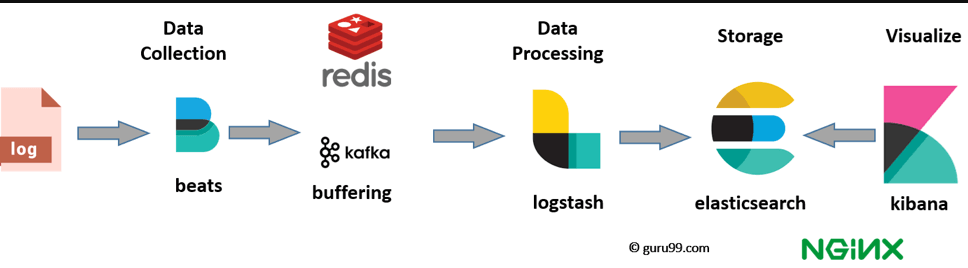
ELK (Elasticsearch, Logstash, Kibana) - gives you the ability to:

* aggregate logs from all your systems and applications
* analyze these logs
* create visualizations for application and infrastructure monitoring, faster troubleshooting, security analytics, and more



Одним из ключевых процессов в разработке ПО является логирование. Этим термином называется фиксация каждого этапа работы программы, как правило, сохраняемого в файл, который называется логом. Этот файл обычно содержит информацию о каждой совершенной программой операции и точном времени ее совершения, что позволяет в случае неполадки просмотреть, в какой момент и на какой операции что-то пошло не так.

Программное обеспечения для сбора и анализа логов не всегда универсально. По классической схеме работы, лог-файл создается при первом запуске программы, фиксирует ее поведение, затем автоматически сохраняется при закрытии программы. При следующем запуске приложения лог-файл заменяется на новый и все начинается сначала.

Однако, с течением времени программы становятся все более сложными, лог-файлы, соответственно, более объемными, а навигация по ним более затруднительной. С течением времени возникла необходимость в специализированных инструментах, которые позволяют быстро и удобно работать с логами. Одним из таких решений стал комплекс программ ELK Stack

Название ELK подобрано не просто так. Это не одна программа, а, как уже было сказано выше, комплекс, состоящий из трех основных программных продуктов Elasticsearch, Logstash и Kibana. Иногда данный комплекс дополняется сторонними программами, но эти "три кита" остаются неизменными инструментами. Разберем подробнее:

* **Elasticsearch** это поисковая система, предназначенная изначально для поиска фрагментов текста, однако с гибким функционалом и широкими возможностями по настройке. Это продукт улучшения решения Apache Lucene за счет добавления нескольких нововведений, делающих поиск информации в проектах с большими объемами данных достаточно оперативным и несложным.
* **Logstash** приложение для сбора информации из различных источников, преобразования их в удобный для работы формат и направления их в хранилище для дальнейшей работы. Простота использования и возможность работать с большими объемами данных обеспечивает Logstach ряд преимуществ перед аналогичными проектами.
* **Kibana** это плагин, предназначенный специально для Elasticsearch. Он отвечает за визуализацию данных, аналитику и представление итоговой информации в удобном для восприятия виде. Данное решение позволяет достаточно быстро анализировать итоги поиска, искать закономерности и представлять на экране Вашего устройства, где именно в проекте находятся слабые места. Этот плагин также обладает широкими возможностями по конфигурированию.

Таким образом, механизм сбора логов выглядит так: Logstash собирает объемные логи и помещает их в хранилище, Elasticsearch используется для поиска нужных строк в этих логах, Kibana позволяет проанализировать и визуализировать результаты поиска. Комплекс этих программных продуктов отличное решение для оперативного поиска и устранения неисправностей в программном коде, и очень удобный инструмент для разработчиков особенно тех, кто занимается созданием или внедрением отдельных элементов в крупные проекты. Кроме того, функциональность ELK позволяет его использовать в качестве централизованного хранилища журналов, агрегатора событий с удобной навигацией, аналитической системы с алгоритмом машинного обучения, а также по иным назначениям.

**Отличие ELK от zabbix**

Zabbix – это система мониторинга, которая ориентирована на какие то ситуации. В случае возникновения аномалий она доносит информацию до тех, кого это касается

ELK – мы храним все, что бы потом пост-фактум проанализировать это, при этом он не оповещает, как в случае с системами мониторинга

**Kibana**

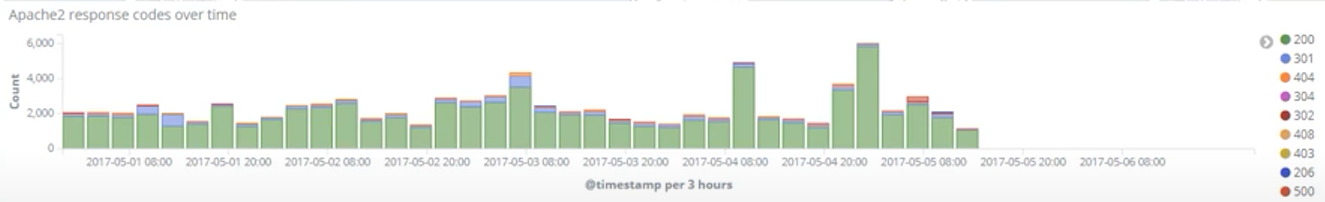
Можно установить время на обновление логов (например каждую минуту), установить за какое время будут показаны логи (например за последние 24 часа), …

**Dashboards**

В dashboards мы создаем свой дэшборд, где указываем какие логи будут попадать в него (например логи на авторизацию)

Внутри выбранного дэшборда можно фильтровать события по статусам, выбирать сервер, откуда мы будем забирать логи, …

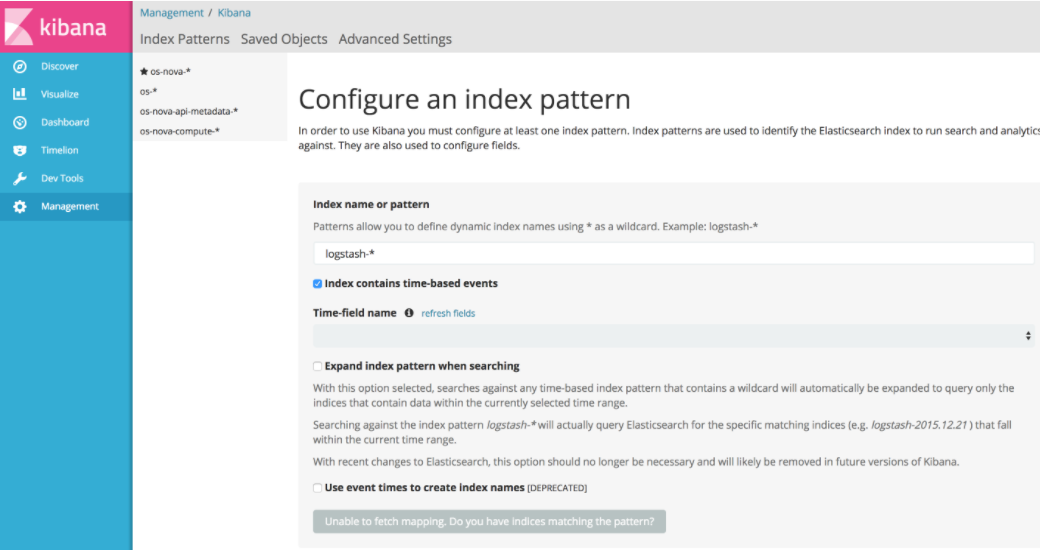
На дэшборде так же видны графики, где можно посмотреть пик запросов, а так же какие статус коды были у них (например можно посмотреть в какое время было максимальное кол-во 404 и 500 ошибок). Отфильтровав эти статус коды можно увидеть с каких устройств и мест были сделаны эти запросы и попытаться понять причину возникновения этих ошибок



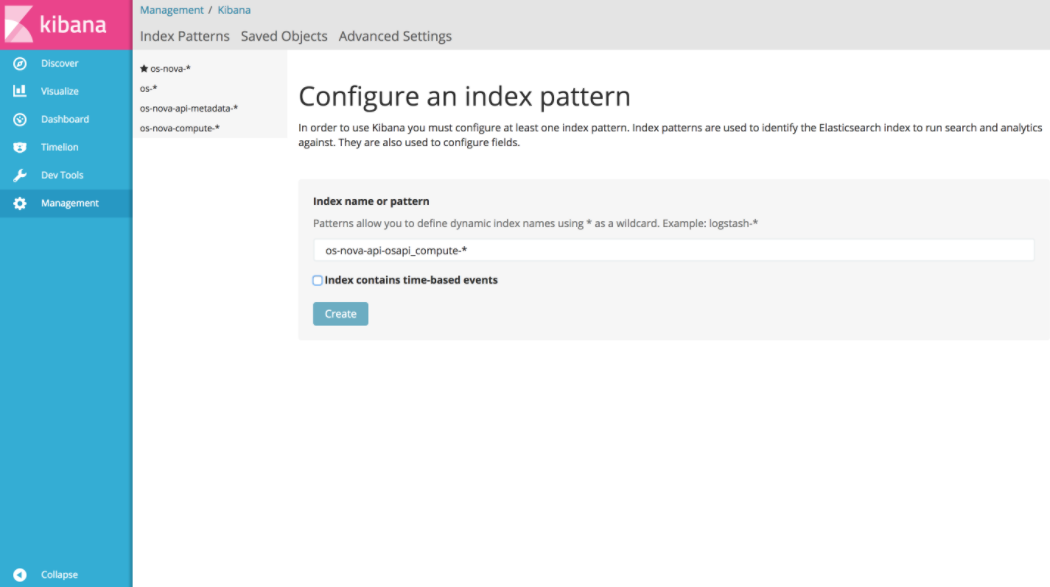
**How to set up Logging**

Step 1: create an index pattern

Open Kibana at kibana.example.com. Select the Management section in the left pane menu, then Index Patterns. Then, depending on Kibana's version, either click Add or +.



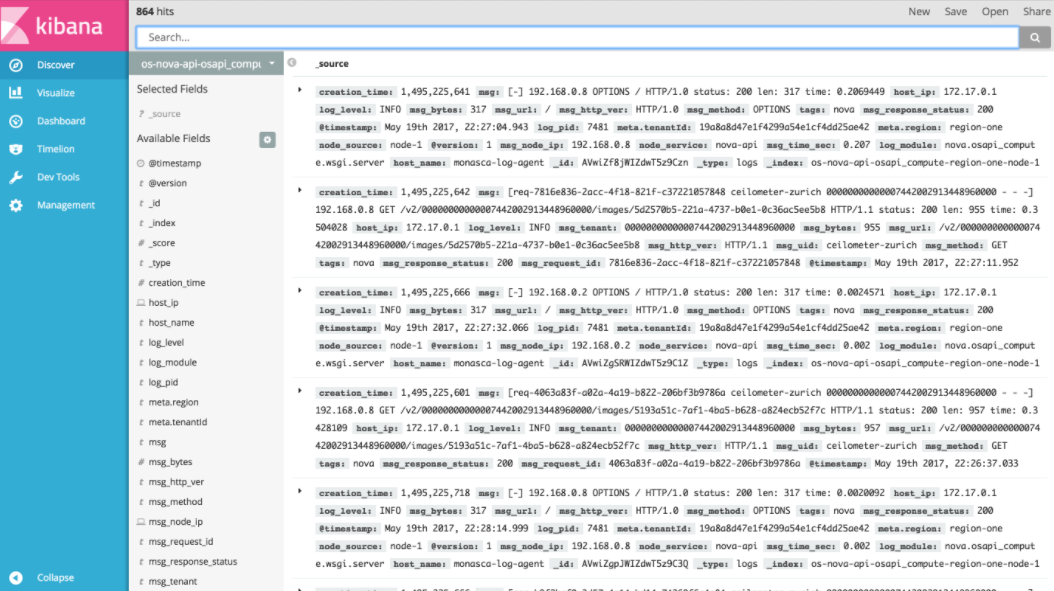
Enter the [index pattern](https://www.elastic.co/guide/en/kibana/5.4/tutorial-define-index.html), and uncheck Index contains time-based events. As soon as Kibana checks the index pattern against Elasticsearch and the result is positive, the button at the bottom will activate and display Create



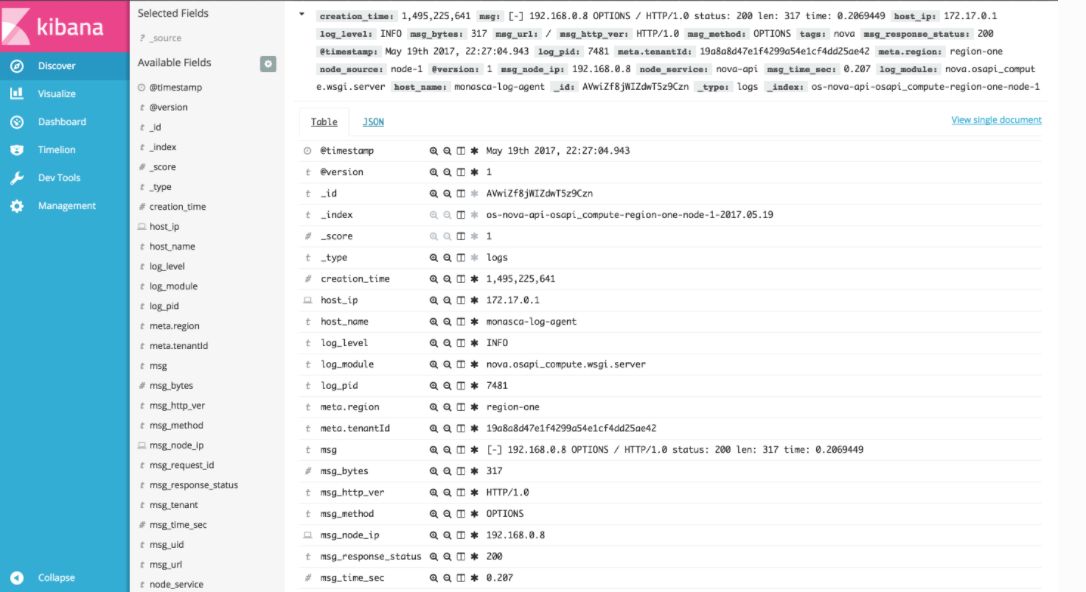
Click Create to configure the index pattern.

#### Step 2: view the logs

Navigate to the Discover section in the left pane menu. On the left of the page, just under the search bar, select the index pattern just created and all the logs matching the index will be displayed.



Every log entry can be inspected by clicking the small triangular bullet just besides it on the left. Each entry can be viewed as either table or JSON.



The search bar is the best place for querying and filtering the logs, using the Lucene [query syntax](https://lucene.apache.org/core/2_9_4/queryparsersyntax.html) or the full JSON-based [Elasticsearch Query DSL](https://www.elastic.co/guide/en/elasticsearch/reference/5.4/query-dsl.html). More details on searching data, managing searches, etc. are [here](https://www.elastic.co/guide/en/kibana/5.4/search.html).

**Kibana search**

* To perform a free text search, simply enter a text string. For example, if you’re searching web server logs, you could enter safari to search all fields for the term safari.
* To search for a value in a specific field, prefix the value with the name of the field. For example, you could enter status:200 to find all of the entries that contain the value 200 in the status field.
* To search for a range of values, you can use the bracketed range syntax, [START\_VALUE TO END\_VALUE]. For example, to find entries that have 4xx status codes, you could enter status:[400 TO 499].
* To specify more complex search criteria, you can use the Boolean operators AND, OR, and NOT. For example, to find entries that have 4xx status codes and have an extension of php or html, you could enter status:[400 TO 499] AND (extension:php OR extension:html).