through representative threat scenarios rather than assumed numerical improvements. For compromised credentials, federated identity controls, appropriate authentication policies, and lifecycle management can reduce opportunities for continued unauthorized access. For stolen or misused access tokens, audience restrictions, scope validation, secure storage, and limited token lifetimes can constrain the resources available to an attacker. For excessive OData exposure, service restrictions, backend authorization, and request validation can reduce unnecessary data disclosure. For interception of network traffic, correctly configured TLS and certificate validation can protect communication confidentiality and integrity. For incomplete incident records, coordinated logging and event correlation can improve investigative visibility. These outcomes describe the expected effects of the proposed controls under suitable implementation conditions. A defensible quantitative comparison would require predefined risk scales, documented control tests, reproducible scoring procedures, and evidence collected from an actual or controlled test environment.

Implementation introduces operational constraints

Table 2: Comparative Results for Baseline and Integrated Security Controls

Evaluation criteria	*Baseline security controls*	*Integrated security framework*	*Evidence status*
Identity management	Separate authentication across systems	Federated identity and single sign-on	Requires testing
API security	Independently configured API restrictions	Centralized gateway policies and token validation	Requires testing
Data protection	Inconsistent encryption and access restrictions	Coordinated encryption and data access controls	Requires configuration review
Access control	System-specific user permissions	Role-based access and least privilege	Requires access review
Security monitoring	Separate audit logs	Coordinated logging and event monitoring	Requires log analysis
Compliance	Independently managed privacy procedures	Integrated data protection and audit requirements	Requires compliance review

that must be considered alongside the expected security benefits. Existing SAP landscapes may contain legacy interfaces, custom applications, technical accounts, and business processes that depend on established authorization arrangements. Introducing stricter gateway rules, changing identity mappings, or reducing service permissions can disrupt legitimate transactions if dependencies are not fully understood. Certificate rotation, key management, token configuration, and centralized monitoring also require ongoing administrative capacity. Excessively restrictive policies may reduce interoperability, while overly permissive exceptions can weaken the intended security model. A phased implementation is therefore preferable to an uncontrolled replacement of existing configurations. Organizations should begin with an inventory of interfaces and identities, identify high-risk exposures, establish baseline measurements, and validate proposed changes through controlled testing before production deployment. Operational acceptance should consider both security effectiveness and the ability to maintain reliable business services.

The principal research contribution is an integrated evaluation perspective that connects federated identity assurance, zero-trust API governance, and resilient data protection within one hybrid SAP security framework. The comparison suggests that architectural consistency, defined control ownership, and traceable enforcement are important characteristics of a defensible integration design. The figure should be placed after the risk-reduction discussion and should compare the baseline and integrated framework across explicitly defined assessment dimensions. If numerical ratings are used, the figure must identify the scoring scale and distinguish illustrative values from measured results. It should accompany the comparison by recording each criterion, baseline condition, proposed control, observed evidence, and validation status. The current assessment remains limited by its conceptual reference environment and the absence of independently measured production outcomes. Future validation should examine representative SAP configurations, controlled attack scenarios, authorization failures, monitoring coverage, and operational performance. Such work would determine whether the anticipated advantages remain evident under real deployment conditions and provide a stronger empirical foundation for subsequent research.

Conclusion and Future Work

Hybrid integration between SAP ECC and SAP Cloud Platform requires a security architecture that protects enterprise information while maintaining reliable communication across applications, services, and administrative boundaries. This study examined the challenges arising from fragmented identity administration, inconsistent API authorization, and uneven data protection practices within connected enterprise environments. Its principal objective was to establish an integrated framework combining federated identity assurance, zero-trust API governance, and resilient data protection. The architectural analysis indicates that these security dimensions should be treated as interdependent responsibilities rather than isolated technical functions. Authentication, service authorization, secure communication, and auditability contribute to a common security objective, but their effectiveness depends on consistent configuration and clearly assigned operational responsibilities. The proposed framework provides a structured basis for examining these relationships and identifying weaknesses that may otherwise remain distributed across individual system configurations.

Federated identity assurance emerged as an essential component of the proposed architecture because it establishes a controlled basis for recognizing users across connected SAP environments. Federation and single sign-on can reduce duplicated authentication arrangements and improve the consistency of identity administration when supported by correctly configured trust relationships. Assertion validation, identity mapping, session management,

and timely account deactivation remain necessary to prevent authentication success from being interpreted as unrestricted access. Role-based authorization and least-privilege permissions provide the complementary controls needed to restrict access to appropriate business functions. The study therefore concludes that identity federation should be evaluated alongside authorization enforcement and identity lifecycle governance. This combined perspective helps organizations address both unauthorized access by unidentified parties and inappropriate privileges held by otherwise legitimate accounts.

API governance represents a second central contribution of the framework. Gateway policies, OAuth 2.0 token validation, OData service restrictions, secure communication, and backend authorization can provide complementary safeguards for business services exposed across hybrid boundaries. Their coordinated application can reduce unnecessary service exposure and improve the consistency of access enforcement. However, the presence of an API gateway or an authorization token does not independently guarantee that every request is legitimate or appropriately restricted. Receiving applications must validate the permissions relevant to the requested operation, while monitoring mechanisms should identify suspicious activity and provide sufficient evidence for investigation. The findings consequently support a layered approach in which gateway controls, application-level authorization, and operational monitoring reinforce one another. This approach is particularly relevant where business transactions cross multiple integration components and require consistent protection throughout the request path.

Integrated data protection extends this security model beyond access control to encompass the handling of information throughout its lifecycle. Classification, data minimization, encryption, cryptographic key management, controlled retention, and audit logging address different aspects of confidentiality, integrity, and accountability. Privacy requirements also influence how information is collected, disclosed, processed, and retained across connected applications. The framework incorporates these considerations into the integration architecture rather than treating compliance as a separate administrative activity. Its contribution lies in linking technical safeguards with governance responsibilities, including data ownership, access review, incident handling, and the maintenance of reliable audit evidence. Such coordination can help organizations identify inconsistencies between their stated information protection policies and the actual behaviour of integrated services. Compliance outcomes nevertheless depend on the applicable legal requirements, organizational practices, and evidence demonstrating that the relevant controls operate effectively.

For SAP security architects and enterprise practitioners, the proposed framework offers a practical structure for reviewing existing integration arrangements and prioritizing corrective action. Organizations can begin by inventorying exposed interfaces, identifying sensitive information flows, documenting identity trust relationships, and examining the permissions assigned to users and technical accounts. Subsequent reviews can evaluate token handling, OData restrictions, certificate management, audit coverage, and the consistency of data protection requirements. A phased implementation allows high-risk weaknesses to be addressed while reducing the possibility that security changes will interrupt legitimate business processes. Particular attention should be given to legacy interfaces, custom applications, and technical dependencies that may not be visible in standard architecture documentation. The framework is therefore intended to support disciplined security engineering and governance rather than prescribe an identical configuration for every SAP deployment.

The academic contribution of this study is its unified treatment of identity assurance, API governance, and data protection as mutually dependent dimensions of hybrid enterprise security. It provides a conceptual basis for investigating how established security mechanisms can be coordinated across systems with different administrative responsibilities and authorization models. The assessment criteria also offer a starting point for comparing fragmented and integrated security arrangements through observable configuration properties and documented test outcomes. This contribution should be understood as an architectural framework and evaluation proposition, not as a validated claim of superior performance. Its wider relevance depends on whether subsequent studies can demonstrate that the proposed relationships remain effective across different enterprise environments, deployment configurations, and operational conditions. By making control dependencies explicit, the framework can support more focused research into the governance and assurance of distributed enterprise applications.

Several limitations must be acknowledged. The assessment is based on a conceptual reference architecture and established security principles rather than a reported production deployment or independently measured experimental results. Consequently, expected benefits such as improved traceability, reduced unnecessary exposure, and more consistent authorization have not been established through quantitative testing in this study. Actual outcomes may vary according to SAP product versions, integration configurations, identity provider capabilities, administrative practices, and the maturity of existing security controls. The comparative tables and assessment framework describe expected differences that require verification through configuration inspection and controlled testing. Future empirical work should define reproducible threat scenarios, establish baseline measurements, document the

scoring method, and evaluate both security outcomes and operational consequences. Such validation is necessary before numerical improvements or general claims of effectiveness can be justified.

Future research should examine the framework through controlled test environments and representative enterprise deployments involving SAP ECC, SAP Cloud Platform, identity providers, API gateways, and integration middleware. Studies could measure unauthorized request rejection, identity mapping accuracy, privilege reduction, audit event completeness, and the time required to investigate defined security incidents. Additional work should evaluate the operational effects of certificate rotation, token expiration, identity deprovisioning, and policy changes across connected services. Scalability, interoperability, maintainability, and the effect of stricter controls on legitimate business transactions also warrant systematic examination. Researchers could develop repeatable assessment procedures that distinguish configuration compliance from demonstrated resistance to specific threats. A measurable and auditable evaluation model would strengthen comparisons across implementations and support evidence-based security decisions. Continued investigation along these lines can establish a stronger empirical foundation for secure hybrid SAP integration and improve the reliability of enterprise security governance as application environments become increasingly interconnected.

References

[1] Fett, D., Küsters, R., et al. (2016). A comprehensive formal security analysis of OAuth 2.0. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1204–1215. https://doi.org/10.1145/2976749.2978385

[2] Indu, I., Anand, P. M. R., et al. (2018). Identity and access management in cloud environment: Mechanisms and challenges. *Engineering Science and Technology, an International Journal, 21*(4), 574–588. https://doi.org/10.1016/j.jestch.2018.05.010

[3] Singh, N., Singh, A. K. (2018). Data privacy protection mechanisms in cloud. *Data Science and Engineering, 3*, 24–39. https://doi.org/10.1007/s41019-017-0046-0

[4] Hu, V. C., Kuhn, D. R., et al. (2018). Access control for emerging distributed systems. *Computer, 51*(10), 100–103. https://doi.org/10.1109/MC.2018.3971347

[5] Luo, J., Wang, H., et al. (2016). A novel role-based access control model in cloud environments. *International Journal of Computational Intelligence Systems, 9*(1), 1–9. https://doi.org/10.1080/18756891.2016.1144149

[6] Verginadis, Y., Michalas, A., et al. (2017). PaaSword: A holistic data privacy and security by design framework for cloud services. *Journal of Grid Computing, 15*, 219–234. https://doi.org/10.1007/s10723-017-9394-2

[7] Fan, K., Yao, X., et al. (2016). A new usage control protocol for data protection of cloud environment. *EURASIP Journal on Information Security, 2016*, Article 7. https://doi.org/10.1186/s13635-016-0031-6

[8] Kim, J., Nepal, S. (2016). A cryptographically enforced access control with a flexible user revocation on untrusted cloud storage. *Data Science and Engineering, 1*, 149–160. https://doi.org/10.1007/s41019-016-0014-0

[9] Yimam, D., Fernandez, E. B. (2016). A survey of compliance issues in cloud computing. *Journal of Internet Services and Applications, 7*, Article 5. https://doi.org/10.1186/s13174-016-0046-8

[10] Russo, B., Valle, L., et al. (2018). Cloud computing and the new EU General Data Protection Regulation. *IEEE Cloud Computing, 5*(6), 58–68. https://doi.org/10.1109/MCC.2018.064181121

[11] Mouli, V. R., Jevitha, K. P. (2016). Web services attacks and security: A systematic literature review. *Procedia Computer Science, 93*, 870–877. https://doi.org/10.1016/j.procs.2016.07.265

[12] Müller, A., Ludwig, A., et al. (2017). Data security in decentralized cloud systems: System comparison, requirements analysis and organizational levels. *Journal of Cloud Computing, 6*, Article 15. https://doi.org/10.1186/s13677-017-0082-3

[13] Razaque, A., Rizvi, S. S. (2017). Privacy preserving model: A new scheme for auditing cloud stakeholders. *Journal of Cloud Computing, 6*, Article 7. https://doi.org/10.1186/s13677-017-0076-1

[14] Tu, T., Rao, L., et al. (2017). Privacy-preserving outsourced auditing scheme for dynamic data storage in cloud. *Security and Communication Networks, 2017*, Article 4603237. https://doi.org/10.1155/2017/4603237

[15] Fremantle, P., Aziz, B. (2018). Cloud-based federated identity for the Internet of Things. *Annals of Telecommunications, 73*, 415–427. https://doi.org/10.1007/s12243-018-0641-8

[16] Paladi, N., Gehrmann, C., et al. (2016). Providing user security guarantees in public infrastructure clouds. *IEEE Transactions on Cloud Computing, 5*(3), 405–419. https://doi.org/10.1109/TCC.2016.2525991

[17] Lang, B., Wang, J., et al. (2017). Achieving flexible and self-contained data protection in cloud computing. *IEEE Access, 5*, 1510–1523. https://doi.org/10.1109/ACCESS.2017.2665586

[18] Jung, S. W., Jung, S. (2017). Personal OAuth authorization server and push OAuth for Internet of Things. *International Journal of Distributed Sensor Networks*. https://doi.org/10.1177/1550147717712627

[19] Cui, H., Yi, X., et al. (2018). Achieving scalable access control over encrypted data for edge computing networks. *IEEE Access, 6*, 30049–30059. https://doi.org/10.1109/ACCESS.2018.2844373

[20] Namasudra, S., Roy, P. (2016). Secure and efficient data access control in cloud computing environment: A survey. *Multiagent and Grid Systems, 12*(2), 69–90. https://doi.org/10.3233/MGS-160244

[21] Alansari, S., Paci, F., et al. (2017). Privacy-preserving access control in cloud federations. *2017 IEEE International Conference on Cloud Computing (CLOUD)*. https://doi.org/10.1109/CLOUD.2017.108

[22] Sookhak, M., et al. (2017). Attribute-based data access control in mobile cloud computing: Taxonomy and open issues. *Future Generation Computer Systems, 72*, 273–287. https://doi.org/10.1016/j.future.2016.08.018

[23] Hussain, S. A., Fatima, M., et al. (2017). Multilevel classification of security concerns in cloud computing. *Applied Computing and Informatics, 13*(1), 57–65. https://doi.org/10.1016/j.aci.2016.03.001

[24] Zhou, L., et al. (2018). Secure and fine-grained access control on e-healthcare records in mobile cloud computing. *Future Generation Computer Systems, 78*(3), 1020–1026. https://doi.org/10.1016/j.future.2016.12.027

[25] Ahuja, R., Mohanty, S. K., et al. (2017). A scalable attribute-

set-based access control with both sharing and full-fledged delegation of access privileges in cloud computing. *Computers & Electrical Engineering, 57*, 241–256. https://doi.org/10.1016/j.compeleceng.2016.11.028

[26] Dos Santos, D. R., et al. (2016). A framework and risk assessment approaches for risk-based access control in the cloud. *Journal of Network and Computer Applications, 74*, 86–97. https://doi.org/10.1016/j.jnca.2016.08.013

[27] Ye, X. (2016). Privacy preserving and delegated access control for cloud applications. *Tsinghua Science and Technology, 21*(1), 40–54. https://doi.org/10.1109/TST.2016.7399282

[28] Sciancalepore, S., Piro, G., et al. (2018). On the design of a decentralized and multiauthority access control scheme in federated and cloud-assisted cyber-physical systems. *IEEE Internet of Things Journal, 5*(6), 5190–5204. https://doi.org/10.1109/JIOT.2018.2864300

[29] Deshmukh, P. (2017). Design of cloud security in the EHR for Indian healthcare services. *Journal of King Saud University Computer and Information Sciences, 29*, 281–287. https://doi.org/10.1016/j.jksuci.2016.01.002

[30] Li, W., Liu, B. M., et al. (2019). Unified fine-grained access control for personal health records in cloud computing. *IEEE Journal of Biomedical and Health Informatics, 23*(3), 1278–1289. https://doi.org/10.1109/JBHI.2018.2850304