H4H

Owner:
Reviewer:
Contributors:
Date Generated: Wed May 28 2025

Executive Summary

High level system description

Not provided

Summary

Total Threats	26
Total Mitigated	15
Not Mitigated	11
Open / High Priority	5
Open / Medium Priority	6
Open / Low Priority	0
Open / Unknown Priority	0

DFD-H4H

Visit Scheduler (Process)

Description: FastAPI App

Number	Title	Type	Priority	Status	Score	Description	Mitigations
11	Forged user ID in requests	Spoofing	High	Open		Attackers could manipulate request bodies to act on behalf of another vendor/client and tampering with the issuing of visits that they book/accept and e.g. pay a deposit for.	
12	ElasticSearch query manipulation	Tampering	High	Open		Injection of malicious search filters to bypass constraints and return unauthorized data.	Use of proper ElasticSearch query builders, more resistant to such attempts.
13	Missing audit logs for bookings	Repudiation	Medium	Mitigated		Without proper event logging, users could deny placing a booking or changing their availability.	Proper logging and persistance and data retention.
14	Improper role enforcement	Elevation of privilege	Medium	Mitigated		Clients potentially accessing vendor-only functionality (calendar availability) and vice-versa, Vendors being able to book visits with other Vendors.	User role validation via JWT.
15	Heavy search queries	Denial of service	Medium	Open		Unbounded queries stressing ElasticSearch and slowing down its performance due to heavy processing.	Employ proper pagination within ES.

Visit Manager (Process)

Description: FastAPI app

Number	Title	Type	Priority	Status	Score	Description	Mitigations
23	Payment data manipulation	Tampering	High	Open		Modifying Stripe request data for altering payment amount, recipient etc.	
24	SQL injection	Tampering	High	Mitigated		Direct injection of malicious code.	Use of parametrized queries and ORM.
25	User Personally Identifiable Information leakage	Information disclosure	Medium	Open		Email, phone number, address exposed via poor API restrictions.	Following of GDPR guidelines in handling of data. Selective logging and masking of potentially sensitive information.
26	Stripe webhook forgery	Spoofing	High	Mitigated		Malicious requests resulting in faking of the payment events and status changes.	Validating of Stripe signatures and secrets.

User Chat (Process)

Description: FastAPI app

Number	Title	Type	Priority	Status	Score	Description	Mitigations
16	Impersonation via JWT replay	Spoofing	High	Mitigated		Interception and reusing of JWT tokens resulting in attackers sending messages on behalf of other users.	Short-lived JWT, enforce TLS encryption (HTTPS).

Number	Title	Type	Priority	Status	Score	Description	Mitigations
17	Unauthenticated message sending	Spoofing	Medium	Mitigated		Bots or anonymous users sending spam messages if endpoints lack strict authentication.	JWT authentication enforced on all endpoints.
18	Denial of message sending	Repudiation	Medium	Mitigated		User denying sending a message, difficult to prove message origin.	Proper logging and storing of message metadata, including origin verification.
21	Accidental logging of sensitive data	Information disclosure	Medium	Open		Sensitive message content may unintentionally get logged.	Log redaction, selective storing of only the necessary metadata.
22	Lack of logging for message delivery	Repudiation	Low	Mitigated		Loss of message traceability upon sending.	Implementation of proper message delivery receipts in the logging system.

App Load Balancer (Process)

Description: Nodejs App

Number	Title	Type	Priority	Status	Score	Description	Mitigations
1	JWT forwarding	Spoofing	High	Open		Attacker intercepting or replaying tokens due to improper security controls at ALB level; could result in full account takeover.	TLS encryption enforced upon the communication between the services (HTTPS). Proper token validation, short-lived JWTs.
2	Volumetric attacks	Denial of service	Medium	Mitigated		Large scale traffic floods designed to overwhelm ALB or downstream resources.	GCP Cloudflare employed.
3	Request header mangling	Tampering	Medium	Open		Attackers replacing specific request headers leading to potential bypass of security checks.	Proper header validation and sanitization.
4	Routing misconfiguration	Information disclosure	Medium	Mitigated		Misconfigured routing may expose internal endpoints to the public.	Blocking and restricting access to internal paths.
5	Lack of request attribution	Repudiation	Low	Mitigated		Without proper logging at the level of App Load Balancer it may be difficult to notice and track malicious activitiy.	Accurate monitoring setup at the outer-most layer and alerting on anomalies (Cloud Logging).
6	CORS misconfiguration	Information disclosure	Medium	Open		Without proper restriction, App Load Balancer may allow cross-origin requests, leaking data to potentially malicious sites.	Proper CORS configuration, specific headers and only trusted origins.

Firestore DB (Store)

Description:

Number	Title	Type	Priority	Status	Score	Description	Mitigations
19	Document manipulation	Tampering	High	Mitigated		If security rules are improperly configured, attackers may modify chat history records.	Strict security rules within the DB.
20	Access to unauthorized chat histories	Information disclosure	Medium	Mitigated		Missing access checks may lead to the potentially malicious users seeing conversations that they should not have access to.	Enforce Firestore access control for all documents.

Web frontend (Process)

Description: Nodejs app

Number	Title	Type	Priority	Status	Score	Description	Mitigations
7	Token leakage from localStorage	Spoofing	High	Open		Attackers getting user's locally stored credentials with the use of malicious scripts.	Use of PKCE in OAuth.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
8	Sensitive data leakage from localStorage	Information disclosure	Medium	Mitigated		Data such as chat history, or payment information, if stored locally can be leaked.	Avoid storing sensitive data client-side.
9	Volumetric attacks on backend endpoints	Denial of service	Medium	Open		Risk of repeated spam on the endpoints of the downstream microservices.	Enforcing rate limits per IP address or user.
10	Unauthorized access to dev tools	Elevation of privilege	High	Mitigated		Routes that are not blocked/restricted properly may expose access to administration/development features.	Routing safeguarding, access restriction via backend authorization.