AKHUWAT COLLEGE KASUR
AFFILIATED WITH
UNIVERSITY OF THE PUNJAB, LAHORE.

FINAL YEAR PROJECT (FYP)

DELIVERABLE-02: SOFTWARE DESIGN SPECIFICATION

PROJECT TITLE

AI-DRIVEN SECURITY MONITORING: ANOMALY DETECTION IN ELK AND WAZUH

BS (IT)

SESSION: 2021-2025

GROUP MEMBERS

Khalid Hussain	058960
Muhammad Osama	058949

SUPERVISED BY:

EBRYX TEAM

&

MR. MUHAMMAD NAEEM AKHTAR

Table of Contents

System Design 3

1. Product Perspective 3

2. Design Considerations 3

3. Requirements Traceability Matrix 4

4. Design Models 5

4.1 Architectural Design 5

4.2 Data Design 6

4.3 User Interface Design 7

4.4 Behavioral Model 8

5. Design Decisions 8

6. Summary 9

7. References 9

System Design

1. Product Perspective

The system is a security monitoring tool that integrates Wazuh (a Security Information and Event Management platform) with the Elastic Stack (Elasticsearch, Logstash, Kibana) to collect, analyze, and visualize security logs. It uses AI-driven anomaly detection to identify threats like unauthorized file creation and suspicious login attempts, providing real-time alerts and dashboards for security teams.

2. Design Considerations

- **Assumptions**:
  - Wazuh, Filebeat, and Elastic Stack are correctly installed and configured.
  - Sufficient log data is available for AI/ML analysis.
  - Users have basic familiarity with Kibana dashboards.
- **Dependencies**:
  - Wazuh for log collection and initial processing.
  - Elastic Stack for storage, processing, and visualization.
  - Python libraries (e.g., scikit-learn) for AI/ML models.
- **Limitations**:
  - Alerts are sent but no automatic threat mitigation (e.g., auto-blocking IPs).
  - Web-based Kibana dashboards only, no mobile app support.
- **Risks and Mitigation**:
  - **Integration Challenges**: Issues connecting to external log sources.
    - **Mitigation**: Test integrations with common log formats (e.g., Syslog, JSON) during development and provide setup guides.
  - **Performance Issues**: Slowdowns with large log volumes (>15 GB monthly).
    - **Mitigation**: Optimize Elasticsearch indexing and scale with additional nodes for larger deployments.
  - **Data Quality**: Poor log data reducing AI accuracy.
    - **Mitigation**: Preprocess logs with filebeat to ensure consistency and validate AI models with diverse datasets.

3. Requirements Traceability Matrix

The matrix links requirements from the SRS to design components and test cases, ensuring all functional and non-functional requirements are addressed.

Requirement ID	Description	Design Component	Test Case	Implementation Status
FR-1	Log Collection	Wazuh Agents, Filebeat	Verify logs collected from 10 endpoints	Completed
FR-2	AI-Based Anomaly Detection	AI/ML Model (Python)	Test anomaly detection accuracy (>90%)	In Progress
FR-3	Alert System	Wazuh Manager, Kibana	Confirm alerts sent within 1 second	Completed
FR-4	Dashboard Visualization	Kibana Dashboards	Validate dashboard updates in real-time	Completed
FR-5	Anomalous File Creation Detection	Wazuh FIM, AI Model	Test alerts for unauthorized file creation	In Progress
FR-6	Suspicious Login Attempt Detection	Wazuh, AI Model	Test alerts for too many failed logins in a minute	In Progress
FR-7	Alert Customization	Kibana Alert Rules	Verify custom threshold settings	Not Started
FR-8	Incident Reporting	Kibana Reporting	Test report generation and export	Completed
NFR-1	Reliability (99.9% uptime)	Elasticsearch Clustering	Monitor uptime over 30 days	Completed
NFR-2	Ease of Use (In-built UI)	Kibana Interface	Usability test with 5 analysts	Not Started
NFR-3	Speed (5-second log processing)	Elasticsearch	Measure log processing time	In Progress

4. Design Models

4.1 Architectural Design

The system uses a **Multi-Tier Architecture** with the following layers:

- **Data Collection Layer**:
  - **Tool**: Wazuh Agents
  - **Purpose**: Collects logs from endpoints (e.g., servers, workstations).
  - **Function**: Monitors file changes, login attempts, and system events.
- **Processing Layer**:
  - **Tool**: Wazuh Manager
  - **Purpose**: Filters and analyzes raw logs.
  - **Function**: Applies security rules and prepares logs for forwarding.
- **Log Forwarding Layer**:
  - **Tool**: Filebeat
  - **Purpose**: Transfers logs to Elasticsearch.
  - **Function**: Ensures efficient, reliable log delivery.
- **Storage Layer**:
  - **Tool**: Elasticsearch
  - **Purpose**: Stores and indexes logs.
  - **Function**: Enables fast search and retrieval for analysis.
- **Analysis Layer**:
  - **Tool**: AI/ML Model (Python-based, e.g., isolation forests)
  - **Purpose**: Detects anomalies like unauthorized file creation or too many login attempts.
  - **Function**: Analyzes logs for unusual patterns with high accuracy.
- **Visualization Layer**:
  - **Tool**: Kibana
  - **Purpose**: Displays logs, alerts, and reports.
  - **Function**: Provides interactive dashboards for security analysts.

4.2 Data Design

Elasticsearch manages all log data, optimized for fast search and analysis. Key data structures include:

- **Log Data**:
  - **Purpose**: Tracks system activities (e.g., file creation, logins).
  - **Example**:
    ```
    {   "timestamp": "2025-04-24T10:00:00",
        "event": "file_created",
        "path": "/etc/config",
        "user": "guest" }
    ```
- **Alert Data**:
  - **Purpose**: Stores details of detected threats.
  - **Example**:
    ```
    { "alert_id": "A001",
      "type": "suspicious_login",
      "source_ip":"192.168.1.10",
      "attempts": 6,
      "time": "2025-04-24T10:01:00" }
    ```
- **User Data**:
  - **Purpose**: Manages access roles.
  - **Example**:
    ```
    {   "user_id": "U001",
        "role": "analyst",
        "permissions":
       ["view_alerts", "generate_reports"]}
    ```

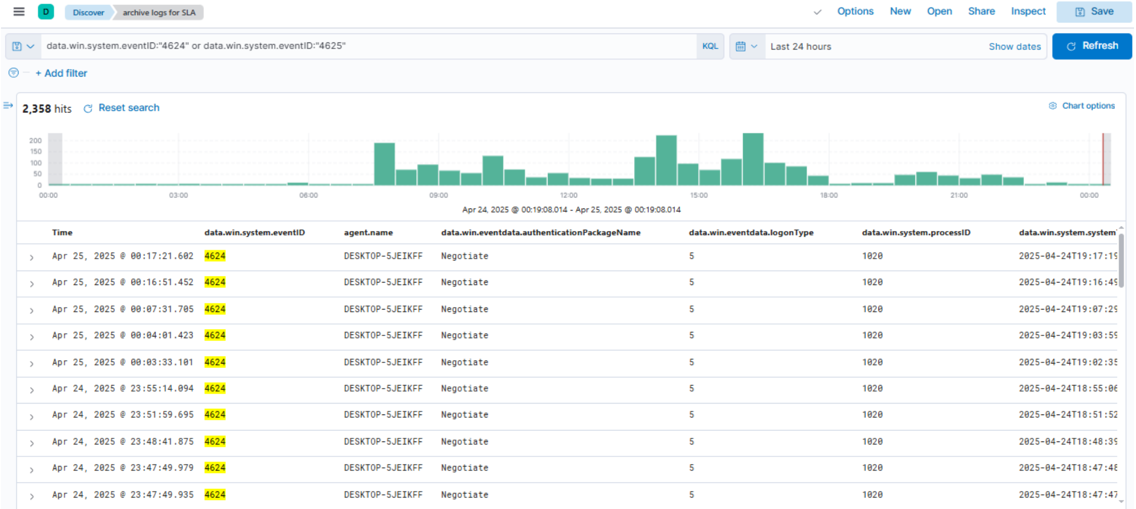
Data Dictionary

Term	Description	Constraints
Log Event	Records system activities (e.g., file changes, logins).	Must include timestamp, event type, source.
Anomaly Alert	Details detected threats (e.g., alert type, source).	Must include alert ID, timestamp.
User Role	Defines user access levels (e.g., analyst, admin).	Must specify role and permissions.

4.3 User Interface Design

The Kibana dashboard provides an intuitive interface for security analysts:

- **Alert Panel**:
  - Displays alerts with threat levels (e.g., low, high), timestamps, and sources.
  - Allows marking alerts as “safe” or “threat” with one click.
- **Log View**:
  - Shows filterable log entries in a table (columns: time, event, source, details).
  - Includes search bar for quick queries (e.g., “failed login”).



data.win.system.eventID:"4624" or data.win.system.eventID:"4625"

2,358 hits

	Time	data.win.system.eventID	agent.name	data.win.eventdata.authenticationPackageName	data.win.eventdata.logonType	data.win.system.processID	data.win.system.system
›	Apr 25, 2025 @ 00:17:21.602	4624	DESKTOP-5JEIKFF	Negotiate	5	1020	2025-04-24T19:17:19
›	Apr 25, 2025 @ 00:16:51.452	4624	DESKTOP-5JEIKFF	Negotiate	5	1020	2025-04-24T19:16:49
›	Apr 25, 2025 @ 00:07:31.705	4624	DESKTOP-5JEIKFF	Negotiate	5	1020	2025-04-24T19:07:29
›	Apr 25, 2025 @ 00:04:01.423	4624	DESKTOP-5JEIKFF	Negotiate	5	1020	2025-04-24T19:03:59
›	Apr 25, 2025 @ 00:03:33.101	4624	DESKTOP-5JEIKFF	Negotiate	5	1020	2025-04-24T19:02:35
›	Apr 24, 2025 @ 23:55:14.094	4624	DESKTOP-5JEIKFF	Negotiate	5	1020	2025-04-24T18:55:06
›	Apr 24, 2025 @ 23:51:59.695	4624	DESKTOP-5JEIKFF	Negotiate	5	1020	2025-04-24T18:51:52
›	Apr 24, 2025 @ 23:48:41.875	4624	DESKTOP-5JEIKFF	Negotiate	5	1020	2025-04-24T18:48:39
›	Apr 24, 2025 @ 23:47:49.979	4624	DESKTOP-5JEIKFF	Negotiate	5	1020	2025-04-24T18:47:48
›	Apr 24, 2025 @ 23:47:49.935	4624	DESKTOP-5JEIKFF	Negotiate	5	1020	2025-04-24T18:47:47

Figure 01

- **Graphs**:
  - Visualizes trends (e.g., login attempts by IP, file creation by directory).
  - Supports time-range filters (e.g., last 24 hours or more).

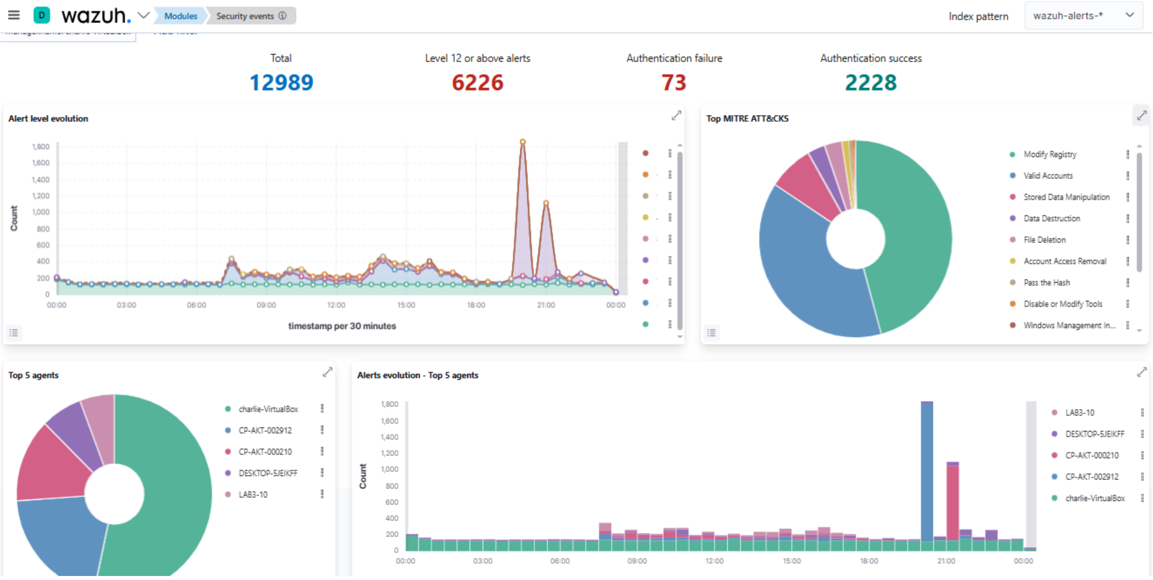


Figure 02

4.4 Behavioral Model

The system operates in a sequential workflow:

1. **Log Collection**: Wazuh agents gather logs from endpoints.
2. **Data Processing**: Wazuh Manager filters logs; Filebeat sends them to Elasticsearch.
3. **Anomaly Detection**: AI/ML model analyzes logs for anomalies
4. **Alert Generation**: Alerts are created and sent via email or displayed on Kibana.
5. **User Action**: Analysts review alerts, investigate, and log incidents.

Interaction Diagrams

- **Sequence Diagram** (*Figure 03*): Illustrates log flow:
  - **Actors**: Endpoint, Wazuh Agent, Wazuh Manager, Filebeat, Elasticsearch, AI Model, Kibana, Analyst.
  - **Flow**:
    1. Endpoint sends log to Wazuh Agent.
    2. Agent forwards to Wazuh Manager.
    3. Manager processes and sends to Filebeat.
    4. Filebeat stores in Elasticsearch.
    5. AI Model analyzes and detects anomaly.
    6. Alert is sent to Kibana and emailed.
    7. Analyst views alert and responds.

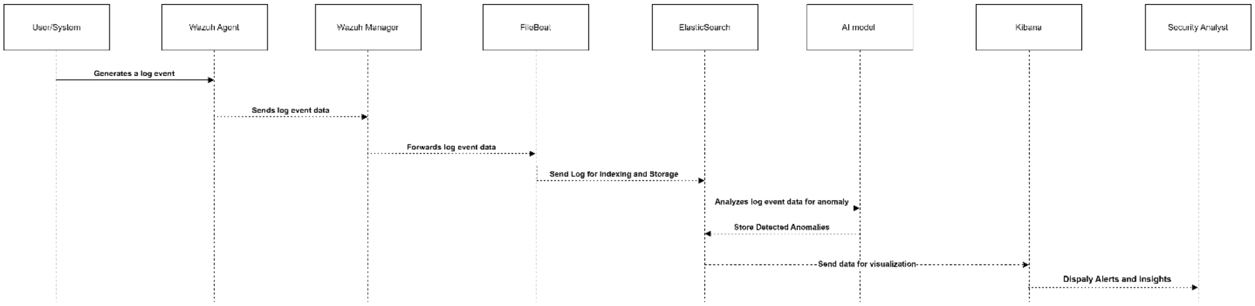


Figure 03

5. Design Decisions

- **AI Model**:
  - **Choice**: Unsupervised ML (e.g., isolation forests).
  - **Reason**: Detects anomalies without labeled data, ideal for diverse log patterns.
- **Data Storage**:
  - **Choice**: Elasticsearch.
  - **Reason**: Fast indexing and search for large log volumes.
- **Frontend**:
  - **Choice**: Kibana.

  - **Reason**: Provides user-friendly dashboards with real-time visualization.

6. Summary

This system combines Wazuh, Elastic Stack, and AI to monitor security logs and detect threats like unauthorized file creation and suspicious logins. Its multi-tier architecture ensures efficient log collection, processing, and visualization. Kibana dashboards provide clear insights, and AI reduces false positives. Designed for 10–50 endpoints, it scales with additional nodes for larger networks, keeping systems secure with fast, reliable alerts.

7. References

- *Wazuh* All-in-one deployment with Elastic Stack. *https://documentation.wazuh.com/4.5/deployment-options/elastic-stack/all-in-onedeployment/index.html*
- Machine learning & security protecting systems with data and algorithms by *clarencechio & david freeman* *https://a.co/d/cFCmldy*

- ***Security Monitoring with Wazuh:*** A hands-on guide to effective enterprise security using real-life use cases in Wazuh by *Rajneesh Gupta* *https://a.co/d/540PA1L*