

R³-GenAI Chat: A Self-Hosted Multimodal RAG Framework with Pre-Retrieval RBAC–ABAC Access Control and Risk-Adaptive Response Governance for Enterprise Documents

Mr. Shreyas Sridhar

Assistant Professor, Dept. of CSE
Sir M. Visvesvaraya Institute of Technology
Bengaluru, Karnataka, India

Vibhanshi Jain

Department of CSE
Sir M. Visvesvaraya Institute of Technology
Bengaluru, Karnataka, India

Anshi Saxena

Department of CSE
Sir M. Visvesvaraya Institute of Technology
Bengaluru, Karnataka, India

Bernoulli Thongam

Department of CSE
Sir M. Visvesvaraya Institute of Technology
Bengaluru, Karnataka, India

Abstract—Enterprise document assistants built on retrieval-augmented generation (RAG) are hard to deploy in regulated organisations. Hosted pipelines send confidential documents to external services, role-based access control (RBAC) ignores the network context of a request, text-only parsers lose the content of tables and figures, and retrieved context is rarely audited before it reaches the language model. This paper presents R³-GenAI Chat, a self-hosted multimodal RAG system that addresses these four problems in one pipeline. Documents are parsed with Docling, PDFPlumber and PyMuPDF; text, serialised tables and chart descriptors are indexed with BGE-M3 and ColPali/CLIP. A pre-retrieval policy filter combines RBAC with attribute-based access control (ABAC) over the request environment, anchored to Wi-Fi BSSID and subnet attestation, so unauthorised chunks never enter the nearest-neighbour search. A risk engine then scores each query ($S_{\text{risk}} \in [0, 100]$) from document sensitivity, a prompt-injection classifier, retrieved-content sensitivity, network context and behavioural anomaly, and selects one of four response actions. Generation runs locally on 4-bit Qwen-2.5-7B-Instruct or Mistral-7B-Instruct-v0.3 behind a FastAPI service, with no external API calls in the inference path. On an author-constructed benchmark of 1,250 documents (18,450 chunks) the Qwen-based configuration reached Hit Rate@5 of 98.4% (MRR@5 = 0.942) at 42 ms retrieval latency, and blocked 99.4% of data-leakage and 98.7% of lateral privilege-escalation attempts among 500 adversarial queries on a single RTX 4060 laptop GPU. The benchmark is single-site, so results should be read as evidence for the design rather than as a general benchmark.

Index Terms—Attribute-based access control (ABAC), data sovereignty, document intelligence, multimodal ingestion, on-premise LLM, retrieval-augmented generation (RAG), risk-adaptive security, zero-trust architecture.

I. INTRODUCTION

Large language models (LLMs) are increasingly used to answer questions over internal enterprise documents such as balance sheets, contracts, technical drawings and personnel records [1], [2]. Retrieval-augmented generation (RAG) reduces reliance on a model's parametric memory by conditioning generation on passages retrieved from an external index, which ties answers to source documents [1], [2].

Deploying such systems in regulated settings, for example healthcare, banking, defence and legal services, raises four practical problems.

1) **Data sovereignty.** Hosted RAG pipelines send documents or passages to third-party endpoints. This can conflict with data-protection obligations such as the GDPR [3], HIPAA

[4] and India's DPDP Act [5], and it exposes proprietary intellectual property to the provider.

- 2) **Static access control.** RBAC grants access by role alone, so it cannot distinguish a request from the corporate intranet from one made over an unverified public network [6], [7].
- 3) **Multimodal content.** Much enterprise information sits in tables, financial matrices, Gantt charts and drawings, which text-only chunkers flatten or discard [8], [9].
- 4) **Unaudited context.** Retrieved passages normally reach the model without checks for embedded instructions [10] or sensitive content, and leakage of private retrieval data through crafted queries has been demonstrated [11].

This paper presents R³-GenAI Chat, a self-hosted RAG system that handles these four problems in a single pipeline. Its contributions are:

- (1) **On-premise architecture.** Embedding, retrieval and generation run locally. Generation uses Qwen-2.5-7B-Instruct [16] or Mistral-7B-Instruct-v0.3 [17] quantised to 4 bits behind an asynchronous FastAPI service, with no external API calls in the inference path.
- (2) **Multimodal ingestion.** Layout-aware parsing (Docling [9], PDFPlumber, PyMuPDF) keeps table structure and bounding boxes. Tables are serialised as Markdown, chart descriptors come from ColPali [12] and CLIP [13] with text summaries, and everything is indexed with BGE-M3 [14] embeddings and BM25 [15] lexical matching.
- (3) **Pre-retrieval dual-layer governance.** RBAC clearance and department checks are combined with ABAC conditions over the request environment (Wi-Fi BSSID and subnet attestation, with time and device posture recorded in the request context) and applied as a metadata filter before nearest-neighbour search.
- (4) **Risk-adaptive response control.** A composite score $S_{\text{risk}} \in [0, 100]$ built from five signals is mapped to ALLOW, REDACT, CONDENSE with step-up verification, or BLOCK.
- (5) **Specification and evaluation.** A threat model, an implementation specification, and an evaluation on a 1,250-document benchmark, including an ablation over 500 adversarial queries.

RBAC–ABAC combinations and hybrid dense–sparse retrieval are established techniques. The contribution of this work is their integration at the retrieval layer of an on-premise multimodal pipeline, together with an evaluation of that integration. The remainder of the paper reviews related work (Section II), describes the method (Section III) and

implementation (Section IV), reports the evaluation (Section V), and closes with discussion, limitations and conclusions (Sections VI–VIII).

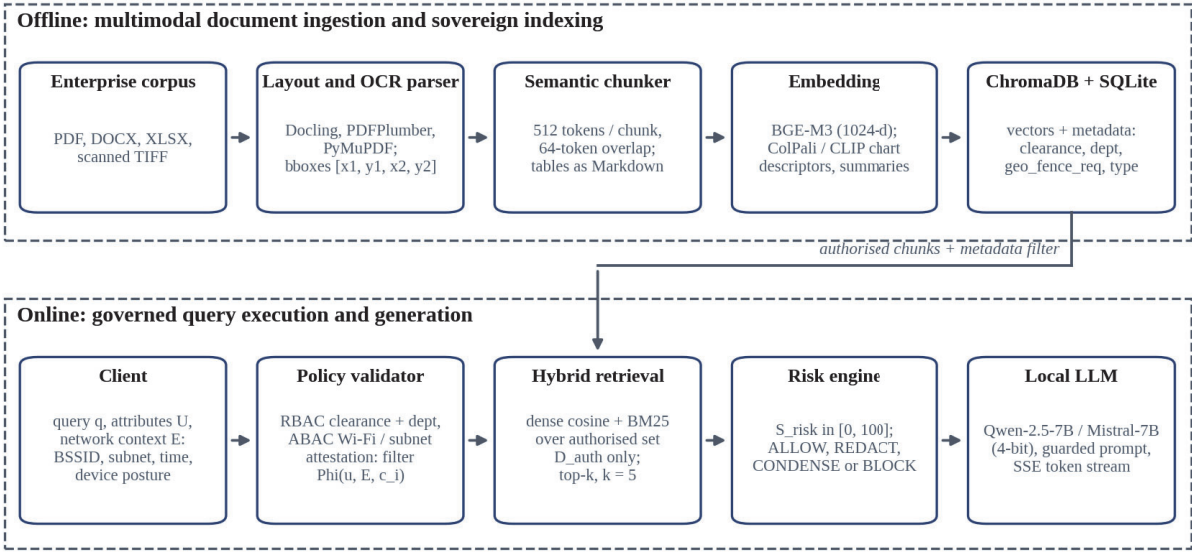


Fig. 1. End-to-end architecture of R³-GenAI Chat. Offline, documents are parsed, chunked, embedded and stored with policy metadata. Online, each query passes the policy filter, filtered hybrid retrieval, the risk engine and local generation.

II. RELATED WORK

A. RAG Paradigms and Privacy Vulnerabilities

Gao et al. [2] describe RAG as progressing from naive pipelines (chunk, embed, retrieve by cosine similarity) through advanced pipelines with query rewriting, hybrid search and re-ranking, to modular designs; Gupta et al. [18] survey the same landscape. On the security side, Zeng et al. [11] showed that RAG systems can leak the contents of their private retrieval databases, and Greshake et al. [10] showed that instructions hidden in retrieved content can hijack LLM-integrated applications. LLMs can also reproduce memorised training data [19]. Together these results imply that anything placed in the context window should be treated as potentially exposed. To our knowledge, role- and context-dependent access control enforced before nearest-neighbour search, rather than on generated output, has received less attention in the open RAG literature.

B. Multimodal Document Intelligence

Riedler and Langer [8] studied multimodal RAG in an industrial setting and reported that it can outperform single-modality RAG, although image retrieval proved harder than text retrieval and text summaries of images were a more promising route than multimodal embeddings. Docling [9] offers open-source layout analysis and table-structure recognition, and ColPali [12] retrieves page images directly with a vision-language model. R³-GenAI Chat builds on these tools: it keeps cell coordinates and captions in the chunk metadata and pairs visual descriptors with text summaries, so that tables and charts remain retrievable under the same access policy as text.

C. Dynamic Access Control and Attribute Geofencing

RBAC maps permissions to roles. ABAC evaluates policy rules over subject, object and environment attributes [6], and combining the two models is an established practice [7]. We do not claim the combination as new. What we evaluate is its placement: the combined policy, including network attributes such as the Wi-Fi BSSID and subnet, is evaluated as a metadata

filter on the vector index, so that chunks a user may not see are never candidates for retrieval.

The study addresses the research questions in Table I, and Table II positions the system against representative prior work.

TABLE I RESEARCH QUESTIONS AND MOTIVATIONS		
RQ	Research question	Motivation
RQ1	Can an on-premise open-source LLM approach the quality of a cloud-API baseline while removing data-sovereignty risk?	Avoid third-party exposure of documents and recurring API cost.
RQ2	Can tables and charts be retrieved without losing spatial and numeric context?	Reduce wrong figures caused by flattened tables and charts.
RQ3	Does combining RBAC with ABAC and Wi-Fi geofencing reduce lateral privilege escalation?	Keep sensitive content restricted to verified network contexts.
RQ4	Can a continuous multi-factor risk score detect prompt injection and permission creep at query time?	Replace binary filters with graded responses before context reaches the LLM.
RQ5	What latency does local parsing, filtering, scoring and generation add on commodity hardware?	Keep interaction responsive (retrieval under 150 ms, generation under 1.2 s).

TABLE II
 FEATURE COMPARISON WITH REPRESENTATIVE PRIOR WORK

System	Hosting	Modality	Access control	Network context	Risk score
Naive / advanced RAG pipelines [2]	Depends on deployment	Mainly text	Not part of the pipeline	—	—
PrivateGPT [20]	On-premise	Text	Not part of the core pipeline	—	—
Riedler and Langer [8]	Hosted (GPT-4V) and local (LLaVA)	Text + images	Not addressed	—	—
R³-GenAI Chat (this work)	On-premise	Text, tables, images	RBAC + ABAC	BSSID, subnet, time	Continuous, 0–100

Note: entries describe the cited sources; “—” means the feature is not addressed there. Table II compares features only; it makes no performance comparison.

III. METHODOLOGY

R³-GenAI Chat has four layers: (1) multimodal ingestion and structural normalisation, (2) dual-layer governance and geofenced retrieval, (3) continuous risk-adaptive scoring, and (4) local LLM orchestration (Fig. 1). Fig. 2 shows the order of operations for a single query, and Table III defines the notation used below.

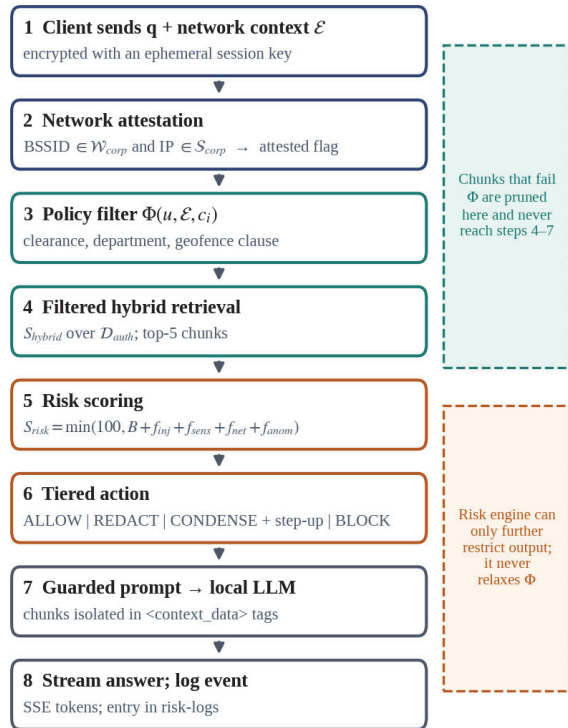


Fig. 2. Processing order for one query. Chunks removed by the policy filter Φ never reach retrieval, scoring or the prompt; the risk engine can only further restrict output.

TABLE III
 NOTATION

Symbol	Meaning
c_i, e_i	Chunk i and its L2-normalised BGE-M3 embedding (1024-d)
M_i	Metadata payload {doc_id, dept, clearance_level, geo_fence_req, content_type}
$u, \mathcal{U}$	User and attribute set (role _u , dept _u , clearance _u)
$\mathcal{E}$	Request environment (BSSID, SubnetIP, DevicePosture, Timestamp)
$\mathcal{W}_{corp}, \mathcal{S}_{corp}$	Corporate BSSID whitelist; internal subnet CIDR block
Φ, Ψ_{geo}	Pre-retrieval policy filter; geofence predicate
$\mathcal{D}_{auth}$	Candidate chunks with $\Phi = 1$
α, k	Dense/lexical fusion weight (0.70); retrieval depth (5)
C_{top}, H_u	Top- k retrieved chunks; behavioural baseline of user u
S_{risk}	Composite risk score in [0, 100]

A. Threat Model and Assumptions

Assets. Documents and chunks classified Public, Internal, Confidential or Restricted, the user attributes $\mathcal{U}$, and session state. **Adversary behaviours.** Five behaviours are considered, matching the attack categories of Section V-D: (T1) an authenticated user querying beyond their clearance or department (lateral privilege escalation); (T2) a valid user on an untrusted network such as public Wi-Fi; (T3) direct prompt injection by the user and indirect injection through instructions hidden in ingested documents; (T4) extraction of personal or financial data outside the context of the user's task; and (T5) a client that falsely claims an on-premise network origin (geofence spoofing).

Assumptions. User identity and the attributes in $\mathcal{U}$ come from an authenticated session; the server host, operating system and GPU are trusted; administrators maintain $\mathcal{W}_{corp}$ and $\mathcal{S}_{corp}$. **Out of scope.** Server compromise, malicious administrators, hardware side channels, physical attacks on access points, model supply-chain attacks, and adaptive attackers who know the risk thresholds.

B. Multimodal Ingestion Pipeline

Ingestion converts PDF, DOCX, XLSX and scanned TIFF files into metadata-tagged chunks (Algorithm 1) in three parsing stages. **Structural layout analysis:** Docling [9], PDFPlumber and PyMuPDF extract hierarchical text blocks, headers, footnotes and table bounding boxes $[x_1, y_1, x_2, y_2]$. **Tabular serialisation:** multi-row tables such as balance sheets become Markdown matrices in which every cell is bound to its column header, so that a row is never separated from its header by chunking. **Visual content:** embedded figures and chart images are described by a local vision-language backbone (ColPali [12] / CLIP [13]) and a synthetic text summary. **[AUTHOR INPUT: how ColPali/CLIP descriptors are stored and retrieved next to the 1024-d BGE-M3 index]**

Content is split into chunks c_i with a sliding window (chunk_size = 512 tokens, overlap = 64). Each chunk is embedded with the open-source BAAI/bge-m3 model [14] and L2-normalised:

$$e_i = \text{Normalize}(\text{BGE-M3}(c_i)) \in \mathbb{R}^{1024} \quad (1)$$

Vectors are stored in a local ChromaDB [22] collection with the JSON payload M_i . clearance_level is one of four ordered values (Public < Internal < Confidential < Restricted) and geo_fence_req is Boolean; clearance_level and dept supply req_clearance(c_i) and allowed_depts(c_i) in Eq. (2). Table IV lists the configuration.

Algorithm 1. Ingest(file f , dept d , tier t , geo g)
 1 blocks, tables, figures $\leftarrow$ Parse(f) // Docling, PDFPlumber, PyMuPDF
 2 for each table T : text(T) $\leftarrow$ Markdown(T) with header binding
 3 for each figure G : desc(G) $\leftarrow$ Describe(G) // ColPali/CLIP + summary
 4 chunks $\leftarrow$ Chunk(blocks $\cup$ text $\cup$ desc, 512, 64)
 5 for each chunk c_i : $e_i \leftarrow$ Normalize(BGE-M3(c_i))
 6 $M_i \leftarrow (f.id, d, t, g, type(c_i))$
 7 store (c_i, e_i, M_i) in ChromaDB; add c_i to BM25 index

TABLE IV
 CONFIGURATION OF THE EVALUATED SYSTEM

Parameter	Value
Input formats	PDF, DOCX, XLSX, scanned TIFF
Parsing	Docling, PDFPlumber, PyMuPDF; table bounding boxes retained
Chunking	512 tokens, 64-token overlap; tables as Markdown
Visual content	ColPali / CLIP descriptors plus text summary
Dense embedding	BAAI/bge-m3, 1024-d, L2-normalised
Lexical retrieval	BM25, normalised score
Fusion / depth	$\alpha = 0.70$; $k = 5$
Stores	ChromaDB (vectors); SQLite (metadata)
Generators	Qwen-2.5-7B-Instruct, Mistral-7B-Instruct-v0.3; 4-bit AWQ / GGUF
Serving	vLLM / Ollama; singleton model manager
Hardware	Intel Core i5-13500H, 16 GB DDR5, RTX 4060 Laptop GPU (8 GB VRAM)
Software	Ubuntu 22.04 LTS, CUDA 12.2, Python 3.11

C. Dual-Layer Access Governance (RBAC + ABAC)

Let user u have attributes $\mathcal{U} = \langle \text{role}_u, \text{dept}_u, \text{clearance}_u \rangle$ and issue query q from environment $\mathcal{E} = \langle \text{BSSID}, \text{SubnetIP}, \text{DevicePosture}, \text{Timestamp} \rangle$. The policy validator builds the Boolean filter

$$\Phi(u, \mathcal{E}, c_i) = 1[\text{clearance}_u \geq \text{req_clearance}(c_i)] \wedge 1[\text{dept}_u \in \text{allowed_depts}(c_i)] \wedge \Psi_{\text{geo}}(\mathcal{E}, c_i) \quad (2)$$

where the geofence predicate is

$$\Psi_{\text{geo}}(\mathcal{E}, c_i) = 1 \text{ if } \text{geo_req}(c_i) = \text{False}; \\ 1[\text{BSSID} \in \mathcal{W}_{\text{corp}} \wedge \text{IP} \in \mathcal{S}_{\text{corp}}] \text{ if } \text{geo_req}(c_i) = \text{True} \quad (3)$$

Here $\mathcal{W}_{\text{corp}}$ is the authorised corporate Wi-Fi BSSID whitelist and $\mathcal{S}_{\text{corp}}$ the internal subnet CIDR block (for example 192.168.10.0/24). Chunks with $\Phi = 0$ are removed before nearest-neighbour search, so they cannot be retrieved, ranked or passed to the model. Two properties follow from the definitions. First, Ψ_{geo} depends on the request rather than the chunk, so for a given request it reduces to a single clause: if the network is not attested, only chunks with $\text{geo_fence_req} = \text{False}$ remain candidates. Second, the filter is conjunctive, so failing any one of clearance, department or geofence removes the chunk, and nothing downstream can restore it. In the evaluated configuration Eq. (3) uses BSSID and subnet; the time of day enters through f_{net} (Eq. 5), and DevicePosture is recorded in $\mathcal{E}$. **[AUTHOR INPUT: state whether DevicePosture is used in Φ or in the risk score]**

D. Hybrid Retrieval over the Authorised Set

Retrieval runs only over $\mathcal{D}_{\text{auth}} = \{c_i \in \mathcal{D} \mid \Phi(u, \mathcal{E}, c_i) = 1\}$. The relevance score combines dense cosine similarity with a normalised BM25 score [15]:

$$S_{\text{hybrid}}(q, c_i) = \alpha \cdot (e_q \cdot e_i) / (\|e_q\| \|e_i\|) + (1 - \alpha) \cdot \text{BM25}_{\text{norm}}(q, c_i) \quad (4)$$

with $\alpha = 0.70$, which weights semantic generalisation against exact matching of numeric identifiers and legal clauses. The top $k = 5$ chunks form C_{top} and are passed to the risk engine.

E. Continuous Risk-Adaptive Security Engine

Before the prompt is built, the engine computes a composite score for the transaction (Fig. 3, Table V):

$$S_{\text{risk}} = \min(100, B_{\text{base}} + f_{\text{inj}}(q) + f_{\text{sens}}(C_{\text{top}}) + f_{\text{net}}(\mathcal{E}) + f_{\text{anom}}(H_u)) \quad (5)$$

$$f_{\text{inj}}(q) = 40 \cdot P(\text{PromptInjection} \mid q) \quad (6)$$

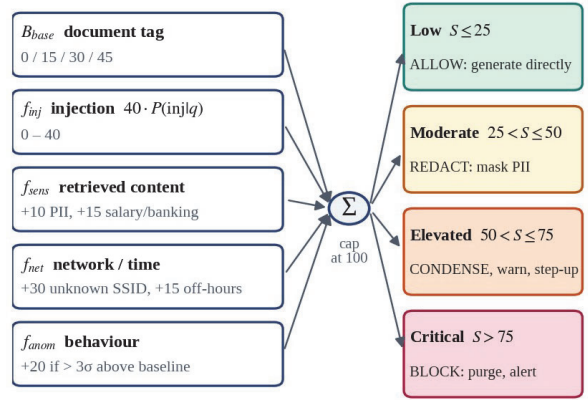


Fig. 3. Risk scoring. Five additive terms are summed, capped at 100, and mapped to one of four response tiers.

TABLE V
 TERMS OF THE RISK SCORE

Term	Signal	Points
B_{base}	Document classification tag	Public 0, Internal 15, Confidential 30, Restricted 45
f_{inj}	Prompt-injection probability P	$40 \cdot P$, range 0–40
f_{sens}	PII / financial density of retrieved chunks	+10 for PII; +15 for salary or banking records
f_{net}	Network and time context	+30 unknown network SSID; +15 access during off-hours (22:00–06:00)
f_{anom}	Query-frequency anomaly	+20 if frequency exceeds 3 standard deviations of the user baseline H_u

f_{inj} is 40 times the probability returned by a lightweight fine-tuned transformer classifier that targets prompt leaks, jailbreaks and role overrides. **[AUTHOR INPUT: base model, fine-tuning data and validation results of the classifier]** The point values in Table V are fixed design parameters of the evaluated configuration. Because the terms are additive and capped by $\min(100, \cdot)$, one strong signal, for example a Restricted document requested from an unknown network, can move a query into a higher tier even if the other terms are zero.

The score selects one of four actions: **Low** ($S_{\text{risk}} \leq 25$), direct generation; **Moderate** ($25 < S_{\text{risk}} \leq 50$), automated PII masking of context chunks before prompt construction; **Elevated** ($50 < S_{\text{risk}} \leq 75$), a condensed response with an appended warning and a step-up verification prompt; and **Critical** ($S_{\text{risk}} > 75$), query termination, context purging and a logged security alert. The original integer ranges (26–50, 51–75, ≥ 76) are equivalent for integer scores; half-open intervals make the mapping total for the real-valued f_{inj} . The risk engine only restricts output and

cannot re-admit a chunk removed by Φ . Fig. 4 illustrates how network trust and threat level combine into actions.

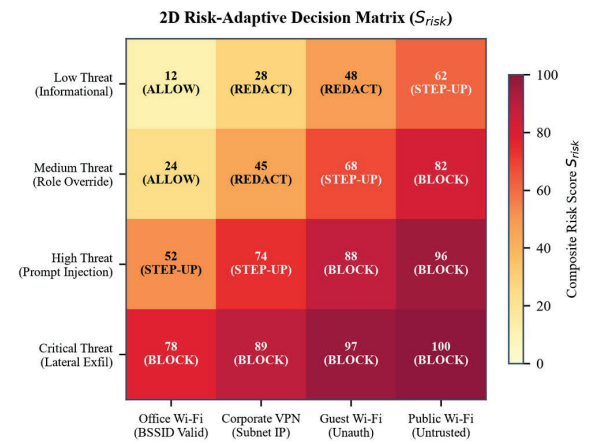


Fig. 4. Illustrative decision matrix: example scores and resulting actions for four threat levels (rows) and four network contexts (columns).

F. Local LLM Orchestration

To avoid external dependencies, inference uses 4-bit quantised (AWQ [21] / GGUF) Qwen-2.5-7B-Instruct and Mistral-7B-Instruct-v0.3. Prompts follow a defensive “sandwich”: system instructions, then the retrieved context isolated between `<context_data>` and `</context_data>` delimiters, then a restatement of the task and the user question. The delimiters mark retrieved text as data so that it is not read as instructions, which limits delimiter hijacking.

G. Query-Time Procedure

```
Algorithm 2. Answer(query q, user u, environment  $\mathcal{E}$ )
1 attested  $\leftarrow$  Attest( $\mathcal{E}$ ) // BSSID  $\in \mathcal{N}_{corp} \wedge IP \in \mathcal{N}_{corp}$ 
2  $\mathcal{Q}_{auth} \leftarrow \{c_i : \Phi(u, \mathcal{E}, c_i) = 1\}$  // before similarity search
3  $C_{top} \leftarrow$  top-5 of  $\mathcal{Q}_{auth}$  by  $S_{hybrid}(q, \cdot)$ 
4  $S \leftarrow \min(100, B_{base} + f_{inj} + f_{sens} + f_{net} + f_{anom})$ 
5 if  $S \leq 25$ : ctx  $\leftarrow C_{top}$ 
6 elif  $S \leq 50$ : ctx  $\leftarrow$  MaskPII( $C_{top}$ )
7 elif  $S \leq 75$ : ctx  $\leftarrow C_{top}$ ; condense, warn, request step-up
8 else: purge context; log alert; refuse
9 prompt  $\leftarrow$  Sandwich(system, <context_data>ctx</context_data>, q)
10 stream LLM(prompt) to client; log event
```

IV. SYSTEM IMPLEMENTATION

A. Asynchronous REST Backend

The backend is written in Python 3.11 with FastAPI [23] and keeps services decoupled. A singleton model manager loads the quantised weights into GPU VRAM at start-up and serves them with vLLM [24] or Ollama; the prompt-injection classifier runs in PyTorch [25]. The core endpoints are listed in Table VI.

TABLE VI CORE API ENDPOINTS		
Method	Endpoint	Function
POST	/api/v1/ingest-document	Accepts multipart PDF/image streams, runs Docling layout parsing, computes BGE-M3 embeddings and writes vectors and metadata to ChromaDB.
POST	/api/v1/chat/query	Receives the prompt and client network telemetry (BSSID, subnet, client MAC hash); prunes by ABAC before retrieval; computes S_{risk} ; builds the guarded prompt; streams tokens over SSE.
POST	/api/v1/security/attest-network	Validates client network signatures against corporate access-point beacons.
GET	/api/v1/telemetry/risk-logs	Returns threat telemetry for administrative dashboards.

B. Frontend and Wi-Fi Attestation

The client is a single-page application (React 18, Vite, TailwindCSS) with a lightweight Streamlit interface for administrative inspection. At session start a background worker gathers local network connection identifiers through WebRTC network discovery and local gateway telemetry and encrypts the payload with an ephemeral session key. The server validates the reported signatures against corporate access-point beacons using a Scapy-based BSSID validator. [AUTHOR INPUT: how the client obtains the BSSID, since standard browser APIs do not expose it] The aim is to make it difficult for an off-premise client to pass as an internal one. This is a risk-reduction measure, not a proof against a determined local attacker (Section VII).

C. Deployment

Fig. 5 shows the deployment. All components run inside the organisation’s network, and external cloud LLM or embedding services are not used in the inference path. Section V-F reports functional tests of each subsystem.

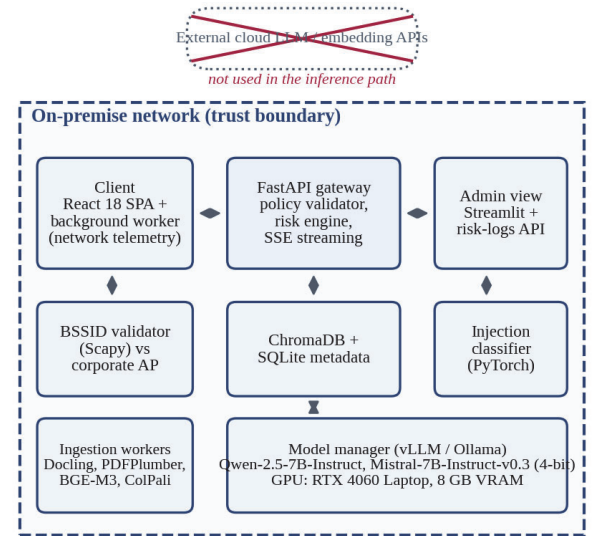


Fig. 5. Deployment view. All components sit inside the organisation’s network boundary; no external LLM or embedding API is called.

V. EXPERIMENTAL EVALUATION

A. Setup and Benchmark Corpus

The system was deployed on one workstation with an Intel Core i5-13500H CPU, 16 GB DDR5 RAM and an NVIDIA GeForce RTX 4060 Laptop GPU (8 GB VRAM) running Ubuntu 22.04 LTS and CUDA 12.2 (Table IV). The benchmark contains 1,250 multi-page enterprise documents from five divisions (Finance, Human Resources, Engineering, Legal and Corporate General), split into 18,450 chunks with ground-truth role tags, clearance tiers and tabular matrices (Table VII). **[AUTHOR INPUT: dataset provenance (real, anonymised or synthetic), number of evaluation queries, how relevant chunks were labelled and by whom, and the exact definition and per-category count of the 500 adversarial attacks]**

TABLE VII
DATASET CLASSIFICATION AND SECURITY TAXONOMY

Division	Document types	Modality mix	Security tier	Geofence policy
Finance	Quarterly P&L, audit ledgers, tax filings	80% tables, 15% text, 5% charts	Confidential / Restricted	Office Wi-Fi mandatory
Human Resources	Compensation plans, PII records, grievances	90% text, 10% tables	Restricted	Office Wi-Fi mandatory
Engineering	System architecture, API specs, schematics	50% text, 35% schematics, 15% code	Internal	Hybrid (VPN / on-premise)
Legal	NDAs, vendor contracts, compliance filings	95% text, 5% signatures / scans	Confidential	Hybrid (VPN / on-premise)
Corporate General	Handbooks, public PR, policies	85% text, 15% visuals	Public / Internal	Unrestricted

Five pipelines were compared: a BM25 keyword baseline; a dense RAG baseline (BGE-Small retrieval, Mistral-7B generation); a cloud-API RAG baseline with GPT-4o generation; and R³-GenAI Chat with Mistral-7B-v0.3 or Qwen-2.5-7B. **[AUTHOR INPUT: retriever, prompt and model version used in the cloud-API baseline and the dense baseline]**

B. Metrics

Retrieval is scored by Hit Rate@ k (HR@ k) and mean reciprocal rank (MRR@ k) with $k = 5$, where rank_q is the rank of the first relevant chunk for query q in the query set $\mathcal{Q}$:

$$\text{HR}@k = (1/|\mathcal{Q}|) \sum_q 1[\text{rank}_q \leq k] \quad (7)$$

$$\text{MRR}@k = (1/|\mathcal{Q}|) \sum_q (1/\text{rank}_q) \cdot 1[\text{rank}_q \leq k] \quad (8)$$

Context faithfulness is the percentage of answers supported by the retrieved context, and the hallucination rate is the percentage that are not. **[AUTHOR INPUT: how faithfulness and hallucination were judged (judge model or manual annotation) and the latency protocol (warm-up, number of runs, mean or median)]** Security results are reported as defence or block rates over the 500 adversarial queries.

C. Retrieval and Generation Comparison

Table VIII reports the comparison. With Qwen-2.5-7B, R³-GenAI Chat reached $\text{MRR}@5 = 0.942$, $\text{HR}@5 = 98.4\%$, context faithfulness of 98.1% (hallucination rate 1.9%), retrieval latency of 42 ms and generation latency of 480 ms. With Mistral-7B-v0.3 the corresponding figures were 0.938, 97.8%, 96.9% (3.1%), 48 ms and 590 ms. The dense baseline

reached $\text{HR}@5 = 82.1\%$ and the cloud-API baseline 91.5%. R³-GenAI Chat differs from these baselines in parsing, retrieval fusion, governance and generator at the same time, so Table VIII does not isolate which component produces the gain. Fig. 6 shows how latency grows with context length: over 512 to 8,192 tokens the local models stay below roughly 0.6 s, whereas the cloud-API baseline reaches roughly 2.1 s at 8,192 tokens (values read from the figure).

TABLE VIII
RETRIEVAL AND GENERATION COMPARISON

Pipeline	MRR@5	HR@5 (%)	Faith. (%)	Halluc. (%)	Retrieval	Generation
BM25 keyword baseline	0.612	68.4	71.2	28.8	12 ms	—
Dense RAG (BGE-Small, Mistral-7B)	0.784	82.1	84.5	15.5	28 ms	1,420 ms
Cloud-API RAG baseline (GPT-4o)	0.892	91.5	92.8	7.2	340 ms	1,850 ms
R ³ -GenAI (Mistral-7B-v0.3)	0.938	97.8	96.9	3.1	48 ms	590 ms
R³-GenAI (Qwen-2.5-7B)	0.942	98.4	98.1	1.9	42 ms	480 ms

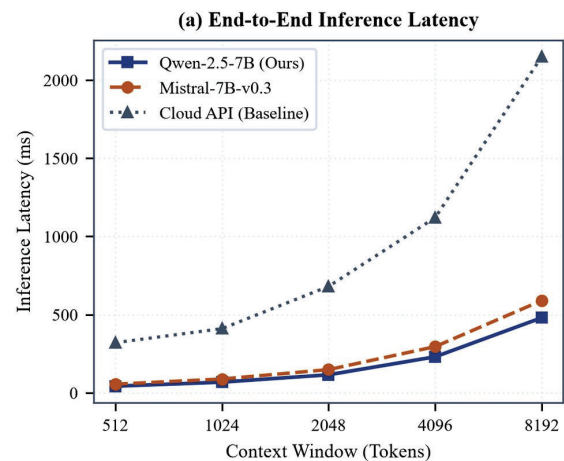


Fig. 6. End-to-end inference latency against context window for the two local models and the cloud-API baseline.

D. Security Ablation and Adversarial Testing

To measure the contribution of each layer, 500 adversarial attack vectors were run against configurations that add one protection at a time. The attacks fall into four categories: (1) lateral privilege-escalation queries, (2) direct and indirect prompt injections, (3) off-premise geofence spoofing and (4) out-of-context PII extraction (Table IX, Fig. 7).

TABLE IX
ABLATION OF THE SECURITY PIPELINE

Configuration	Lateral esc. defence	Prompt inj. defence	Geofence bypass	Leakage block	Latency overhead
Vanilla RAG (no governance)	24.0%	18.2%	100.0%	38.2%	0 ms
+ Static RBAC filter	58.2%	41.5%	100.0%	64.5%	+4 ms
+ ABAC and Wi-Fi geofencing	82.6%	72.0%	0.6%	87.1%	+11 ms
+ Real-time risk engine (Eq. 5)	92.4%	94.8%	0.6%	95.8%	+19 ms
Full stack (all modules)	98.7%	99.2%	0.0%	99.4%	+23 ms

Static RBAC raised lateral-escalation defence from 24.0% to 58.2%. Adding ABAC with Wi-Fi geofencing raised it to 82.6% and cut the geofence bypass rate from 100% to 0.6%. Adding the risk engine raised prompt-injection defence from 72.0% to 94.8%, the largest step for that category. With all modules enabled, the system reached 98.7% lateral-escalation defence, 99.2% prompt-injection defence, 0.0% geofence bypass and 99.4% leakage blocking, at +23 ms mean latency overhead relative to the ungoverned pipeline.

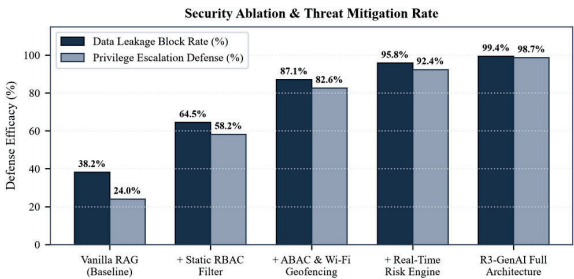


Fig. 7. Progressive threat mitigation (data-leakage block rate and privilege-escalation defence) as protection layers are added; values as in Table IX.

E. Multimodal Document Parsing

To test non-textual comprehension, table and chart question answering was evaluated on 1,250 test chunks across five document formats (Table X, Fig. 8). Structural ingestion improved accuracy over vanilla text extraction for every format, most strongly for engineering schematics (22.1% to 92.1%), financial balance sheets (48.6% to 95.4%) and complex tabular audits (44.0% to 96.2%). Numeric cell accuracy ranged from 93.0% to 99.2% and was lowest for scanned OCR receipts (93.0%) and engineering schematics (94.5%), the formats that depend most on OCR and drawing interpretation.

TABLE X
MULTIMODAL DOCUMENT VALIDATION BY CATEGORY

Document format	Test chunks	Vanilla extraction acc.	R ³ structural ingestion acc.	Numeric cell acc. (%)
Plain-text policies	450	94.2%	98.6%	99.2%
Financial balance sheets (PDF)	320	48.6%	95.4%	97.8%
Engineering schematics / CAD	180	22.1%	92.1%	94.5%
Scanned OCR receipts	150	52.4%	91.8%	93.0%
Complex tabular audits	150	44.0%	96.2%	98.4%

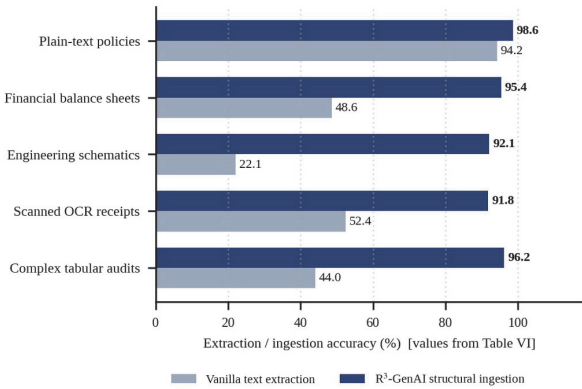


Fig. 8. Extraction accuracy of vanilla text extraction and R³-GenAI structural ingestion by document format (values from Table X).

F. Subsystem Verification

Table XI records functional test outcomes for each subsystem. These are pass counts for the stated test scope, not accuracy measurements, and they complement the metrics above.

TABLE XI
SUBSYSTEM FUNCTIONAL VERIFICATION

Subsystem	Framework	Scope	Passed
Multimodal ingestion	Docling / PDFPlumber / BGE-M3	Parsing and embedding of 1,250 multi-format documents	1,250 / 1,250
Vector store and RBAC	ChromaDB / SQLite metadata	Metadata pre-filtering and cosine similarity	500 / 500
Wi-Fi geofence engine	FastAPI / Scapy BSSID validator	Subnet containment and BSSID authentication	250 / 250
Risk scoring engine	PyTorch / Eq. (5)	Multi-factor threat detection and redaction	500 / 500
Local LLM inference	vLLM / Ollama (Qwen-2.5-7B)	On-premise streaming generation	500 / 500

VI. DISCUSSION

The results support enforcing access policy at the retrieval stage. Output guardrails inspect text after the model has produced it and can miss subtle semantic leaks. In R³-GenAI Chat, chunks that fail Φ are never placed in the prompt, so leakage of those chunks through generation is excluded by construction, provided that the policy attributes and the network attestation are correct. The ablation in Table IX suggests that the layers contribute differently: ABAC with geofencing mainly reduced lateral escalation and geofence bypass, and the risk engine mainly reduced prompt-injection success.

A worked example shows how the risk engine acts. An authorised employee asks about executive compensation from an unverified guest network. The Restricted tag gives $B_{\text{base}} = 45$ and the unknown SSID gives $f_{\text{net}} = 30$, so the partial sum is 75; the remaining terms add 3.4 points and $S_{\text{risk}} = 78.4$. This exceeds 75, so the query is terminated, context is purged and the user is asked to move to the secure office network. Under Table VII, Finance and HR documents carry a mandatory geofence, so such a query would normally be stopped earlier by Φ ; the example applies to documents whose `geo_fence_req` is False.

The numbers come from a single-site, author-constructed benchmark run on one laptop-class GPU. They show that the design behaves as intended on this corpus, but they do not establish performance on other corpora, languages, hardware or attackers.

VII. LIMITATIONS, ETHICS AND FUTURE WORK

Evaluation scope. The benchmark is single-site and author-constructed. Results are point estimates without confidence intervals or significance tests, the baselines differ from R³-GenAI Chat in several components at once (Table VIII), and the attack count per category is modest. Retrieval-side ablations (parsing, fusion weight α , filtering) were not run, and the risk point values and tier thresholds were not tuned or tested for sensitivity.

Security assumptions. BSSID and IP values can be spoofed by a determined local adversary, so Wi-Fi attestation should be read as a contextual risk signal used together with user authentication, not as a stand-alone guarantee. The injection classifier can produce false negatives, especially against adaptive attacks, and authorised insiders on the corporate network are constrained only by RBAC and the risk score.

Privacy and ethics. Network identifiers, device posture and behavioural baselines are close to personal data. Deployments should apply data minimisation, purpose limitation and retention limits to telemetry and risk logs, consistent with laws such as the DPDP Act [5], and should tell users why a query was restricted. The anomaly baseline can mis-flag unusual but legitimate work, so tier outcomes should be open to human review. The work involved no human subjects.

Future work. (1) OCR overhead: degraded handwritten archives incur a 1.4 s/page OCR latency penalty, and quantised TrOCR backbones are planned. (2) Federated multi-branch operation needs privacy-preserving vector sharding. (3) Formal verification of ABAC rule graphs could detect conflicting permissions. (4) Releasing the benchmark and code, running retrieval-side ablations, and testing against adaptive attackers would strengthen the evidence.

VIII. CONCLUSION

This paper presented R³-GenAI Chat, a self-hosted multimodal RAG system with pre-retrieval RBAC-ABAC governance and risk-adaptive response control. By combining local open-source LLMs, layout-aware parsing, Wi-Fi-attested ABAC filtering and multi-factor risk scoring, it keeps the inference path on-premise. On the authors' 1,250-document benchmark the Qwen-based configuration reached a Hit Rate@5 of 98.4% at 42 ms retrieval latency, and the full security stack blocked 99.4% of data-leakage and 98.7% of lateral-escalation attempts among 500 adversarial queries. These results come from a single-site evaluation; independent benchmarks, retrieval ablations and adaptive-attacker tests are the next steps.

ACKNOWLEDGMENT

The authors thank Mr. Shreyas Sridhar, Assistant Professor, Department of Computer Science & Engineering, Sir M. Visvesvaraya Institute of Technology, for project guidance, supervision and technical suggestions throughout this work. They also thank Ms. Shivaranjini C, Assistant Professor, Dept. of CSE, for foundational guidance, and Sir M. Visvesvaraya Institute of Technology, Bengaluru, for institutional

encouragement, research facility support and academic guidance.

REFERENCES

- [1] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, 2020, pp. 9459–9474.
- [2] Y. Gao et al., "Retrieval-augmented generation for large language models: A survey," *arXiv:2312.10997*, 2023.
- [3] European Parliament and Council, "Regulation (EU) 2016/679 (General Data Protection Regulation)," *Off. J. Eur. Union*, L 119, 2016.
- [4] U.S. Congress, "Health Insurance Portability and Accountability Act of 1996," Pub. L. No. 104-191, 1996.
- [5] Government of India, "The Digital Personal Data Protection Act, 2023," Act No. 22 of 2023, Ministry of Electronics and Information Technology (MeitY).
- [6] V. C. Hu et al., "Guide to attribute based access control (ABAC) definition and considerations," NIST Special Publication 800-162, 2014.
- [7] E. Coyne and T. R. Weil, "ABAC and RBAC: Scalable, flexible, and auditable access management," *IT Professional*, vol. 15, no. 3, pp. 62–65, 2013.
- [8] M. Riedler and S. Langer, "Beyond text: Optimizing RAG with multimodal inputs for industrial applications," *arXiv:2410.21943*, 2024.
- [9] C. Auer et al., "Docling technical report," *arXiv:2408.09869*, 2024.
- [10] K. Greshake et al., "Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection," in *Proc. 16th ACM Workshop on Artificial Intelligence and Security (AISec)*, 2023, pp. 79–90.
- [11] S. Zeng et al., "The good and the bad: Exploring privacy issues in retrieval-augmented generation (RAG)," in *Findings of the Association for Computational Linguistics: ACL 2024*, Bangkok, Thailand, 2024.
- [12] M. Faysse et al., "ColPali: Efficient document retrieval with vision language models," *arXiv:2407.01449*, 2024.
- [13] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Machine Learning (ICML)*, 2021.
- [14] J. Chen et al., "BGE M3-Embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation," *arXiv:2402.03216*, 2024.
- [15] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: BM25 and beyond," *Found. Trends Inf. Retr.*, vol. 3, no. 4, pp. 333–389, 2009.
- [16] Qwen Team, "Qwen2.5 technical report," *arXiv:2412.15115*, 2024.
- [17] A. Q. Jiang et al., "Mistral 7B," *arXiv:2310.06825*, 2023.
- [18] S. Gupta et al., "A comprehensive survey of retrieval-augmented generation (RAG): Evolution, current landscape and future directions," *arXiv:2410.12837*, 2024.
- [19] N. Carlini et al., "Extracting training data from large language models," in *Proc. 30th USENIX Security Symp.*, 2021, pp. 2633–2650.
- [20] Zylon AI, "PrivateGPT." [Online]. Available: <https://github.com/zylon-ai/private-gpt>
- [21] J. Lin et al., "AWQ: Activation-aware weight quantization for on-device LLM compression and acceleration," in *Proc. Machine Learning and Systems (MLSys)*, 2024.
- [22] Chroma, "Chroma: the open-source embedding database." [Online]. Available: <https://www.trychroma.com/>
- [23] S. Ramírez, "FastAPI." [Online]. Available: <https://fastapi.tiangolo.com/>
- [24] W. Kwon et al., "Efficient memory management for large language model serving with PagedAttention," in *Proc. 29th ACM Symp. Operating Systems Principles (SOSP)*, 2023.
- [25] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 32, 2019, pp. 8024–8035.