ELK Stack Deployment with Ansible

T. Gatsi, X.P. Baloyi, J.L.Lekganyane
South African Radio Astronomy Observatory
Cape Town, South Africa

Abstract

The MeerKAT, a 64-dish radio telescope in South Africa, is the largest and most sensitive radio Telescope in the Southern Hemisphere until integrated with the Square Kilometre Array (SKA). Our Control and Monitoring(CAM) system for Radio Astronomy Project such as MeerKAT produces a lot of data and logs that require proper handling. In order to track, analyse and investigate technical software issues an ELK(Elasticsearch, Logstash, Kibana)software stack was deployed.

1.Introduction

MeerKAT utilizes the ELK stack cluster, to process, manage and analyze daily logs generated by the radio telescope. The cluster, consisting of

- Elasticsearch
- Logstash
- Kibana

provides a robust framework for data handling. In the Control and Monitoring (CAM) system environment, the ELK cluster, a cluster of three LXC nodes within a Proxmox virtual environment, facilitates efficient analysis and display of logs.

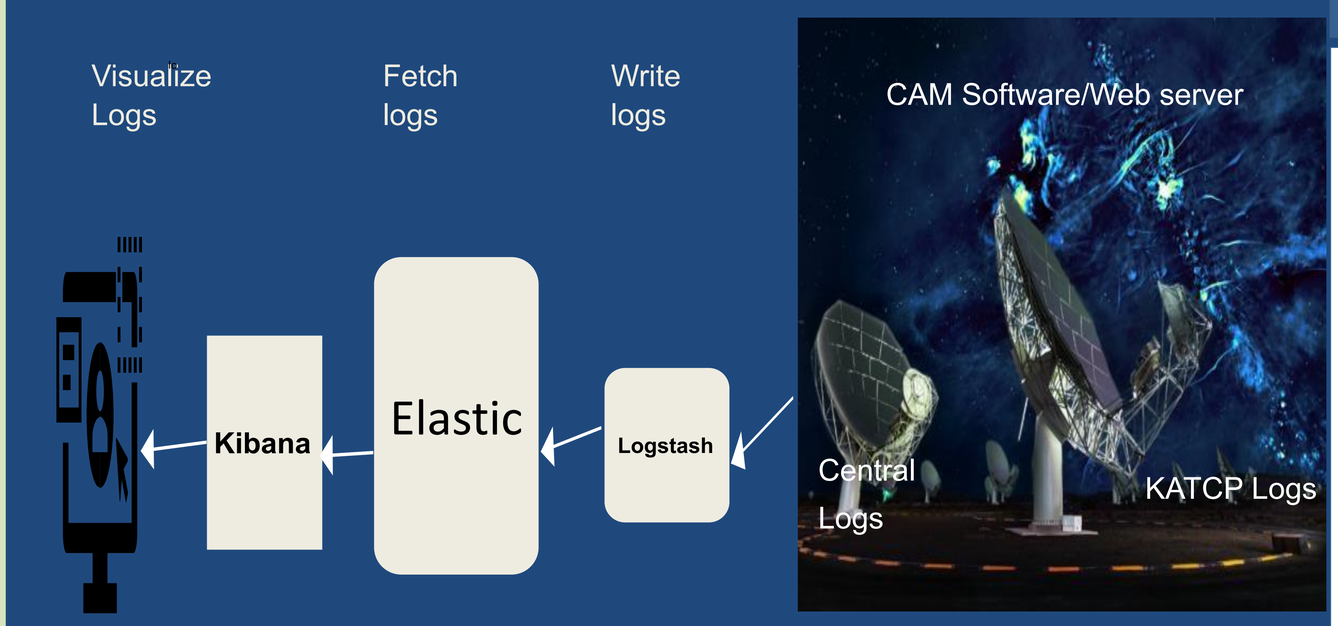

Figure 2 . How Logs move from CAM to the User

2. Access CAM logs

Figure 2 depicts a visual representation of log data originating from the CAM system, specifically comprising KATCP and Central logs. These log entries undergo an initial ingestion process via Logstash and are subsequently transmitted into Elasticsearch for advanced processing and analysis.Following the processing phase, the logs are retrieved and presented through the Kibana interface, enabling users to visualize the CAM system's log data in a graphical and user-friendly format.

ELK Stack Architecture in CAM

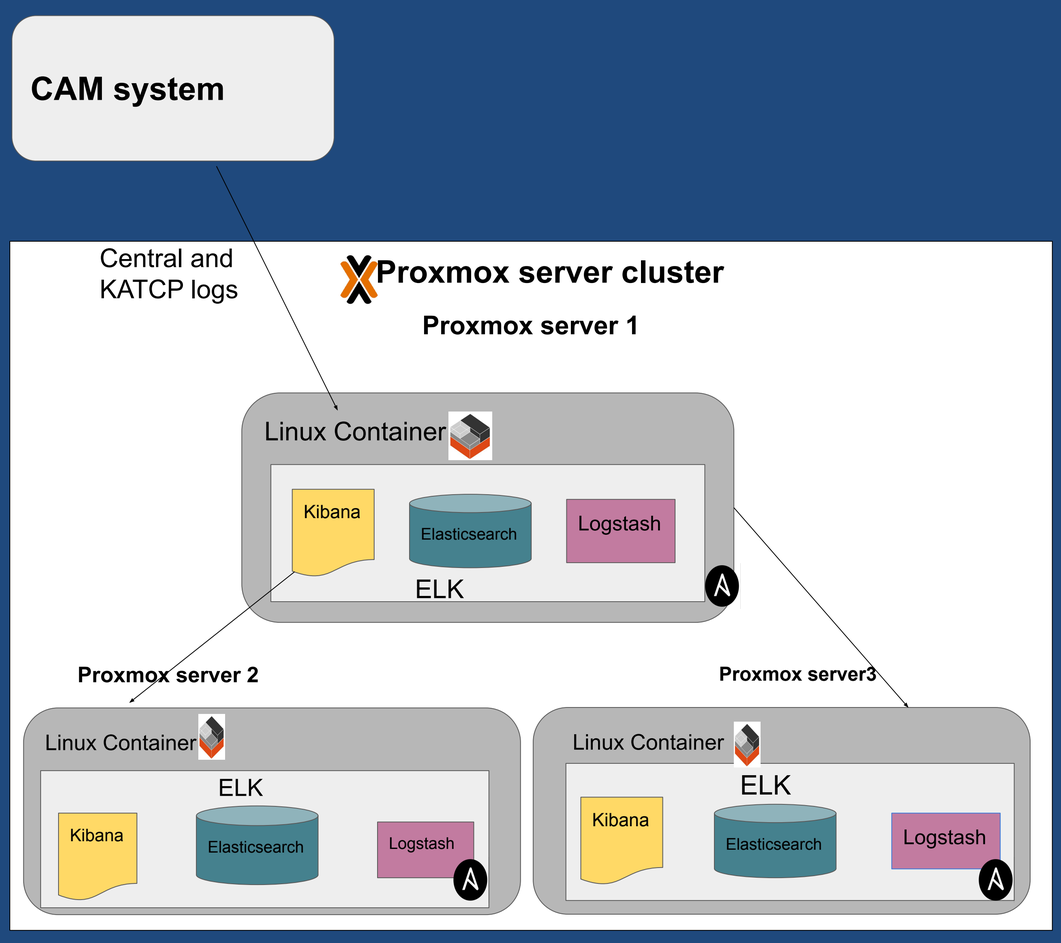

Figure 1 . How ELK stack is configured and deployed using ansible

3. ELK stack architecture in CAM

Figure 1 presents an overview of the configuration of the ELK (Elasticsearch, Logstash, Kibana) stack within a virtualized Proxmox environment. Within this setup, we have established a Proxmox Cluster consisting of three Proxmox servers. Each Proxmox server hosts a Linux container (LXC) running the Ubuntu operating system. Within these LXC containers, we have implemented instances of Elasticsearch, Logstash, and Kibana, collectively forming an ELK cluster that operates seamlessly within the Proxmox Cluster infrastructure. Notably, the deployment of this ELK cluster within the LXC containers has been optimized and streamlined through the utilization of Ansible, which enhances the deployment process's efficiency and ensures greater consistency throughout the configuration.

4. Conclusions

- The deployment of the ELK stack within the CAM system, facilitated by Ansible, has brought about notable enhancements.
- It has allowed us to expedite the ELK deployment process, curbing the occurrence of configuration errors while affording us robust configuration management capabilities.
- We have successfully achieved consistency by deploying a uniform ELK stack configuration across a spectrum of environments.As we continue to evolve, we are actively exploring the deployment of the ELK stack within Docker containers.
- This forward-thinking approach is rooted in the recognition of Docker's innate advantages, particularly its lightweight and portable nature, which promises to further optimize our operational efficiency and adaptability moving forward.

Contacts

T. Gasti, X.P. Baloyi, J.L. Lekganyane
South African Radio Astronomy Observatory
Email: enquiries@ska.ac.za
Website: www.sarao.ac.za
Tel: +27 (0) 21 506 7300