**1. Configuration and Setup**

* **Imports**:
  + Standard modules: os, logging, datetime, smtplib, etc.
  + External modules: elasticsearch, requests, msal, bs4, pandas, etc.
  + Auto\_notify\_kibana\_events\_config: Holds credentials and config variables like e\_username, SMTP\_SERVER, etc.
* **Logging**:
  + Creates a timestamped log file inside C:\\wwwroot\\AppLogs\\Auto\_notify\_kibana\_events to log all events/errors for traceability.

**2. Fetching Alerts from Kibana**

**Function: get\_alerts\_from\_kibana()**

* Makes a query to the common-events-\* index using the Elasticsearch DSL.
* Filters alerts generated in the last 5 minutes using a range query on @timestamp.
* If alerts are found, it logs the count and processes them via extracting\_data\_into\_variable().

**3. Extracting Alert Fields**

**Function: extracting\_data\_into\_variable(result, latest\_records)**

* Loops through each alert (Elasticsearch \_source).
* Extracts the following fields:
  + EventSource, EventNode, EventTitle, EventLevel, EventBlob, EventResource, EventInstance, EventStatus
  + The \_id of the event is saved as unique\_id for building URLs.
* Logs and prints all this information.
* Then calls check\_event\_in\_rule\_index(...) to determine if this event matches any predefined rule.

**4. Rule-Based Matching**

**Function: check\_event\_in\_rule\_index(...)**

* Searches the common-events-notification index to check if there's a rule for this event (based on source, node, resource).
* If found:
  + Extracts EventNotifier and EventOwner.
  + Calls event\_found\_in\_rule\_index\_send\_email\_to\_owner(...).
* If not found:
  + Sends a default alert email via event\_not\_found\_in\_rule\_index\_send\_email\_default(...).

**5. Sending Email Notifications**

**Function: event\_not\_found\_in\_rule\_index\_send\_email\_default(...)**

* Sends an email to default recipients when no rule match is found.
* Uses the alert severity (EventLevel) to set the subject (Alert Fired or Alert Resolved).
* Includes:
  + Full alert details
  + A Kibana URL built using the unique\_id for quick access to the alert in Kibana Discover.

**Function: event\_found\_in\_rule\_index\_send\_email\_to\_owner(...)**

* Similar to the default email function, but uses the matched rule’s EventOwner and EventNotifier as recipients.
* Subject and content structure are the same.

**6. Security and Secrets**

* Email credentials and Elasticsearch authentication are secured via a config file (Auto\_notify\_kibana\_events\_config.py).
* Uses basic authentication (base64) for Elasticsearch.

**Key Changes Based on Andy’s Feedback:**

* Removed EventIncidentNo from the email body.
* Added **Monitor Everything (ME) URL** in the email body using unique\_id.

<p><b>EventURL:</b> <a href="https://kibana.fareportal.com/...common-events-\*?id={unique\_id}">...</a></p>

### Example Execution Flow:

1. Every 5 minutes, the script queries recent alerts from Elasticsearch.
2. Each alert is processed and logged.
3. If matched in the rule index, a notification is sent to the owner.
4. If not, a default notification is sent to a fallback email address.