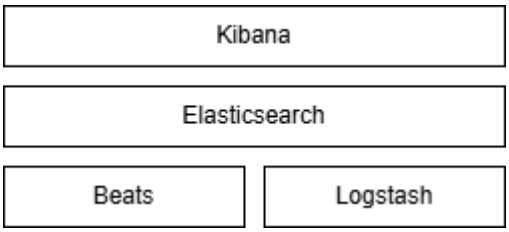
????????

Структура решения:



- Elasticsearch - полнотекстовый поиск, агрегация и хранение данных.
- Kibana - пользовательский интерфейс Elasticsearch. Спроектирован для Elasticsearch. Модульный за счет приложений
- Beats - сбор и отправка данных непосредственно из различных исходных систем (конечные точки, сетевые
  устройства или облака) в Logstash или Elasticsearch.
- Logstash - извлечение, преобразование и загрузка (ETL), используется для обработки и приема данных из различных источников (таких как файлы журналов на серверах, агенты Beats в вашей среде, очереди сообщений и платформы потоковой передачи) в Elasticsearch. Основная фишка - парсинг и преобразование полей для сохранения в Elasticsearch.

Elasticsearch

Основан на фреймворке Lucene для структурирования и поиска. Lucene индексирует элементы входного текста (индекс) и строит обратный индекс. Пример:

Элемент данных	Документ 1	Документ 2	Документ 3
Спагетти	+		
Сыр	+		+
Рецепт	+	+	+
Помидор		+	
Майонез			+
Картошка			+

Происходит объединение / пересечение полученных данных, ранжирование и отдача в соответствии с рангом.

Типы агрегаций:

- блоковая (bucket) агрегация. Группировка в зависимости от значений полей

- агрегация на основе метрик.

Могут использоваться схемы данных.

Архитектура

Данные -> Документы -> Сегменты -> Ноды

Горизонтальное масштабирование. Добавление нодов без простоя, автоматическое перераспределение сегментов данных по нодам.

Высокая доступность и надежность. Основной сегмент доступен на чтение и запись, остальные - чтение. Индексные и поисковые запросы выполняются параллельно.

Снимки, межкластерный поиск.

Схема хранения данных (Elastic Common Schema). ECS устанавливает сопоставления индексов для полей. Например целые числа могут быть как количеством переданных байт и подлежат суммированию, так может быть статусом ответа HTTP и являются строкой.

Модули Beats могут автоматически конвертировать логи и метрики в ECS схему.

Когда использовать Beast	**Когда использовать Logstash**
• Необходимо объединять данные из большого количества однотипных систем. • Есть модуль для данной системы • Не нужно проводить серьезные изменения перед передачей данных	• Когда большой объем данных поступает из централизованного хранилища (например, из общего файлового хранилища, AWS S3, Kafka и AWS Kinesis) и вам необходимо иметь возможность масштабировать пропускную способность. • Необходимость сложного преобразования данных • Балансировка нагрузки

Revision #1
Created 25 October 2025 16:00:03 by Admin
Updated 25 October 2025 18:20:27 by Admin