**International Journal of Information and Cybersecurity**
**DLpress** is a publisher of scholarly books and peer-reviewed scientific research. With a dedication to academic excellence, DLpress publishes books and research papers on a diverse range of topics spanning various disciplines, including but not limited to, science, technology, engineering, mathematics, social sciences, humanities, and arts.
Published 15, May, 2021

# Implementing Encryption and Cybersecurity Strategies across Client, Communication, Response Generation, and Database Modules in E-Commerce Conversational AI Systems

**Rahul Khurana** [1]

[1]**Bothell, WA, USA**

RESEARCH ARTICLE

## Abstract

The integration of conversational AI into e-commerce enables better customer service and personalization of shopping experiences. This advancement in technology has also warranted the focus of a number of important concerns regarding security, such as sensitive user data protection and protection of transaction information. This paper will be proposing a systematic architecture model for securing conversational AI systems in an e-commerce domain through encryption techniques in tandem with adherence to cybersecurity practices. This model contains four major modules: *client module*, *communication module*, *response generation module*, *and database module*. In this, the client module uses strict authentication mechanisms for access. Even multi-factor authentication can be used for the same. The coding should be done in such a way to avoid as many vulnerabilities as possible, including injection attacks, by using secure coding practices. The communication module should implement secure data transmission protocols like TLS. This module is integrated with rate limiting to defend against Denial of Service attacks. The security features are integrated inside the AI engine through adversarial training with Differential Privacy in the Response Generation module, serving to reduce risks related to data leakage and manipulation. This involves access control so as to provide mechanisms against unauthorized use of functions. Data encryption in the database is at rest using algorithms like Advanced Encryption Standard, or AES, for secure data storage. Besides, the foundation for access control is the principle of least privilege; monitoring provides an added capability for the detection of unauthorized attempts at access. The model also formalizes definitions of security functions and properties to represent authentication, encryption, and access control mechanisms. The proposed model will further advance the design of secure AI systems with a practical framework that can be adapted for real-world e-commerce applications, with the aim of reaching data protection standards and improving user trust.

Keywords: adversarial training, conversational AI, data encryption, e-commerce security, multi-factor authentication, secure architecture, TLS

**OPEN ACCESS**
**Reproducible Model**

***Edited by***
*Associate Editor*

***Curated by***
*The Editor-in-Chief*

***Submitted*** *18, Janury, 2021*

***Accepted*** *11, May, 2021*

***Citation***
*Khurana, R. (2021) Implementing Encryption and Cybersecurity Strategies across Client, Communication, Response Generation, and Database Modules in E-Commerce Conversational AI Systems*

## 1 Introduction

E-commerce, a term describing the trading of goods and services via electronic networks, has been catalyzed by the digitization of international commerce. This phenomenon is a result of major improvements in networking technologies, secure data transfer protocols, and web interfaces, all of which allow efficient transactions between businesses and other transactional groupings [1, 2]. E-commerce doesn't just revolve around transactions; it incorporates digital marketing, logistics, payment systems, analytics on consumer data, which basically turns age-old commerce models into dynamic, data-driven interplay further efficient, more scalable, and accessible over borders [1, 3].

Different elements in the ecosystem of e-commerce play their role in maintaining function and integration in digital marketplaces. At the core lies web servers and application servers, where online storefronts are housed and run. These in turn connect to the backend databases, which may keep inventory, order processing, and user information in structures using SQL or NoSQL to handle the disparate data requirements. The payment gateways allow for the processing of transactions, while keeping customers safe with encryptions such as SSL or TLS. Combined with the user experience, CDNs work to optimize digital content delivered on geographically dispersed servers, reducing latency times and increasing response. More often than not, APIs from e-commerce platforms connect these components with third-party services, increasing functionality and extensibility of the overall ecosystem [4, 3].

The architecture of e-commerce platforms has a core role in their performance and scalability and, therefore, adaptability to several business models. The multilayer architecture is one of the widely adopted architectures in the e-commerce domain, where usually a three-tier approach exists. The presentation layer provides the user interface and interfaces with the users through web browsers or mobile apps [5]. The application layer houses the business logic processing and maintains the operations like pricing algorithms, shopping cart functionality, and user authentication. Finally, the data layer is used for data storage, retrieval, and manipulation related to products, transactions, and user profiles [6]. The trend has now shifted more towards microservices architectures which decompose monolithic systems into modular, loosely coupled services communicating over RESTful APIs or gRPC. This is enabling scalability whereby a number of different services-order management, product cataloging, or payment processing-can be scaled independently, thus allowing better utilization of resources.

Electronic commerce growth goes along with major changes in technology and everyday life. First, the Internet of the 1990s made available basic transactions such as static websites, which could list products and take orders via email. This was when businesses were getting their feet wet, testing the digital channels by basically digitizing their catalogs and allowing limited forms of interaction [7]. Eventually, an increase in internet speeds and browser capabilities allowed dynamic web pages, enabling real-time retrieval and updates of data. This allowed for more interactive e-commerce experiences, whereby customer interactions could be tailored depending on their browsing behaviors. Integration of DBMS with web platforms further streamlined inventory and customer relationship management, making way for comprehensive online storefronts [4].

The succeeding development of secure online payment systems was fundamental to the development of electronic commerce, turning what was once the exception into the norm in business. Then, protocols such as SSL, followed by TLS, picked up steam and became the standard for the industry that allowed encrypted transactions to build up consumer trust. This generation also introduced far more specialized e-commerce platforms with pre-set functionalities for product listing management, user account management, and order processing. Web 2.0 technologies recast e-commerce with an emphasis on user-generated content and participatory digital experiences that shaped not only the development of online marketplaces but also the architecture of social commerce [8].

With increased mobile devices, m-commerce then came to provide users with e-commerce access through mobile-optimized websites and apps. It needed adaptive designs, responsive layouts, and APIs for just mobile, so that the user experiences can be delivered consistently across different devices. This further catalyzes the integration of digital wallets, including those using NFC protocols, as it simplifies payment processes for mobile users. The omnichannel strategies allowed the businesses to integrate the user experience over physical stores, online platforms, and mobile interfaces for best customer engagement and retention [7].

Conversational AI has become an elemental integrant of e-commerce sites in the wake of demanding needs for enhanced user engagement and effectively deploying customer service [9]. Applications of models in NLP would parse user input and interpret customer queries and intent with increasing precision. These models use techniques of deep learning that include transformer architecture in order to analyze the textual data of the user message context, sentiment, and entity recognition. It, therefore, empowers the AI systems with better interpretation of user

queries, hence returning more relevant and timely responses. Ongoing improvements of these models through fine-tuning of LLMs on domain-specific data have let e-commerce platforms offer customer interactions customized to their preference and requirements.

These conversational AI systems basically integrate with various APIs at the backend through which they fetch data from the inventory of e-commerce [10]. Such integration allows the AI to fetch real-time data on inventory, status of orders, product information, and user history-all very valuable in formulating responses which directly address customer needs. It uses API-driven communication to make the conversational AI provide a layer of abstraction between the users and the underlying data structures, sending simplified queries to what the user would want. Suppose the user has asked, for example, whether the product is in stock, it fetches the data from stock, further does runtime query processing, and comes up with an accurate response to the user. This capability smoothes the user experience and optimizes back-end operations by reducing the need to manually fetch data and intervene with customer service.

Many e-commerce applications of conversational AI have been deployed on cloud-based architectures that support scalability and high availability. Cloud platforms provide the computational resources needed to train and deploy NLP models, which can then process large volumes of concurrent user interactions. This strategy also relies on distributed computing frameworks, such as Kubernetes or serverless functions, to work out model inference across multiple nodes. Containerization technologies allow seamless deployment of these AI models, maintaining consistency between the development and production environments. Besides, cloud services provide pre-trained NLP models and machine learning tools that can be further tuned by transfer learning to fit specific needs related to e-commerce. What this architecture does is enable e-commerce platforms to scale their AI-powered interactions with efficiency, handle fluctuating demands, and maintain response latency accordingly [11, 8].

Another reinforcement learning algorithm employed in conversational AI further enhanced the ability of these systems to learn and adapt from user interactions over time. The reinforcement learning will let the AI agent optimize the responses through some sort of reward mechanism, iteratively improving user satisfaction. It refines its methodology of guiding customers through the shopping journey in different conversation scenarios, whether navigating categories of products or recommending personalized ideas for customers. This approach integrates user feedback into the learning process, developing a more intuitive understanding of patterns in user behaviors. The resultant models are further fine-tuned for improved conversational fluency and can engage with complex queries while generating responses sensitive to context-an indication that understanding user needs has been accomplished [12].

Sentiment analysis also takes on an important position with respect to the functions of conversational AI in e-commerce. The models of sentiment analysis classify user inputs based on their emotional tone, range between positive and negative, and neutral sentiments. This enables the AI to alter the response strategies, putting emphasis on solving the complaints or dissatisfaction of the user with more empathy. For instance, frustration about delays in shipment can have the user routed to a higher priority support queue, with automatic compensatory options. Sentiment analysis will, therefore, enable e-commerce platforms to keep their metrics positive and ensure that the interactions fall within a reasonable level of responsiveness and tone. This application of AI helps a platform enhance its capability to sustain positive user experiences by nipping the potentially negative ones in the bud.

The integration of voice-based conversational agents, enabled through advances in the area of ASR and TTS, is becoming increasingly prominent. This is mainly because ASR systems translate spoken languages into text that the AI then processes using existing NLP pipes. Similarly, TTS systems generate human-like speech based on the text responses formulated by the AI; thus, it realizes natural and intuitive voice interactions. Such voice-based interactions are highly relevant for mobile and IoT-based e-commerce platforms where users can interact with digital store fronts through their voice-activated devices. Voice commands are recognized correctly, for the recognition uses pre-trained ASR models combined with domain-specific fine-tuning. Adding TTS technologies enhances such e-commerce services for users who prefer auditory interactions

rather than text-based ones [13].

## 2 Problem Statement

This means that the deployment of Conversational AI will manifold increase the incidence of related security issues in e-commerce platforms regarding sensitive user data and transactional information, as conversational AI systems would deal with high volumes of PII information, including payment information and interaction histories [14, 15, 16, 17], fairly regularly. Users usually leave sensitive data, in the form of text or voice queries, during various interactions that get further processed by AI models on the back. The data exchanged during interaction between users and the backend servers may be intercepted using techniques like man-in-the-middle attacks if security is not properly considered. This risk is further complicated by the nature of natural language interactions; user inputs can vary widely, thereby exposing more data than in traditional e-commerce platforms [18, 19].

Besides the risk of data in transit, security vulnerabilities arise from the exposure brought about by the AI models themselves through adversarial inputs. These are attacks made via crafted user inputs to force the AI into producing a desired output or to probe for information, which should otherwise be confidential. The result of such attacks could be inability on the AI's part to maintain interactions intact and confidential; potential scenarios include user data exposure as a result of manipulated responses. Large, complex models in the conversational AI system raise the risk level, as they are very likely to emit fragments of information if poorly trained or fine-tuned on sensitive data.

The challenge of secure data storage with conversational AI also becomes pronounced. Most interaction logs tend to have more elaborate records of user behavior, queries, and responses; thus, in essence, they are a repository of sensitive information that needs elaborate management. Retaining such data for model refinement or improving the user experience increases the attack surface, since, in the face of any breach, one could end up with detailed customer information. Data privacy regulations require tight controls over the storage and access of such data. Even with compliance, retention of conversational data possesses the inherent risk of unauthorized access through internal or external data breaches.

These systems usually depend on several external services for fetching real-time product details, handling payments, and verification of user information [20]. With this reliance on third-party services comes the potential for security gaps, as every API interaction introduces the avenues of weaknesses that attackers can use to attack the system. Malicious actors could then use these interactions to perform API injections to manipulate the flow of data between the AI system and the backend services. Because these systems interact with each other, any breach at the API level may lead to leakage of a broad base of transaction and user data, increasing overall security risk. The training process of AI models themselves can sometimes inadvertently pose security challenges.

Large language models used in conversational AI are trained from large datasets that may contain sensitive information unless properly curated. This indicates that these models, being intrinsically complex, sometimes remember even elements of the data they have been trained on and reproduce them when reproduction increases the risk of leakage through user queries that may invoke memorized information. This concern becomes much more sensitive in those training data scenarios that contain unstructured customer interactions or other forms of sensitive records. This would certainly raise significant concerns with regard to how conversational AI systems should be designed and deployed in the context of electronic commerce.

## 3 System Overview

The architecture of a conversational AI system in e-commerce is built around four key modules: the Client Module, the Communication Module, the Response Generation Module, and the Database Module. Each module has a certain functionality that assists the users in interacting with the AI system.

The Client Module represents the interface with which users will be interacting with the chatbot; it includes web applications, mobile apps, and voice. It takes input from users, whether it be text, voice, or touch, and it authenticates users through various methods like OAuth-based logins or biometric verification. This layer ensures that messages can be sent by a user to the system for processing and action by the chatbot.

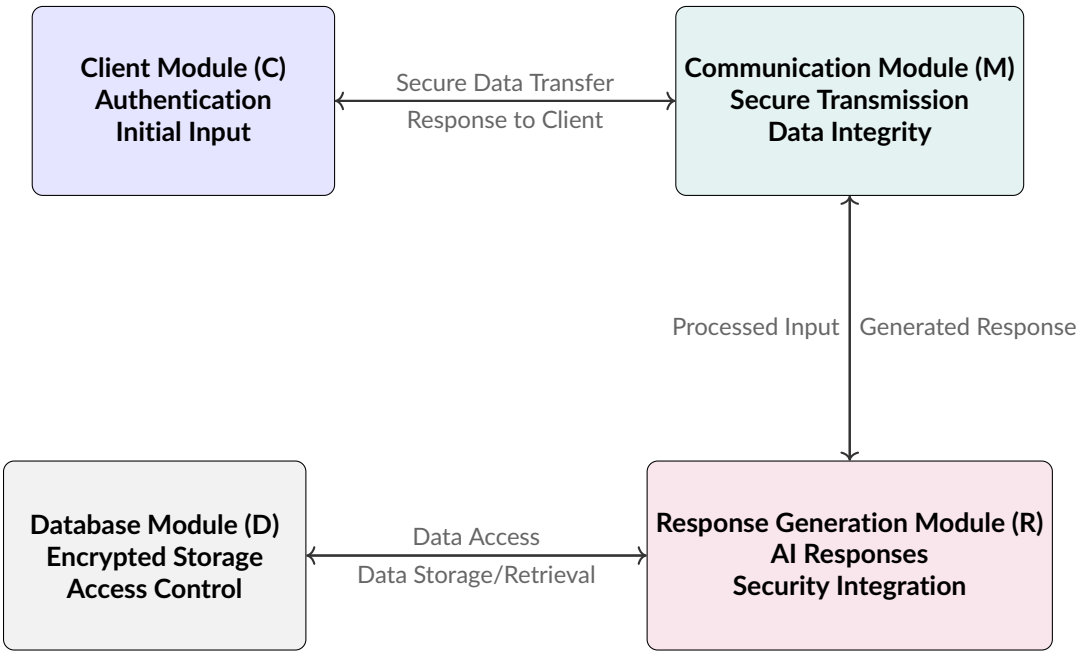


**Figure 1.** Architectural structure of the proposed model, showing the interaction between the client, communication, response generation, and database modules.

The Communication Module takes over the task of sending that message securely from the client-side to the server-side once a message has been submitted by the user. This comes about with the assistance of encryption protocols, such as HTTPS, which will protect the data in transit and maintain the confidentiality and integrity of communications. Included are the authentication protocols in this module that serve to validate all communication sessions, hence data integrity. It provides monitoring of network traffic in order to form patterns that might cause harm, such as DDoS attacks, and initiates mitigations against such risks. This module thus ensures that there is reliable and safe data exchange between users and the backend.

The Response Generation Module will take that input from the user and create an appropriate reply. This would make use of the method in NLP, analyzing and understanding the meaning of the input message and parsing it into a format understandable by the system. In other applications, embeddings convert inputs to vector representations such as those used in deep learning models, enabling the system to capture context and meaning of messages. This module would, in turn, respond-whether by directly synthesizing a response using models like sequence-to-sequence architectures or retrieving an appropriate response from some sort of pre-stored repository-so that the response would be correct in respect to the user's query and conversation context.

The Database Module retains the data about conversations-conversation history and user preferences and other contextual data. The chatbot then uses this storage to establish continuity within the conversations. Past conversations or specific user information, like previous purchases, can therefore easily be referenced. The database also allows for structured knowledge, like product information or user preferences, that informs the chatbot's responses. In such a way, the AI system will get to adjust its behavior over time to best fit the response of interest for any particular user.

It allows a user's message to pass through a series of well-defined stages: after the user has

| Symbol | Definition |
|---|---|
| $U$ | The set of all users interacting with the system, $U = \{u_1, u_2, \ldots\}$. |
| $C$ | The set of all client devices used to access the system, $C = \{c_1, c_2, \ldots\}$. |
| $F$ | The set of possible authentication factors (e.g., passwords, tokens, biometrics). |
| $M$ | The set of messages transmitted between modules. |
| $E$ | The set of encrypted messages, where $E = \text{Enc}(M)$. |
| $D$ | The set of all data elements, including:<br>$D_{\text{user}}$: User Data<br>$D_{\text{transaction}}$: Transaction Data<br>$D_{\text{sensitive}}$: Sensitive Data |
| $K_{\text{pub}}$, $K_{\text{priv}}$ | Sets of cryptographic keys used for encryption and decryption. |
| $R$ | The set of defined roles within the system for access control purposes. |
| $P$ | A function mapping roles to sets of permitted actions or resources, $P : R \rightarrow 2^A$, where $A$ is the set of actions or resources. |

**Table 1.** Notation and Preliminaries for Proposed Security Mechanisms

logged into the platform and sent a message via the Client Module, the Communication Module makes sure the message reaches the Response Generation Module securely. Then the Response Generation Module processes the input and formulates a response. Meanwhile, the Database Module stores new information simultaneously with the view for an even more refined experience for users in subsequent interactions. Each module thus plays a specific role within the well-fabricated implementation of effective and secure communication of users with the e-commerce AI system.

## 4 Client Module ($C$)

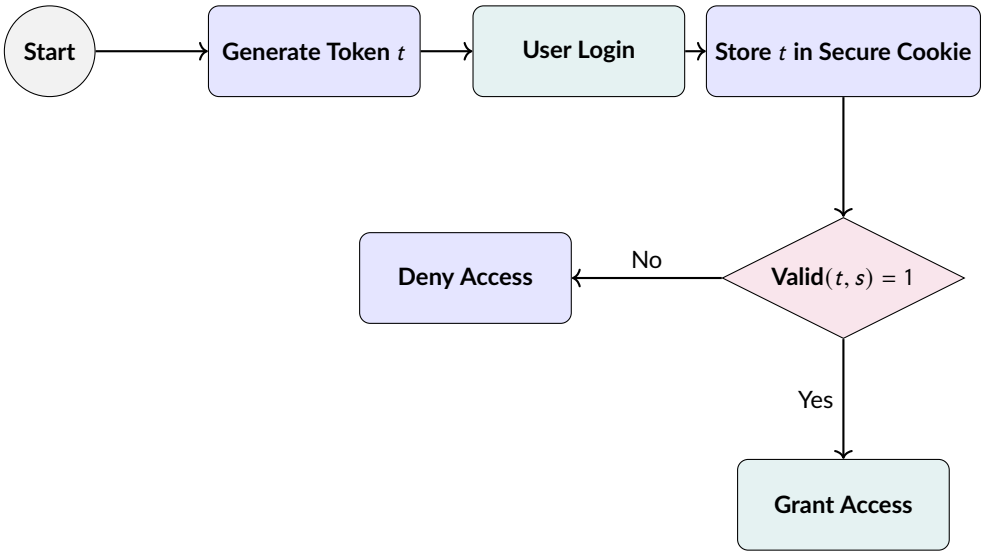


**Figure 2.** Flow of Session Token Validation Process

The client module is considered the entry through which the user interacts with the conversational AI. This should be responsible for authentication, preliminary data inputting, and enforcing all

kinds of security measures on the client side. The E-commerce UI is designed for interaction in terms of browsing products, maintaining shopping carts, and checkout-related processes. This interface needs to support secure session management so as to maintain the safety of user interactions. Session tokens $t \in T$ are created at the time of user authentication, stored as secure cookies and linked to a session $s$. A token $t$ is said to be valid if the following condition holds:

$$\text{Valid}(t, s) = \begin{cases} 1 & \text{if time}(t) < \text{expiry}(s) \text{ and user}(t) = \text{user}(s) \\ 0 & \text{otherwise} \end{cases}$$

This logic ensures that tokens expire after a set time or upon user logout, hence mitigating a number of risks such as session hijacking. In addition, input validation is applied on all the users' inputs to ensure that data conforms to security rules before processing can take place. Such processes can be represented by the function:

$$\text{Validate}(i) = \begin{cases} 1 & \text{if } i \in I \text{ satisfies all validation criteria} \\ 0 & \text{otherwise} \end{cases}$$

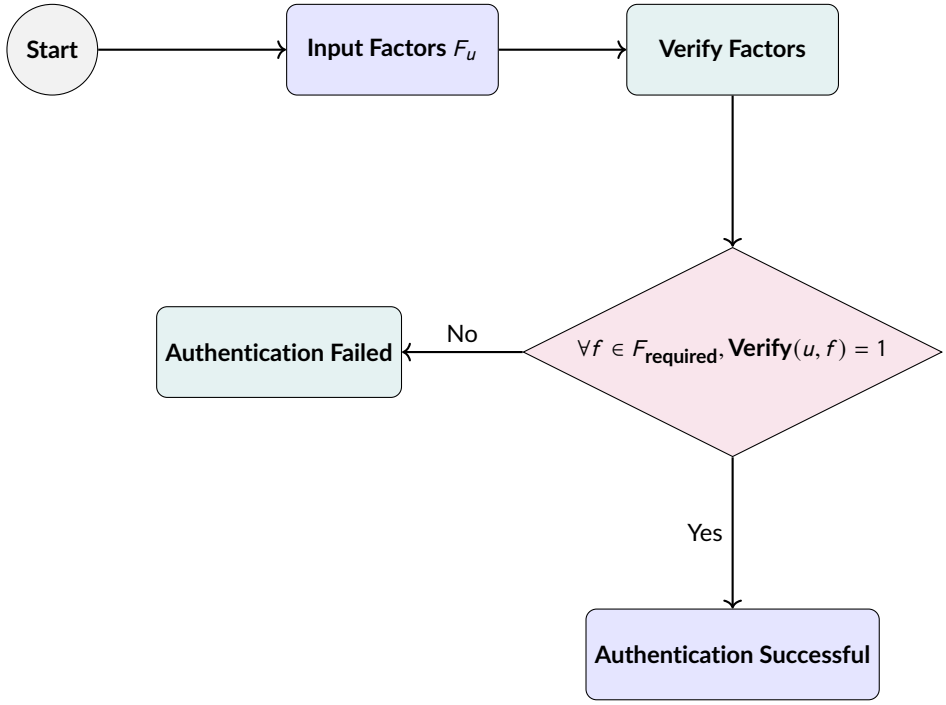


**Figure 3.** Authentication Process with Multi-Factor Verification

Where $i \in I$ is an input provided by the user, and $I$ is the set of all the valid inputs. CAPTCHA systems are employed to identify the human user from a bot by solving challenges that would lower the number of automated attempts to access the platform.

The authentication and payment processing layer manages functions related to user identification and transaction validation. Authentication is a key ingredient here, primarily MFA. The user will have to prove her identity through several independent credentials. Let $F_u \subseteq F$ be the set of authentication factors provided by user $u$, where $F$ is the set of possible factors in general such as password, OTP (One-Time Password), and biometric data. Let the authentication succeed if the following is satisfied:

$$\text{Auth}(u, c, F_u) = \begin{cases} 1 & \text{if } \forall f \in F_{\text{required}}, \text{Verify}(u, f) = 1 \\ 0 & \text{otherwise} \end{cases}$$

Here, $F_{\text{required}} \subseteq F$ is the subset of the factors that are required for authentication. Strong password policies dictate that passwords $p$ must follow a set of complexity rules. For a password $p \in P$ to be valid:

$$\text{Valid}(p) = \begin{cases} 1 & \text{if length}(p) \geq 12 \text{ and contains required character types} \\ 0 & \text{otherwise} \end{cases}$$

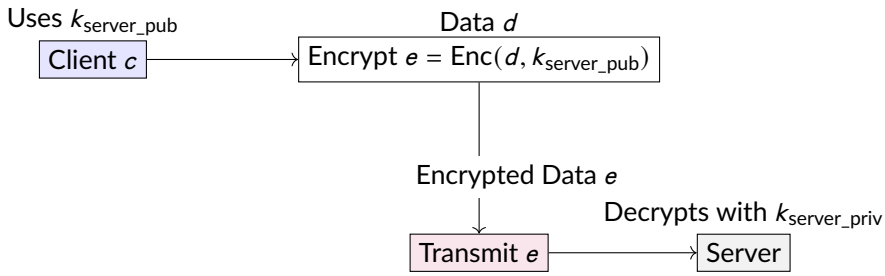


**Figure 4.** Encryption Process for Secure Data Transmission

The system integrates with PCI DSS-compliant payment gateways through secure APIs. Tokenization is used to replace sensitive payment information $d \in D$ with a non-sensitive equivalent $t_d \in T$, defined as:

$$\text{Tokenize}(d) = t_d \quad \text{such that } t_d \notin D$$

By doing this, actual payment information is not directly processed or stored by the system, which reduces the circle of exposure in case of any breach.

Data protection in the client module is provided through some form of encryption when data is in transit. For any data $d$ to be sent from client $c$ to a server, encryption can be described as:

$$e = \text{Enc}(d, k_{\text{server\_pub}})$$

where $e$ is the encrypted data and $k_{\text{server\_pub}}$ denotes the server's public key used for encryption. TLS 1.3 with strong cipher suites is used to offer confidentiality and integrity of the data. It doesn't forget to enforce HSTS, ensuring all browsers can speak only over secure HTTPS connections by eliminating protocol downgrade attacks. It implements a Content Security Policy, CSP, which limits resources being loaded by the user's browser to ensure integrity while reducing risks of cross-site scripting.

The client module follows the best practices of secure coding to minimize the possibility of common, web application-related vulnerabilities. The input validation functions defined earlier ensure that user-submitted data conforms to predefined rules before processing can occur, thus preventing attacks such as SQL injection, cross-site scripting, and cross-site request forgery. Error handling routines are also set up to prevent the disclosure of sensitive system information that could be leveraged by an attacker.

Device security is ensured by trusted platform verification processes. The security of a client device $c$ is assessed using a verification function:

$$\text{DeviceSecure}(c) = \begin{cases} 1 & \text{if device } c \text{ meets all security standards} \\ 0 & \text{otherwise} \end{cases}$$

Only through this verification are devices allowed to access sensitive functionalities, thus ensuring that indeed, compromised or non-compliant are prevented from interacting with parts of a system.

Data privacy: Data privacy is ensured by collecting only the minimal necessary data $D_{\text{minimal}}$ from users in the process of an interaction between the user and the system. For each request to collect data $D_c \subseteq D$, the following constraint holds:

$$|D_c| \leq |D_{\text{minimal}}|$$

This would mean that through data minimization principles, when storing and processing large volumes of data about individuals, there is less risk of data breaches. This puts a restriction to ensure the system acquires and processes data in no more than the minimum amount necessary for operational functions, ensuring compliance with data protection regulations.

## 5 Communication Module ($\mathcal{M}$)

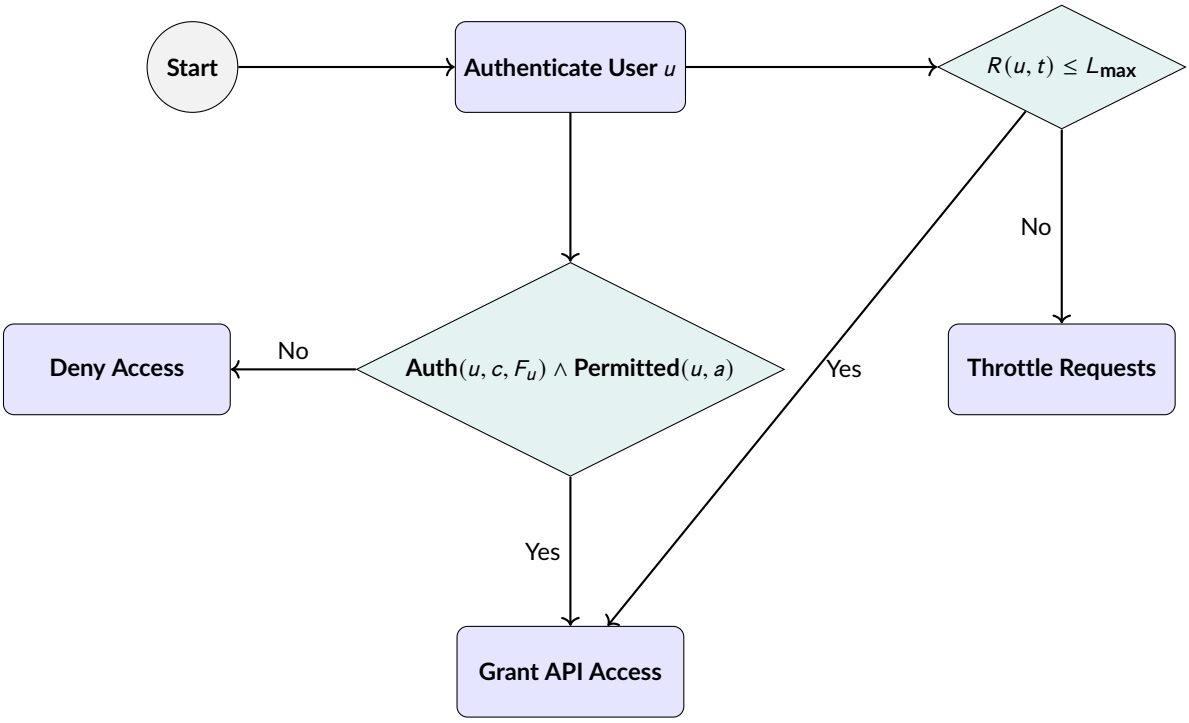


**Figure 5.** API Request Validation and Rate Limiting Process

The communication module, $\mathcal{M}$, is responsible for the secure interaction between a client and a server with external entities like a payment processor. It plays a vital role in keeping API security tight, communication encryption, deploying firewalls, intrusion detection, and the like for the integrity, confidentiality, and availability of data within an e-commerce platform.

Any incoming request from the client for any transaction or request regarding user information will go through this secure API Gateway. This gateway will ensure that all the API interactions are

secured by OAuth 2.0 and OpenID Connect when it comes to authentication and authorization. A function verifies each API request $a \in A_{\text{API}}$:

$$\text{AC}(u, a) = \text{Auth}(u, c, F_u) \wedge \text{Permitted}(u, a)$$

where $\text{AC}(u, a)$ represents the access control decision for user $u$, based on successful authentication $\text{Auth}(u, c, F_u)$ and permissions. This ensures that only authorized users can access certain API endpoints, enhancing the security of sensitive data transactions. To prevent abuse through excessive API calls, rate limiting and throttling are employed, restricting the number of requests per user or IP address. The rate limiting function is defined as:

$$R(u, t) \leq L_{\text{max}}$$

where $R(u, t)$ is the count of requests from user $u$ within time period $t$, and $L_{\text{max}}$ is the maximum threshold. Requests exceeding $L_{\text{max}}$ are throttled, preventing denial-of-service attacks and resource exhaustion. API inputs are sanitized and validated to protect against injection attacks, ensuring that only compliant data is processed.

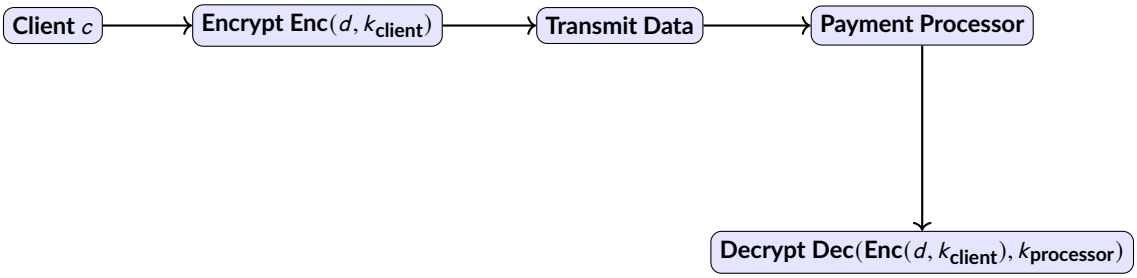


**Figure 6.** End-to-End Encrypted Communication with Payment Processor

It allows interaction with external payment processors and financial institutions through the running of operations of the module, which are handled through encrypted communication channels. Included in these are E2EE, allowing assurance of data $d \in D$ being kept encrypted from a point of transmission at the client to a point of decryption at the processor of the information. This is defined by:

$$\text{Enc}(d, k_{\text{client}}) \xrightarrow{\mathcal{M}} \text{Dec}(\text{Enc}(d, k_{\text{client}}), k_{\text{processor}})$$

where $k_{\text{client}}$ and $k_{\text{processor}}$ are the encryption and decryption keys, respectively. This interaction is further secured by Mutual TLS authentication, which verifies both the identities of the client and the server against impersonation attacks. The communication module enforces periodic key rotation to minimize the time that any given encryption key is exposed, reducing the possible impact if a key becomes compromised.

Network security is improved by firewalls and intrusion detection systems (IDS). A web application firewall (WAF) filters web traffic, aiming at blocking web-oriented attacks, such as SQL injection and cross-site scripting (XSS). Fire wall rules are defined over a set of allowed network connections $N_{\text{allowed}} \subseteq N$, where $N$ is the set of all possible network connections:

$$\forall n \in N, \text{Allow}(n) = \begin{cases} 1 & \text{if } n \in N_{\text{allowed}} \\ 0 & \text{otherwise} \end{cases}$$

This set of rules allows only the whitelisted connections and blocks all unauthorized ones so that only the trusted data sources can communicate with the e-commerce platform. IDS monitors network traffic for signs of anomalous activities. Anomaly detection is done by using machine learning models to learn deviation from established patterns that may indicate attempted fraud or breaches.

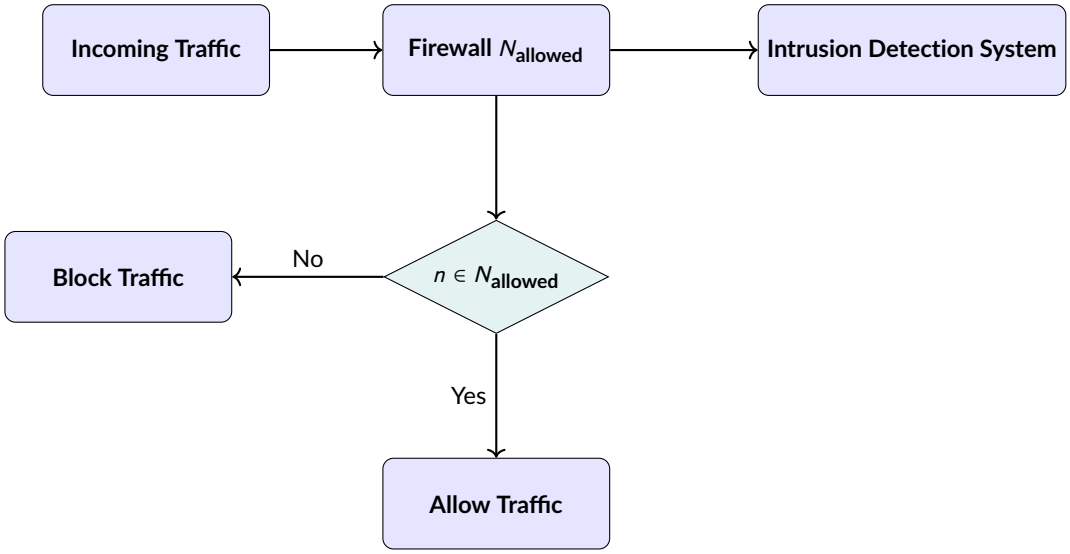


**Figure 7.** Firewall and Intrusion Detection System Process

Similarly, encrypted communication and API security measures are complemented by a robust logging and monitoring system. Every event $e \in E$ is recorded through a logging function:

$$\text{Log} : E \rightarrow L$$

where $E$ is a set of events such as login attempts, transaction requests, and security alerts, and $L$ is the set of logs generated. The logs are detailed records of activities involving the system. These enable both real-time monitoring and retroactive analysis in case any incident related to security has taken place. Such tracking details help identify the attack vectors, hence informing better defense strategies for the future.

## 6 Response Generation Module ($\mathcal{R}$)

It includes the response generation module $\mathcal{R}$, which communicates with users through the AI engine about personalized information on fraudulent activities, access control, and security. By definition, the response generation module does this by making product recommendations and practicing security protocols without revealing any sensitive user information.

The AI internally processes customer inputs to provide personalized responses, recommendations of products, or help with other enquiries. In practice, AI is provided with only the data strictly necessary for providing the answer. Therefore, to ensure AI works securely, the AI is made subject to data minimization, where access is restricted to just that data which is needed for response generation. For a dataset $D$, the AI-accessible subset $D_{\text{AI}} \subseteq D$ is restricted by:

$$|D_{\text{AI}}| \leq |D_{\text{minimal}}|$$

where $D_{\text{minimal}}$ denotes the minimum amount of data necessary for the AI to function as intended. For privacy-preserving methods, such as homomorphic encryption, the AI will receive inputs

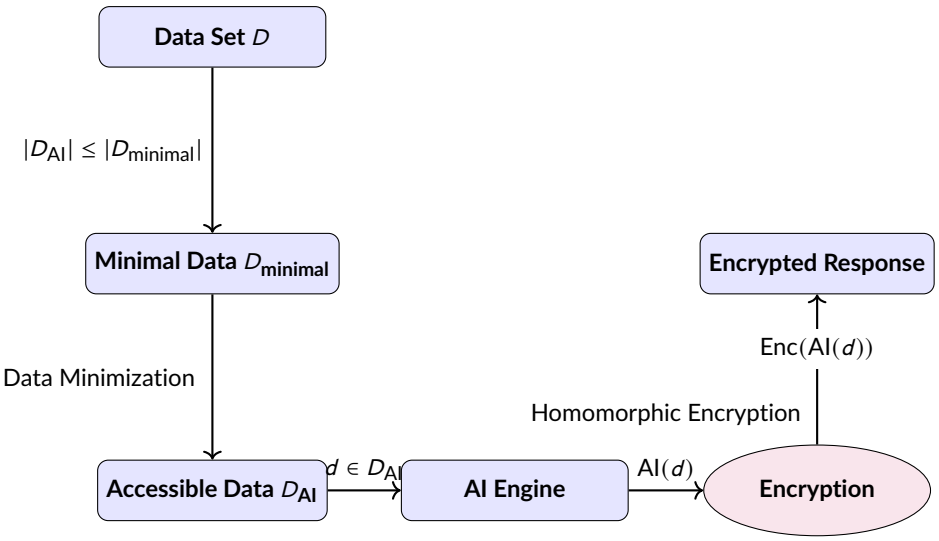


**Figure 8.** Data minimization process for the AI engine with encrypted response generation. Data accessible by the AI is restricted to a minimal subset, ensuring privacy through homomorphic encryption.

$d \in D$ in encrypted form only; no actual content is disclosed. The generation of responses in encrypted fashion may be set as:

$$\text{AI}(\text{Enc}(d)) = \text{Enc}(\text{AI}(d))$$

This ensures that even during the processing of data, sensitive information remains secure. Regular security audits are performed on the AI engine, testing for any kind of vulnerability or exploit that could compromise user data or the integrity of generated responses.

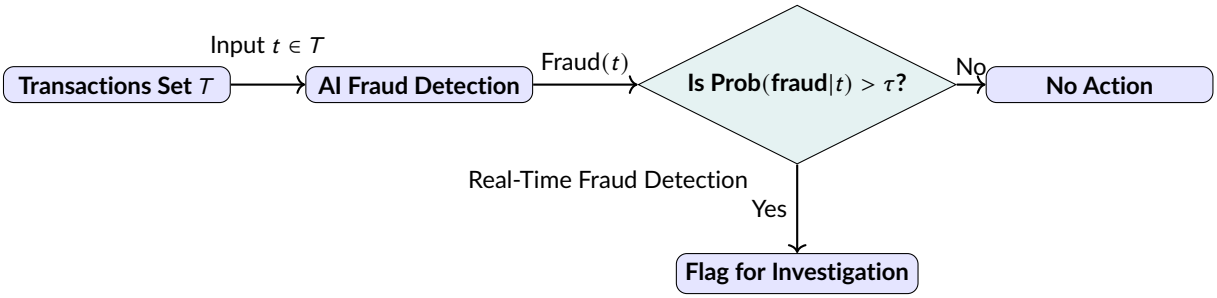


**Figure 9.** Real-time fraud detection process using AI. The AI engine evaluates each transaction for fraudulent behavior, flagging those that exceed the probability threshold $\tau$.

Fraud detection is one aspect of the response generation module. With models in AI, the system would be trained for fraud pattern detection during a transaction or user interaction. Let $T$ be the set of transactions and $\text{Fraud}(t)$ denote the likelihood that a transaction $t \in T$ is fraudulent. Realtime detection applies a function:

$$\text{Fraud}(t) = \begin{cases} 1 & \text{if Prob}(\text{fraud}|t) > \tau \\ 0 & \text{otherwise} \end{cases}$$

where $\tau$ is some predefined threshold. If this fraud probability exceeds $\tau$, the system flags t for further investigation. Besides regulatory compliances with GDPR and PCI DSS, it makes the data

handling practices compliant with legal obligations. There are content filtering mechanisms in place to prevent AI from generating or transmitting responses that contain sensitive personal or financial information. For any response $r \in R$, a filtering function is performed to ensure:

$$r' = \text{Filter}(r) \quad \text{where } r' \subseteq R \text{ meets content policies.}$$

The system provides more advanced access control features to ensure the protection of the AI engine and sensitive e-commerce data against unauthorized access. RBAC assigns permissions based on user roles. Let Roles $= \{r_1, r_2, \ldots\}$ be the set of roles and Permissions : Roles $\rightarrow 2^A$ be a function mapping roles to their respective permissions. Access to an action $a \in A$ is granted provided:

$$a \in \text{Permissions}(\text{Role}(u))$$

where $\text{Role}(u)$ is the role assigned to user $u$. This will, in turn, enable the imposing of restrictions on the use of functionalities based on the role a user plays within the system. For extended access control, Attribute-Based Access Control will be utilized, which factors in attributes such as time, location, or device type in its decisions. Continuous authentication ensures re-verification of user identity over regular intervals during the interaction session, hence making session hijacking or unauthorized access less likely.

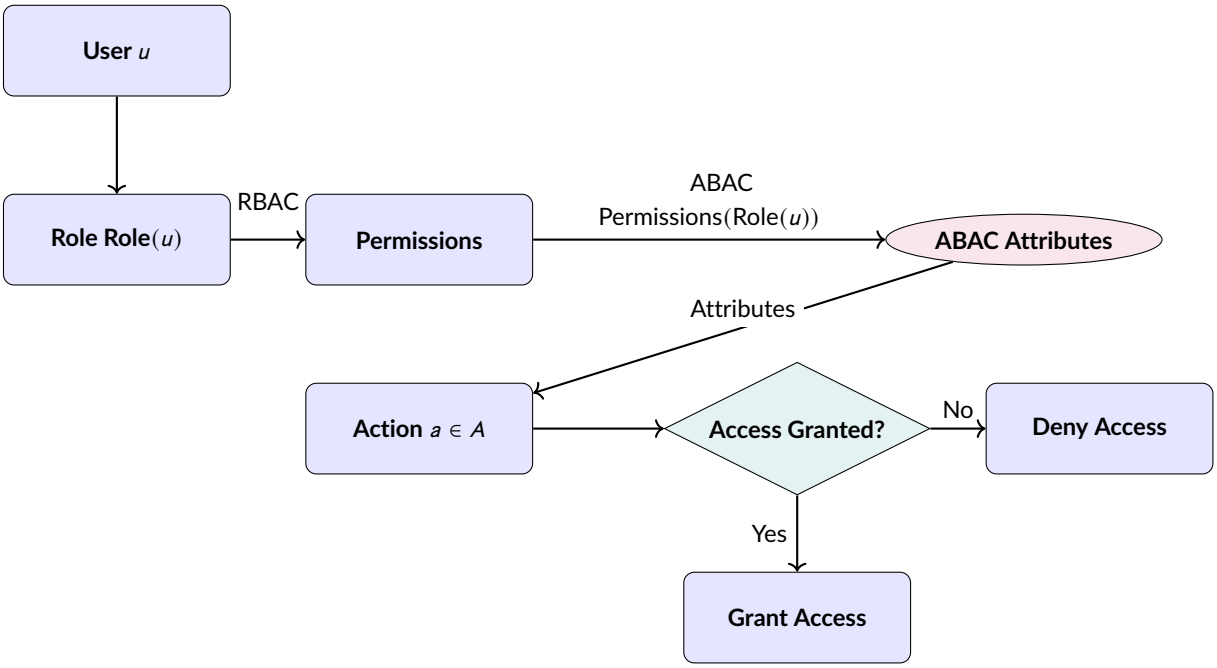


**Figure 10.** Role-Based and Attribute-Based Access Control (RBAC and ABAC). Users are assigned roles, and their access is controlled through permissions and additional attributes. Access to an action is granted based on RBAC and ABAC policies.

The response generation module ensures that the AI model remains resilient against input manipulations. For any input $i$ and perturbation $\delta i$ with acceptable bounds $||\delta i|| < \epsilon$, the bound of difference in outputs can be described as :

$$||\text{AI}(i) - \text{AI}(i + \delta i)|| < \delta$$

where $\epsilon$ defines the maximum allowed perturbation and $\delta$ represents the threshold for the output variation. Such a constraint will ensure that small changes in inputs will not yield significantly dif-

ferent responses, which essentially is called for to ward off adversarial attacks that may manipulate the outputs of an AI.

This module also incorporates non-stop monitoring and auditing functionalities for higher traceability and accountability. Any transaction $t \in T$ is traceable by means of the audit function:

$$\text{Audit} : T \rightarrow A_{\text{audit}}$$

where $A_{\text{audit}}$ is the set of audit logs generated for transactions. These logs include information on access attempts, data processing activities, and suspicious behavior, enabling thorough analysis and review in the event of a security incident.

## 7 Database Module ($\mathcal{D}$)

The database module $\mathcal{D}$ is used within the e-commerce system to securely store customer information, transaction details, and sensitive financial data. It has been designed to provide confidentiality, integrity, and availability of stored data through advanced encryption techniques, strict access management, and comprehensive activity monitoring.

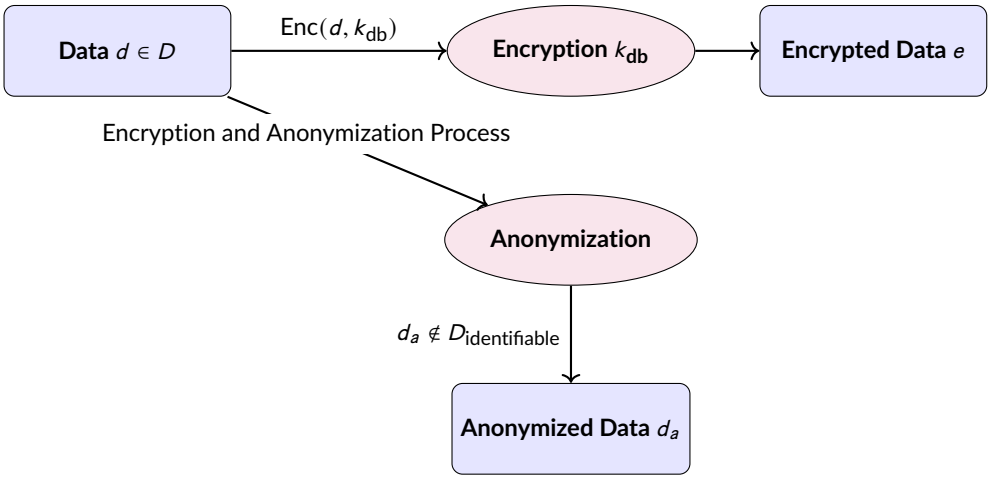


**Figure 11.** Database module: Data encryption and anonymization process. Using sensitive data, encryption by a key $k_{\text{db}}$ is performed; personally identifiable information is anonymized in such a way that it becomes non-identifiable.

Customer and transaction data storage needs to be provided with strong mechanisms for encryption in order to avoid unauthorized access to sensitive information. Data encryption at rest: All data at rest, $d \in D$, should be protected by using a strong encryption algorithm like AES-256. The encrypted form $e$ of data $d$ will be computed as:

$$e = \text{Enc}(d, k_{\text{db}})$$

Where $k_{\text{db}} \in K$ is the key used to encrypt the database. Given this, TDE further enhances this by allowing full-disk encryption of the database, its backups, and transaction logs so that unauthorized access to the physical database files cannot lead to data compromise. In addition, other data protection measures such as anonymization and pseudonymization are performed to minimize risks regarding the storage of personally identifiable information. It transforms data $d$ into anonymized data $d_a$ such that $d_a$ satisfies:

$$d_a \notin D_{\text{identifiable}}$$

This makes the reconstruction of personal identities rather tricky in nature from stored data.

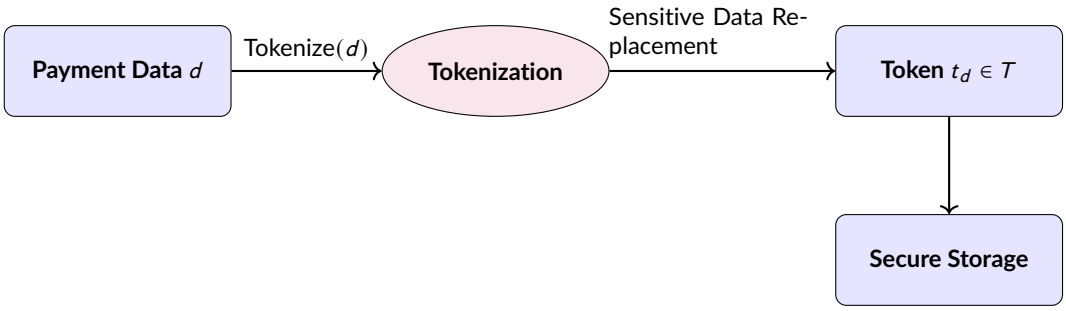


**Figure 12.** Tokenization process of sensitive financial data in the database. The payment data $d$ is replaced by a token $t_d$, thereby reducing the direct exposure of sensitive information while in storage.

Advanced encryption methods of the database module are applied only to financial data. The payment information is tokenized along with other sensitive records by replacing the real data with non-sensitive tokens $t_d \in T$. Tokenization can ensure that:

$$\text{Tokenize}(d) = t_d \quad \text{such that } t_d \notin D$$

This limits the exposure of sensitive data both during storage and while processing. Now the database can handle transactions without raw payment details. The proposed system implements KMS in order to securely manage and store keys $k \in K$, which manages access to encrypted data. As a good practice, keys should be changed after a certain amount of time to minimize the damage caused by key compromise. In addition, all database backups are encrypted using keys $k_{\text{backup}} \in K$ before they are written to the backup media, so that backups have equal protection to the mother database.

The database module uses the principle of separation of duties and the least privilege for access control. A user $u$ should only have access to the data and functions that are strictly needed by them as decided by their role, and it is defined in the following way:

$$\text{AC}(u, a) = \begin{cases} 1 & \text{if Need}(u, a) = 1 \\ 0 & \text{otherwise} \end{cases}$$

where $\text{Need}(u, a)$ determines whether a user $u$ needs access to perform action $a$. This approach limits access to sensitive data fields, thereby minimizing the possibility of unauthorized data exposure and reduces the vectors of probable attack. Also, role-based permissions ensure that the administrative functions are kept separate, so access with high privileges is granted only to a minimum number of users.

A checksum function provides data integrity, which means it checks for the integrity of the data stored and retrieved. For any data $d$ stored in the database, a hash value $h \in H$ is computed using a checksum function:

$$\text{Checksum}(d) = h$$

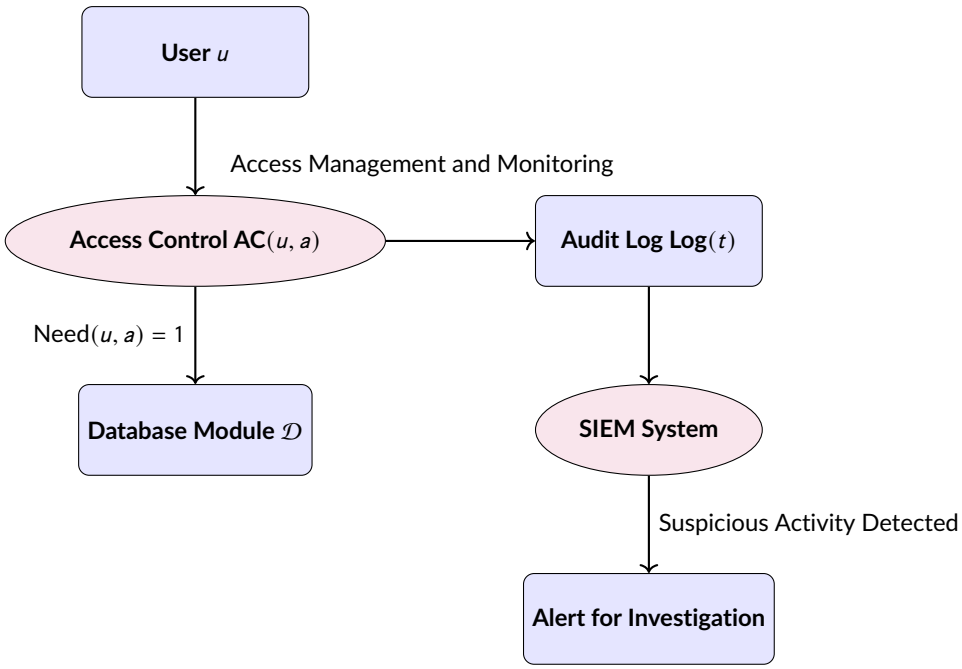


**Figure 13.** Access control and activity monitoring of the database module. The access of each user is based on the principle of least privilege, while audit logs and SIEM systems provide security threat interactions.

Data integrity is verified when:

$$\text{Checksum}(d_{\text{stored}}) = \text{Checksum}(d_{\text{retrieved}})$$

This process ensures that any tampering or corruption of data during storage or transmission is detected, preserving the accuracy and reliability of the information.

Database activity monitoring is integral to the security framework of the database module. Audit trails maintain comprehensive logs of all database interactions, recording user activities such as data retrievals, updates, and administrative actions. For each transaction $t \in T$, an audit log Log$(t)$ is created, containing details of the user, timestamp, and nature of the interaction. Real-time alerts are generated using Security Information and Event Management (SIEM) systems, which analyze logs to identify suspicious activities or deviations from normal behavior patterns. When a potential threat is detected, the system triggers an alert for immediate investigation. This monitoring process is essential for detecting unauthorized access attempts and responding to security incidents promptly.

Regular compliance audits ensure that database practices align with regulatory requirements such as PCI DSS for payment data and GDPR for user privacy. These audits involve evaluating encryption practices, access controls, and data handling procedures, verifying that the e-commerce platform adheres to mandated security standards. Compliance audits help maintain a secure database environment, reducing legal risks and building trust with users by ensuring that their data is managed securely.

## 8 Integration and Data Flow in E-Commerce

The integration scope is described as a structured flow of data across modules for security and efficiency from the very first step that the customer takes. It involves user interaction where customers interact with the e-commerce through the Client Module, which provides the customer with the ability to view available products, manage their shopping carts, and create orders. In such interactions, the authentication schemes at the backend, such as MFA and secure

password policies, validate a customer's identity to ensure that only authorized users gain access. Information that could be prone to leakage, such as credentials about login or payment details, is encrypted before transmission, so that even upon interception, such information shall not be accessible unless the cryptographic keys are known.

Once the data gets ready to transfer, the Communication Module does its work in ensuring that the transfer gets executed securely with encrypted information. It provides secure APIs that handle data exchange between the client and the server in a way that no customer data is compromised in the process of communication. Mutual TLS and end-to-end encryption have been implemented for securing the channel of communication with external payment processors so that the integrity of data is maintained right from the client's system to the payment gateway. Such an approach ensures that any data cannot be intercepted or tampered with in the process of a transaction.

It then takes the customer's data input and, with the help of the Response Generation Module, creates personalized shopping experiences. It analyzes user interactions to provide personalized product recommendations and assist with customer inquiries. Real-time fraud detection systems integrated into the module continuously monitor transactions and interactions for a pattern indicative of fraud. In such a case, if a potential fraud is detected, the system flags the transaction for further review. Besides this, the AI engine involved in this module operates under strict access control, and therefore, sensitive customer data reaches only the respective components of the system that have been allowed to do so for this type of operation, hence minimizing leakage chances.

Response generation is then safely stored in the Database Module for customer and transaction data. Advanced encryption methods and key management protect data at rest in this database, even in cases when unauthorized physical access to its storage took place. The encryption keys are managed by key management systems in an extremely secure form. Rotation of keys is done in a manner that minimizes exposure to risk. The database activities are continuously monitored for compliance with regulatory standards such as PCI DSS, and audit logs of access attempts and changes to the data are recorded, thus allowing full transparency and traceability for data handling.

The process concludes by sending the secured responses back to the client. The responses will include an order confirmation, personalized recommendations, or a transaction status update. It stays encrypted all the way through its transmission and is decrypted only when the response reaches the Client Module. In that way, sensitive information will not be disclosed during the response process, thus keeping data confidential from one end to another. The integrative flow of data across these modules ensures the framework considers security, usability, and compliance in ensuring efficient operability of an e-commerce platform in the best interest of gained user trust.

## 9 Implementation Considerations

Modularity of integration is needed regarding the integration of new components into existing e-commerce systems, where each module has to be designed to act on its own without preventing interaction with other parts of the system. This will surely pay off during maintenance and scaling farther, when updating or changing certain components will not destroy the operation of the whole. Scalability in general demands that the performance of the system be preserved within secure e-commerce environments, balancing security against efficiency. Besides, cryptographic operations are efficiently implemented where acceleration by hardware can be applied. This accelerates every encryption and decryption process. Through the application of load balancing strategies, the processing load is distributed across many servers, reducing the likelihood of any bottleneck by making sure that no single point is overwhelmed. Again, the use of protected caching strategies will improve response times for frequently accessed data quite well, while making sure that information cached remains protected against unauthorized access.

The best practices for key management are essential in maintaining the security of data throughout its life cycle. Secure storage of keys is normally handled through Hardware Security Modules or

secure key vaults that offer a highly controlled environment for key storage. Access controls are strictly ensured; only authorized persons can access the keys to minimize the possibility of key misuse or exposure. It ranges from secure key generation and distribution to the regular rotation of keys with regard to minimizing risks of their compromise, up to secure destruction of the keys that are no longer needed.

In such a context, proactive measures will be able to ensure a secure environment and prompt responses in case of all kinds of threats. Setting up a SOC requires a committed team for continuous system monitoring, security alert analysis, and the necessary responses. Automation in this context greatly helps with automated tools in threat detection and response of security incidents that will reduce the time taken for responses and reduce any further damage. A well-articulated incident response plan stipulates the activities required to manage security breaches or anomalies. In that way, the organization is able to respond promptly in a way that minimizes the impact of security incidents.

## 10 Conclusion

Deployment of conversational AI in e-commerce has facilitated customer engagement and smoothened the transactions. The result is new lines of security threats-from unauthorized access to sensitive customer information to fraudulent transactions that may threaten the integrity of financial systems. These, in turn, render the customer data vulnerable and demand an effective and complete security framework so as to impose compliance regulations such as PCI DSS and GDPR. To handle these challenges, an architectural model for securing conversational AI in e-commerce is proposed in this paper by using advanced encryption techniques, secure authentication, and continuous monitoring.

It is designed based on four major components: Client Module, Communication Module, Response Generation Module, and Database Module. Each module works in the perspective of satisfying certain security needs and some inherent vulnerabilities within e-commerce systems. The Client Module manages interactions of a user with the system for secure access and data handling. Encryption and transmission of client-server data by the Communication Module, where sensitive transactions will be executed with secure channels. Response Generation Module: It makes the data and customer queries secure with effective processing integrated with fraud detection capabilities. Finally, the Database Module secures customer and transaction data by encryption, access control, and activity monitoring.

The Client Module involves a user interface, different mechanisms for authentication, and encryption layers to securely handle the user's data in the transaction process. This interface is designed to enable secure interactions with respect to browsing, selecting a product, and checkout. Sessions could be managed securely using secure cookies and session tokens to counter session hijacking, with mechanisms for input validation to protect against injection attacks that guarantee the integrity of the system from malicious input.

Authentication is critical in the Client Module, where the account and transactions of the user concerned must be closely guarded against access by unauthorized persons. The model uses multi-factor authentication, meaning that several means of identification must be presented by the user, and this reduces unauthorized access. Strong password policies impose complexity requirements and periodic updates, thus increasing the security. In the model, the payment transactions are integrated with secure electronic payment gateways that comply with PCI DSS in order to protect sensitive financial information of customers at all touch points of payment processing. Tokenization techniques replace sensitive payment details into tokens, hence keeping risk exposure to a minimum.

Data protection in the Client Module is ensured through robust encryption of data. TLS is used for encrypting the data that is in transit between the user and server all the way to ensure confidentiality and integrity. It sets a prevention against protocol downgrade via HTTP Strict Transport Security, while Content Security Policy restricts the loading of resources that may be malicious to mitigate cross-site scripting attacks. These encryption and protection measures

altogether ensure that user data is secure from the point of input to the point of transmission.

The Communications Module is responsible for the security of data exchange between the client and server, handling API interaction, and offering the security of communication with third-party services that include payment processors. The module will make use of a secure API gateway, which acts in centralizing the management of the requests at the client's end and provides a single source of authentication and access control. OAuth 2.0 and OpenID Connect protocols are implemented to add an extra layer of security to the API requests where the specific clients are allowed to use only particular services. Protection against injection attacks on the servers, which could compromise their integrity, is provided through the validation and sanitation of API data inputs.

E2EE encryption at the channel level secures the communication channel through the inaccessibility and unreadability of data to unauthorized parties along its whole delivery path. This prevents any interception or tampering of information exchanged between the e-commerce system and any other entity, such as banks or payment gateways. This gets further strengthened by mutual TLS authentication, which validates the identity both at the client and server ends before any communication begins, hence avoiding impersonation attacks. Rotation of encryption keys on time ensures that, in case any key gets compromised, the window of vulnerability remains very small.

Firewalls and IDS improve the network security level of the Communication Module. WAFs do detect and prevent some of the most common web-based attacks, like SQL injections and cross-site scripting by filtering and monitoring HTTP traffic. Intrusion detection systems supplement these with constant network traffic monitoring for suspicious activities, and they allow for quick responses to real or perceived threats. With the development of anomaly detection systems, machine learning algorithms can identify patterns that are out of the ordinary and hence provide extra security against certain sophisticated kinds of attacks.

Response Generation Module: This is tasked with the security of the AI system that interfaces with customers through response generation, recommendations, and answering questions. It processes user data in such a way that during the interaction, the privacy and integrity of the data are maintained. In this module, the AI engine configuration is minimalistic for the data access approach, enough to produce valid and contextual results for the exposure of data to be restricted to the limits necessary to be at risk for leakage or unauthorized use in certain cases.

Further enhancing this privacy, the deployment of various privacy-enhancing techniques, such as data anonymization, makes it possible for the AI to process information without exposing personally identifiable information directly. Regular security audits are made in understanding the vulnerabilities of AI algorithms and hence keep the system resistant to adversarial attacks that would like to manipulate the system's behavior. It will also include assessments regarding an AI's ability in terms of variation in input and the consistency and security of outputs under different conditions.

It is also part of the Response Generation Module where fraud detection, via real-time analysis of user interactions, uncovers suspicious behaviors that might point to fraudulent activities. By integrating the fraud detection systems using AI capabilities, the e-commerce platform identifies patterns typical for fraud attempts, such as unusual purchasing behaviour or abnormal login locations. The transactions and interactions will be monitored by the module to ensure that requirements by regulatory bodies such as GDPR are observed through the imposition of strict data handling protocols.

Access control mechanisms in this module will ensure that sensitive AI data and functions are accessible only to the authorized staff. Events specific to RBAC entail assigning users in particular roles with permissions which provide them limited access to data and functionality based on the position that a user holds within an organization. This is complemented by ABAC-considering factors such as time of access, device location, and thus extending more granular control on data access. Continuous authentication methods ensure that the identity of a user is checked at periodic intervals during a session to reduce risks of session hijacking.

The Database Module has the task of securely storing and managing all customer and transaction data. The sensitive information is protected from unauthorized access and breaches by encryption techniques, mechanisms of access control, and activity monitoring. Data encryption at rest is used through the use of strong encryption standards, including AES-256, that ensure even in the compromise of storage media, customer data remains protected. TDE is utilized in the offering of an additional layer of security to stored information through the automatic encryption of data in databases.

This module also points out the necessity of secure data backups through the encryption of backup databases to prevent unauthorized access. In this way, it is guaranteed that such data will be recoverable and not lost should an event of system failure or data loss occur. Further, personal identifiers are removed or altered through data anonymization and pseudonymization, reducing resultant risks in case of exposure of a data breach event.

One of the roles this module can enter is database activity monitoring, which would imply that logging systems monitoring every interaction with the database log activities for compliance and security reasons. Such logs allow for detailed audit trails, which are crucial in tracking security incidents and ensuring compliance to regulations such as PCI DSS. The SIEM systems analyze logs in real time and show alerts for any anomaly or suspicious activity that might point toward a breach. Regular compliance audits are performed, and database security practices follow needed standards where gaps identified are addressed promptly.

This integration of these modules will ensure smooth data flow on one hand while ensuring high-level security. Customer data is first processed at the Client Module when any interaction between the customer and the electronic commerce platform has occurred. Further, the data is authenticated and encrypted for transmission through the Communication Module. This processed data will then be used by the Response Generation Module for personalized responses, while it is stored and maintained securely at the Database Module. This architecture protects data through its entire life cycle from initial input to storage and retrieval with the integrity and confidentiality of customer interaction.

Additional measures include 3D Secure authentication, which further enhances the security of the payment process by adding an additional layer of verification at the time of making an online payment. Payment tokenization, in the case of an EMV payment, replaces raw card numbers with tokens, hence minimizing the risks involved in handling sensitive information related to payments. This, together with strong encryption of data both in transit and at rest, secures the information against interception and unauthorized access.

Analysis of behavioral biometrics further empowers fraud detection systems to identify user behavior not acting in concert with typical activities. Device fingerprinting adds another layer of security to identify devices used in transactions and helps in the detection of suspicious activities linked to changes in devices. Whitelists of known fraudulent entities and lists of trusted users automatically block or approve the transactions to further streamline fraud prevention measures.

Proper incident response is very important in order to lessen the impact of a security breach. In this case, a specific incident response team can make sure that the security incident will be taken care of in the quickest possible time without any delay. Pre-defined procedures for responses related to different types of e-commerce threats will help in understanding what to do in case a particular breach occurs. Communications plans will make sure that proper notification of a certain security incident reaches customers, authorities, and stakeholders. It is also covered under cyber insurance policies upon losses that may result from data breaches and other incidents, which gives the organization some degree of financial security against such losses.

The proposed security model of Conversational AI systems in e-commerce has various limitations that may influence the practical effectiveness and applicability of the model. These limitations may include, but are not limited to: challenges brought about by computational overhead, the nature of cyber threats, and complexity issues arising from compliance within various jurisdictions.

Advanced encryption techniques, multi-factor authentication, and secure communication proto-

cols will naturally bring significant computational overhead to a system. For example, the TLS protocols for encrypted communication at transport layers, or even the end-to-end encryption mechanisms, demand more processing power and could introduce latency in data transfer. Similarly, the processes of multi-factor authentication and continuous session verification require repeated checks that can add latencies in user interactions. This may impede the user experience in high-traffic e-commerce environments where response time is critical. Further, ongoing monitoring and logging of database activities and system events for compliance and security analytics require storage and processing resources that, in turn, may slow down the overall system performance. Therefore, security needs and responsiveness of the system have to be balanced, and further research might be done in order to optimize those processes in order to decrease latency without compromising security.

The fact that cyber threats change dynamically significantly diminishes the long-term efficiency of the proposed security model. Despite the fact that the framework embeds different means for advanced security, most of them are designed to react to known attack vectors and patterns. Cybercriminals work continuously on finding new modes of exploitation, such as advanced phishing attacks, zero-day vulnerabilities, and APTs, which sometimes are too tricky to be detected by already-in-place detection algorithms. All this reliance on machine learning models for fraud and anomaly detection also has its downside: they require updates and retraining continuously in order to retain their accuracy at detecting new threat patterns. However, in real life this is resource-intensive and may not be possible for all e-commerce given their meager cybersecurity budget. The inability to adapt fast enough by the proposed model maybe leaves the system open to other vulnerabilities during that interim period before updates can be done.

The main challenge that arises in deploying the security framework, which is compliant with various regulatory standards like PCI DSS and GDPR, varies across different geographical regions since the legal requirements vary from region to region. While the model hence includes data anonymization, encryption, and access control mechanisms to meet some compliance requirements, complexities arise when adapting those measures to be in line with particular local regulations that may impose other restrictions or differently define data privacy and security. For example, GDPR is extremely oriented toward individual rights over their data, such as access, rectification, and erasure, which calls for the implementation of features related to data subject access requests. In comparison, other regulations like the California Consumer Privacy Act prescribe different requirements with respect to data processing, which can also be a source of inconsistencies in data management practices across different regions. This would also mean that adapting the security model to comply with various, sometimes conflicting, regulatory frameworks could further complicate the design of the system and operational processes. This complexity may necessitate additional resources for legal consultation, audits, and updates of changes related to data handling protocols, which in turn could affect the viability of the model in terms of global e-commerce platforms. This could only be surmounted if the compliance framework was flexible and adaptable to new changes, usually a resource-consuming process that may take a long time to develop and maintain.

## References


[1] May P. The business of ecommerce: From corporate strategy to technology. vol. 1. Cambridge University Press; 2000.

[2] Datta P. A preliminary study of ecommerce adoption in developing countries. Information systems journal. 2011;21(1):3-32.

[3] VanHoose D. Ecommerce economics. Routledge; 2011.

[4] Raisch W, Milley D, Kane WJ. The eMarketplace: Strategies for success in B2B eCommerce. McGraw-Hill, Inc.; 2001.

[5] Trepper C. ECommerce Strategies. Microsoft Press; 2000.

[6] Molla A, Licker PS. eCommerce adoption in developing countries: a model and instrument. Information & management. 2005;42(6):877-99.

[7] Steward S, Callaghan J, Rea T. The eCommerce revolution. BT technology journal. 1999;17:124-32.

[8] Reynolds J. eCommerce: a critical review. International Journal of Retail & Distribution Management. 2000;28(10):417-44.

[9] Cassell J. Embodied conversational agents: representation and intelligence in user interfaces. AI magazine. 2001;22(4):67-7.

[10] Inkster B, Sarda S, Subramanian V, et al. An empathy-driven, conversational artificial intelligence agent (Wysa) for digital mental well-being: real-world data evaluation mixed-methods study. JMIR mHealth and uHealth. 2018;6(11):e12106.

[11] Li R, Ebrahimi Kahou S, Schulz H, Michalski V, Charlin L, Pal C. Towards deep conversational recommendations. Advances in neural information processing systems. 2018;31.

[12] Maedche A, Legner C, Benlian A, Berger B, Gimpel H, Hess T, et al. AI-based digital assistants: Opportunities, threats, and research perspectives. Business & Information Systems Engineering. 2019;61:535-44.

[13] Vinyals O, Le Q. A neural conversational model. arXiv preprint arXiv:150605869. 2015.

[14] Bezovski Z. The future of the mobile payment as electronic payment system. European Journal of Business and Management. 2016;8(8):127-32.

[15] Lai P. Design and Security impact on consumers' intention to use single platform E-payment. Interdisciplinary Information Sciences. 2016;22(1):111-22.

[16] Trautman LJ. E-Commerce, cyber, and electronic payment system risks: lessons from PayPal. UC Davis Bus LJ. 2015;16:261.

[17] Yenisey MM, Ozok AA, Salvendy G. Perceived security determinants in e-commerce among Turkish university students. Behaviour & Information Technology. 2005;24(4):259-74.

[18] Furnell SM, Karweni T. Security implications of electronic commerce: a survey of consumers and businesses. Internet research. 1999;9(5):372-82.

[19] Kim C, Tao W, Shin N, Kim KS. An empirical study of customers' perceptions of security and trust in e-payment systems. Electronic commerce research and applications. 2010;9(1):84-95.

[20] Dewan SG, Chen Ld. Mobile payment adoption in the US: A cross-industry, crossplatform solution. Journal of Information Privacy and Security. 2005;1(2):4-28.