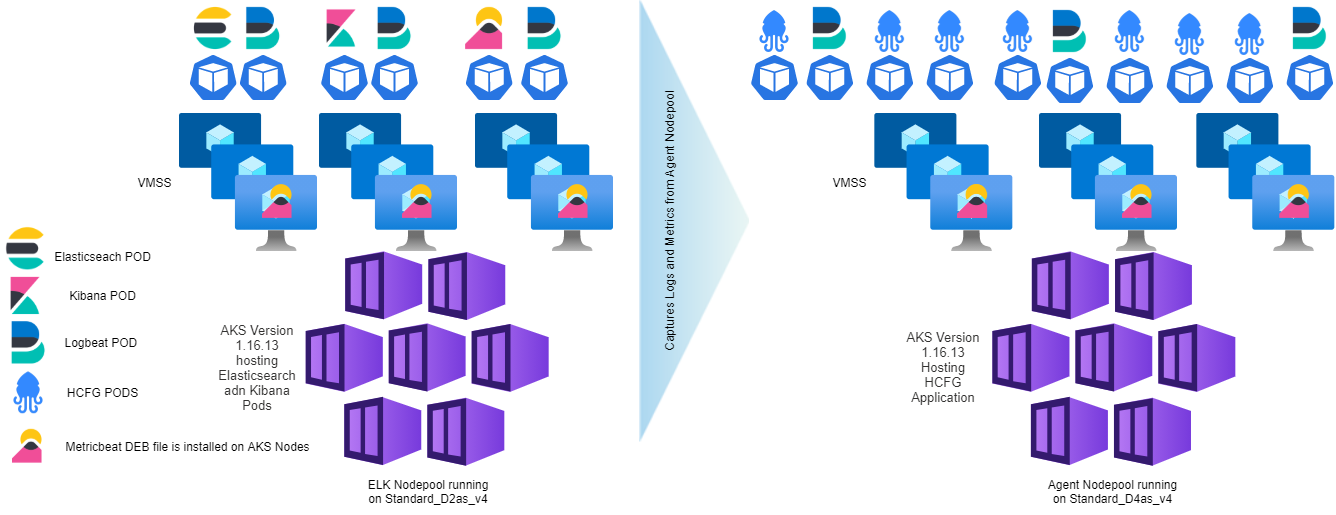
ELK Architecture

We have deployed ELK on Azure AKS as ELK Cloud on AKS using Kubernetes operators. We deploy Elasticsearch CRD’s and which will inturn deploy ELK components. In our Architecture we are using Elasticsearch to store and retrieve the data and Kibana to query and display the data we are intetrested in.



Deployment details of each components:

* Elastic search is deployed as POD with a Persistent volume of 100GB to collect the data and store it for querying ( /etc/elasticsearch/data )
* 100GB PV is dynamically created and mounted using Azurefile Stoarge Class
* Kibana also is deployed as POD and is encrypted with our wildcard certificate of app.gotteron.ch
* Deploy Filebeat as Daemon sets in all the nodes
* Configure SSH on AKS nodes and logging into them and install Metricbeat Deb files

Components and its Functionality:

**Elasticsearch**is the heart of the solution which comprises of a distributed JSON based search and analytics engine. It acts like a central repository for storing data and helps us query any details from it. These data can be logs, metrics, Application Performance Metrics and many more

**Beats**is a lightweight agent that ships the data from various sources to Elasticsearch. Types of beats are as follows:  
Filebeat for collecting logs and its related data  
Metricbeat for collecting metric data like CPU, Memory and other metrics  
  
**Kibana** is the UI or the dashboard component of ELK Stack which helps us configure Elasticsearch cluster its indexes and shards, perform many other operations on Elasticsearch.

How is Data stored in Elasticsearch?  
> Data inside Elasticsearch is stored as Elasticsearch **indices**.  
> An index may contain a single shard or may contain multiple **shards**  
> A single instance of a Lucene index is referred to as shards  
> Data is written to shards as immutable Lucene segments to the disk and is available for querying  
> All replicas of a shard must reside on different data nodes  
In case of failure of one of the node the Master node will make an existing replica shard as primary which will again start new replica Shard on a different data node

ELK sizing? Scaling up?

In our current architecture we don’t High Availability for ELK and Kibana, The solution could be more expensive with an extra node and extra Pods for Elastic search and Kibana. We can only decide the number of replica Pods once we see the performance of ELK while querying the AKS cluster.

Also we can choose ECE which is Elastic hosted Cloud SaaS offering which will automatically configure HA on multiple Availability Zones

ELK Backup and DR Strategy?

Elasticsearch provides typical snapshots that takes snapshots incrementally. The snapshotting process copies the data to a new repository that was not already copied there by an earlier snapshot, avoiding unnecessary duplication of work or storage space.

We can design a snapshot or make use of existing Kasten Backup to take snapshots of our ELK nodepool cluster state. Requires additional storage that is compatible with repository plugin for configuring snapshots.

We can use snapshot retention policy API to create, retain and delete the snapshots created by elasticsearch. Snapshots lifecycle management API : <https://www.elastic.co/guide/en/elasticsearch/reference/current/snapshot-lifecycle-management-api.html>

Retention Policy?

By default, metrics indices are kept for one day and logging indices are kept for seven days. We can set the retention period for logging and metrics using <https://www.elastic.co/guide/en/cloud-enterprise/current/ece-installation-script-set-logging-and-metrics-policy.html>