**Project Documentation: FortiAnalyzer-like Application Using ELK Stack**

**Project Overview**

The goal of this project is to build an application similar to FortiAnalyzer using the open-source ELK stack (Elasticsearch, Logstash, Kibana). The primary focus is on collecting and visualizing logs from FortiGate firewalls.

**Project Phases**

**Phase 1**: Integrating firewall logs from FortiGate devices and developing a custom dashboard in Kibana.

**Timeline**

* **Start Date**: 7th August, 2024
* **Current Phase**: Testing the ELK Stack setup on Linux server okrcrpapplt01.

**Progress (7th August – 16th August, 2024)**

**1. Elasticsearch Installation and Configuration**

* **Objective**: Set up a powerful search and analytics engine to store and manage FortiGate firewall logs.
* **Tasks Completed**:
  + **Installation**: Successfully installed Elasticsearch on the Linux server okrcrpapplt01.
  + **Configuration**:
    - Configured Elasticsearch to allow for indexing and searching of log data.
    - Verified the setup by creating an initial index and ensuring the cluster status was green.
  + **Testing**:
    - Tested the Elasticsearch instance by sending sample data and confirming the ability to query and retrieve it.

**2. Logstash Installation and Configuration**

* **Objective**: Establish a data processing pipeline to ingest and transform logs from FortiGate devices.
* **Tasks Completed**:
  + **Installation**: Installed Logstash on the Linux server okrcrpapplt01.
  + **Input Configuration**:
    - Defined an input block in the Logstash configuration to receive logs from the FortiGate firewall.
    - Configured Logstash to listen on the designated port and protocol as per the FortiGate firewall settings.
  + **Filter Configuration**:
    - Developed filters to parse the log data, extracting relevant fields such as date, time, srcip, dstip, action, etc.
    - Applied grok patterns and conditional logic to format the logs for better visualization in Kibana.
  + **Output Configuration**:
    - Set up the output block to send the processed logs to Elasticsearch.
    - Verified that the logs were correctly indexed in Elasticsearch.
  + **Testing**:
    - Conducted tests to ensure that Logstash was correctly ingesting and processing log data from FortiGate devices.

**3. FortiGate Firewall Configuration**

* **Objective**: Configure the FortiGate firewall NDJB Lab 2 to send logs to Logstash for processing.
* **Tasks Completed**:
  + **Syslog Configuration**:
    - Configured the FortiGate firewall to forward logs to the Logstash server using syslog.
    - Set the appropriate logging levels and log types (e.g., traffic, event logs) to be forwarded.
  + **Verification**:
    - Verified that the firewall was successfully sending logs to the Logstash server.
    - Ensured that all relevant log data was being captured and processed by Logstash.

**4. Kibana Installation and Configuration**

* **Objective**: Use Kibana to visualize log data and develop a dashboard that emulates FortiAnalyzer.
* **Tasks Completed**:
  + **Installation**: Installed Kibana on the Linux server okrcrpapplt01.
  + **Configuration**:
    - Set up Kibana to connect to the Elasticsearch instance for data visualization.
    - Verified the connection and ensured that Kibana could access the Elasticsearch indices.
  + **Data Discovery**:
    - Accessed the Discover section in Kibana to review the ingested logs.
    - Confirmed that logs from the FortiGate firewall were visible and correctly parsed.
  + **Visualization**:
    - Created initial visualizations based on key log fields (e.g., source IPs, destination IPs, actions, events).
    - Configured charts, graphs, and tables to represent the log data meaningfully.
  + **Testing**:
    - Tested the visualizations with real log data to ensure accuracy and relevance.

**5. Dashboard Creation**

* **Objective**: Develop a comprehensive dashboard in Kibana that replicates FortiAnalyzer’s functionality.
* **Tasks Completed**:
  + **Design**:
    - Began designing a custom dashboard using the created visualizations.
    - Focused on key metrics and data points essential for firewall log analysis.
  + **Implementation**:
    - Integrated various visualizations into a single dashboard.
    - Adjusted the layout for optimal user experience and data readability.
  + **Testing**:
    - Tested the dashboard to ensure that all visualizations were functioning correctly.
    - Verified that the dashboard provided actionable insights based on the ingested log data.

**Next Steps**

* **Dashboard Finalization**: Continue refining the Kibana dashboard based on further analysis and feedback.
* **Scaling and Optimization**: Consider scaling the setup for additional data sources or higher log volumes.
* **Documentation**: Document the entire process, including screenshots, to serve as a reference for future implementations.