

# Amazon Elasticsearch Service

## Deep Dive

Jon Handler – Principal Solutions Architect, Search Services

March 21, 2018

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



"Amazon Elasticsearch Service makes it easy to deploy, secure, operate, and scale Elasticsearch for log analytics, full text search, application monitoring, and more. In this webinar, we provide an in-depth overview of Amazon Elasticsearch Service and how to get started with it. Learning Objectives: - Learn how to configure a secure, petabyte-scale Amazon ES cluster and ingest data into it - Learn how to build Kibana dashboards to analyze and visualize your data in Amazon ES - Take away best practices to make your cluster reliable, take backups, and debug slow-running queries and indexing operations"

## A tale of a small workload

Search 5000 IMDb titles

Search movies star wars -VII

Directors

George Lucas	4
Irvin Kershner	1
Kyle Newman	1
Richard Marquand	1

Related actors

Mark Hamill	3
Carrie Fisher	3
Natalie Portman	3
Ewan McGregor	3
Harrison Ford	3

Actors

Carrie Fisher	3
Ewan McGregor	3
Harrison Ford	3
Mark Hamill	3
Natalie Portman	3

Genres

Adventure	7
Action	6
Sci-Fi	6
Fantasy	5
Comedy	1

  
**Title:** Star Wars  
**ID:** AV-oc0M0eUvfbVYhNzA  
**Score:** 36.090027  
**Rating:** 8.7  
**Plot:** Luke Skywalker joins forces with a Jedi Knight, a cocky pilot, a wookiee and two droids to save the universe from the Empire's world-destroying battle-station, while also attempting to rescue Princess Leia from the evil Darth Vader.

  
**Title:** Star Wars: Episode III - Revenge of the Sith  
**ID:** AV-oo0PqU1TBSXomp6EH  
**Score:** 22.442822  
**Rating:** 7.7  
**Plot:** After three years of fighting in the Clone Wars, Anakin Skywalker falls prey to the Sith Lord's lies and makes an enemy of the Jedi and those he loves, concluding his journey to the Dark Side.

  
**Title:** Star Wars: Episode I - The Phantom Menace  
**ID:** AV-oc0RiaJvttbVYhNsV  
**Score:** 21.54602  
**Rating:** 6.5

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



We will assume a small video company wants to provide search over their movies titles to drive up movie sales to the customers and also make sure their application is working fine with a search functionality.

# Elasticsearch can help

## BENEFITS

- ✓ Open source
- ✓ Fast time to value
- ✓ Easy ingestion
- ✓ Easy visualization
- ✓ High performance and distributed
- ✓ Best analytics and search

Rank	Project Name	Overall Project Rating
1	Linux	100.00
2	Git	31.10
3	MySQL	25.23
4	Node.js	22.75
5	Docker	22.61
6	Hadoop	16.19
7	Elasticsearch	15.72
8	Spark	14.99
9	MongoDB	14.68
10	Selenium	12.81
11	NPM	12.31
12	Redis	11.61

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Source: TechCrunch survey of popular open source software from April'17



Elasticsearch is OSS and available for you to download and use for free.

Search 5000 IMDb titles

Search movies star wars -VII

Directors

George Lucas	4
Ivan Kershner	1
Kyle Newman	1
Richard Marquand	1

Related actors

Mark Hamill	3
Carey Mulligan	3
Natalie Portman	3
Ewan McGregor	3
Harrison Ford	3

Actors

Carey Mulligan	5
Ewan McGregor	3
Harrison Ford	3
Mark Hamill	3
Natalie Portman	3

Genres

Adventure	2
Action	6
Sci-Fi	6
Fantasy	5
Comedy	1

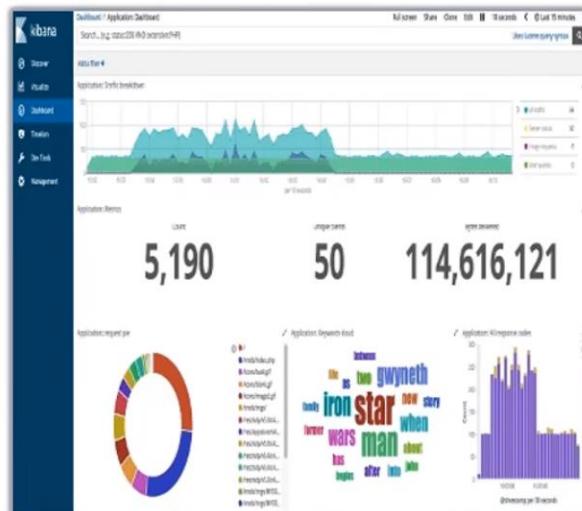
Title: Star Wars  
ID: Aa-ec0D6UWVWVWA  
Score: 35.98207  
Rating: 8.7

Plot: Luke Skywalker joins forces with a Jedi Knight, a cocky pilot, a wookiee and five丝 to save the universe from the Empire's world-destroying battle station while also attempting to rescue Princess Leia from the evil Darth Vader.

Title: Star Wars: Episode II - Attack of the Clones  
ID: Aa-ec0D6Uph7B56tqfH  
Score: 22.42622  
Rating: 7.7

Plot: After three years of fighting in the Clone Wars, Anakin Skywalker falls prey to the Sith Lord's lies and makes an outcry to the Jedi and those he loves, constituting his journey to the Dark Side.

Title: Star Wars: Episode I - The Phantom Menace  
ID: Aa-ec0D6Uv8TfHax  
Score: 21.3482  
Rating: 8.3



With Elasticsearch, you can run your business AND monitor it as well

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



You can also use ES to provide search functionality for things like application data and also monitor your infrastructure in near real-time.

# Amazon Elasticsearch Service



Amazon Elasticsearch Service is a **fully managed service** that makes it easy to deploy, manage, and scale Elasticsearch and Kibana in the AWS cloud



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



## Benefits of Amazon Elasticsearch Service



### Supports Open-Source APIs and Tools

Drop-in replacement with no need to learn new APIs or skills



### Easy to Use

Deploy a production-ready Elasticsearch cluster in minutes



### Scalable

Resize your cluster with a few clicks or a single API call



### Secure

Deploy into your VPC and restrict access using security groups and IAM policies



### Highly Available

Replicate across Availability Zones, with monitoring and automated self-healing



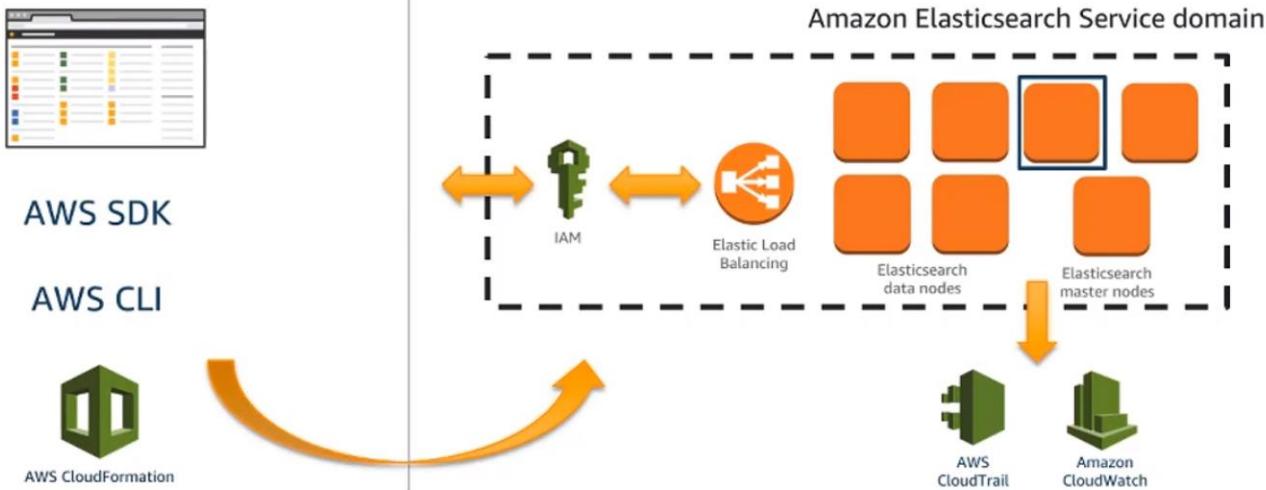
### Tightly Integrated with Other AWS Services

Seamless data ingestion, security, auditing and orchestration

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



You can provision an ES cluster easily on the console, CLI, using CF, etc. You can also easily scale out your instance types or cluster, we also provide security and HA features that makes it easy to deploy in different regions.



## Service Architecture

Key Idea

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

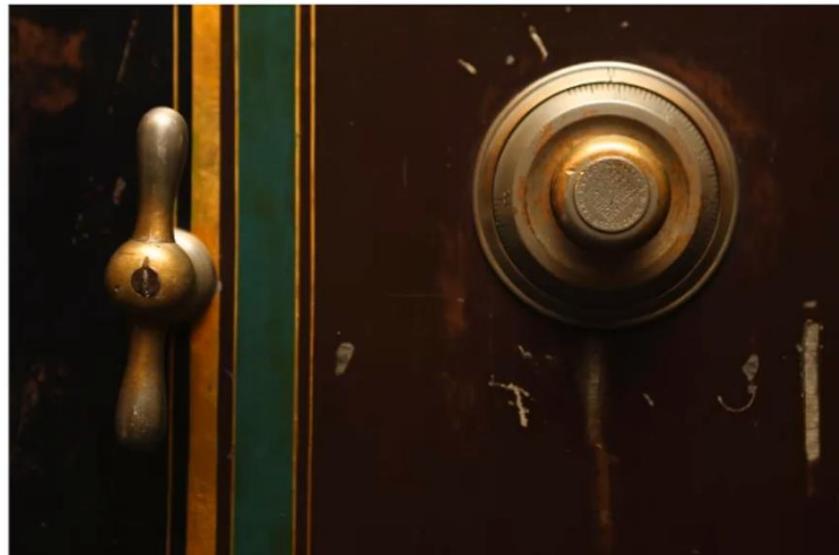


When you use the managed ES service, you deploy an Elasticsearch Domain that is used to wrap all the hardware and software needed to run your ES cluster. You can then deploy that domain through the console, CLI, or CF to spin up a set of ES instances for your data nodes and the master node. The data nodes hold data and respond to updates and queries, the master node are orchestrators of the ES cluster. AWS then fronts the ES cluster with an ELB and provide you with an endpoint that resolves through DNS so that you can then interact with the ES APIs over that Route53 endpoint. You can secure your endpoint that you get for your ES domain via DNS using IAM and VPC access, this is how you communicate with your ES cluster.

## Security

- VPC access (Recommended)
- Public access

- Public endpoints – IAM
- Private endpoints – IAM and security groups
- Encryption



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



You have 2 choices of where to put your ES endpoint, you can have a public endpoint that will resolve through DNS to a publicly available IP address or you can have a private endpoint that will resolve to an IP address within the address space of your VPC (this is the recommended solution).

## Use IAM for public endpoints

```
{ "Version": "2012-10-17",  
  "Statement": [ {  
    "Effect": "Allow",  
    "Principal": {  
      "AWS":  
        "arn:aws:iam::XXXX:root"  
    },  
    "Action": "es:*",  
    "Resource":  
      "arn:aws:es:  
        us-west-2:XXXX:domain/YYYY/*"  
  }  
]
```

- To grant access for Kibana, use a **CIDR Condition**
- To grant read-only access to an account or role, specify an **es:HttpGet Action**
- To limit a user to a specific index, or API add it to the **Resource**
- For Admin access, specify administrative **es:\*** **Actions**

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



We recommend setting an IAM policy so that your ES cluster is not open to the entire internet, you can write a policy with a CIDR condition that allows access from a specific IP address only, say for your Kibana endpoint.

## Add security groups for private endpoints

VPC access (Recommended)  
 Public access

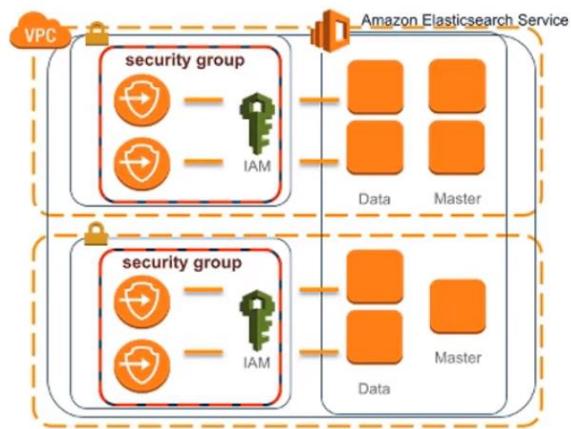
VPC  ?

Subnet  ?

Security Groups  ?

IAM Role [AWSRoleForAmazonElasticsearchService](#) ?

Specify a subnet and security group to apply CIDR restrictions on inbound/outbound traffic



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



# Encrypt your data

- Encrypted data at rest on Amazon ES instances
- Both EBS and ephemeral store
- Encrypted automatic snapshots



## The front end

### Search 5000 IMDb titles

Search movies star wars -VII

**Directors**

George Lucas	4
Irvin Kershner	1
Kyle Neiman	1
Richard Marquand	1

**Related actors**

Mark Hamill	3
Carrie Fisher	3
Natalie Portman	3
Ewan McGregor	3
Harrison Ford	3

**Actors**

Carrie Fisher	3
Ewan McGregor	3
Harrison Ford	3
Mark Hamill	3
Natalie Portman	3

**Genres**

Adventure	7
Action	6
Sci-Fi	6
Fantasy	5
Comedy	1

**Title: Star Wars** **ID: AV-oc0M0aUvtfbVYhNxA**  
Score: 38.090927  
Rating: 8.7  
**Plot:** Luke Skywalker joins forces with a Jedi Knight, a cocky pilot, a wookiee and two droids to save the universe from the Empire's world-destroying battle-station, while also attempting to rescue Princess Leia from the evil Darth Vader.

**Title: Star Wars: Episode III - Revenge of the Sith** **ID: AV-oc0PqU1TBSXmp6EH**  
Score: 22.442822  
Rating: 7.7  
**Plot:** After three years of fighting in the Clone Wars, Anakin Skywalker falls prey to the Sith Lord's lies and makes an enemy of the Jedi and those he loves, concluding his journey to the Dark Side.

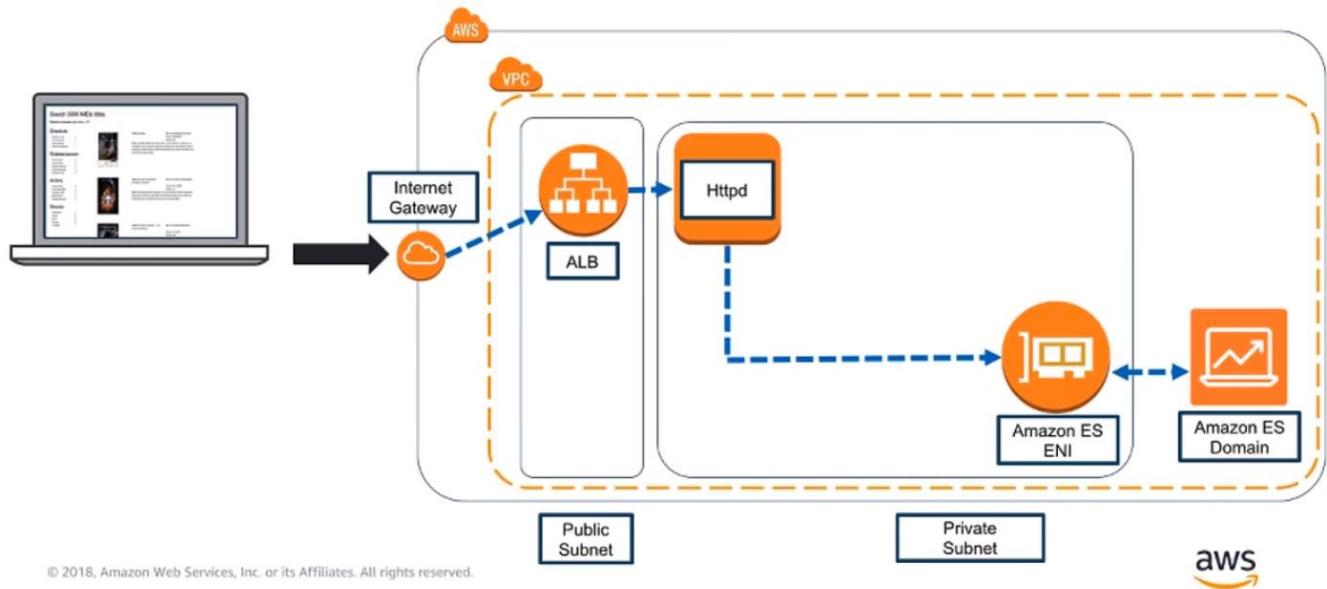
**Title: Star Wars: Episode I - The Phantom Menace** **ID: AV-oc0RisUvtbVYhNsv**  
Score: 21.546002  
Rating: 6.5

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



Let us talk about the web application that the users will use to search for movies

# Architecture: Web Serving



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



The web app takes a couple of search key words from the customer's client, formulates a query for ES, sends the query to the Elastic Network Interface ENI, this gets forwarded to the ES cluster running in the ES Domain that replies with the search results, the web app then formats the search results and display to the user.

## Application search

A screenshot of an Amazon search result for the movie "Iron Man". The main image is the movie poster. Below it, the title "Iron Man" is displayed along with its release information: PG-13, 126 min, 2 May 2008 (USA). It shows a Metascore of 79/100 and 983 reviews. The plot summary states: "When wealthy industrialist Tony Stark is forced to build an armored suit after a life-threatening incident, he ultimately decides to use its technology to fight against evil." Below the plot, there are links for the director (Jon Favreau), writers (Mark Fergus, Hawk Ostby), and stars (Robert Downey Jr., Gwyneth Paltrow, Terrence Howard). At the bottom, there are buttons for "Watch now" (Amazon Instant Video), "See all 3 watching options", and links for DVD/Blu-ray and release date (Wed, Oct. 30). A yellow banner at the bottom indicates the movie was nominated for 2 Oscars. Below the main movie card, there are sections for "Videos" and "Photos", each showing thumbnails of related content.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



When we deal with search engines, the core concept in the search document. The search document is the entity that you put into the search engine and it is the entity that you retrieve when you run a search request. Documents comprise of a set of information about an underlying entity like Iron Man above



Key Idea    Each document has a set of fields

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



We have a lot of different fields and the document itself is a little structured. We have fields that we might like to search with. ES itself provides you search at the field level, you start off with the single entity that you want to retrieve and then specify the set of fields that you want to comprise the values you want to get.

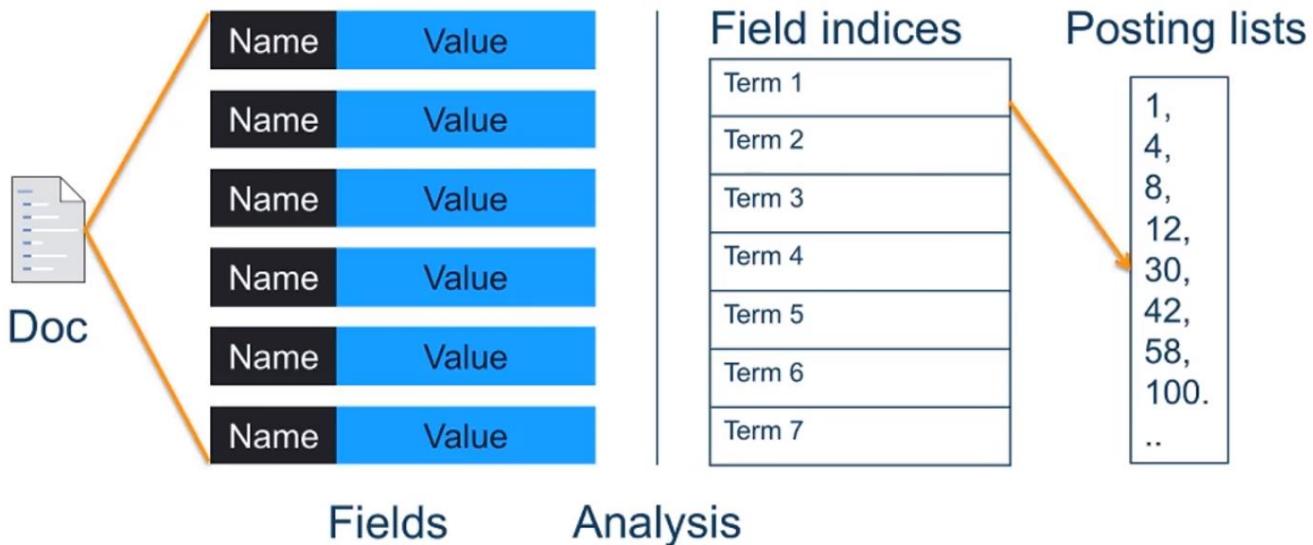
```
{   "title": "Star Wars",
    "plot": "Luke Skywalker joins forces with a
Jedi Knight, a cocky pilot, a wookiee and two
droids to save the universe from the Empire's
world-destroying battle-station, while also
attempting to rescue Princess Leia from the evil
Darth Vader.",
    "year": 1977,
    "actors": [
        "Mark Hamill",
        "Harrison Ford",
        "Carrie Fisher"
    ],
    "directors": [
        "George Lucas"
    ],
    "rating": 8.7,
    "genres": [
        "Action",
        "Adventure",
        "Fantasy",
        "Sci-Fi"
    ]
}
```

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



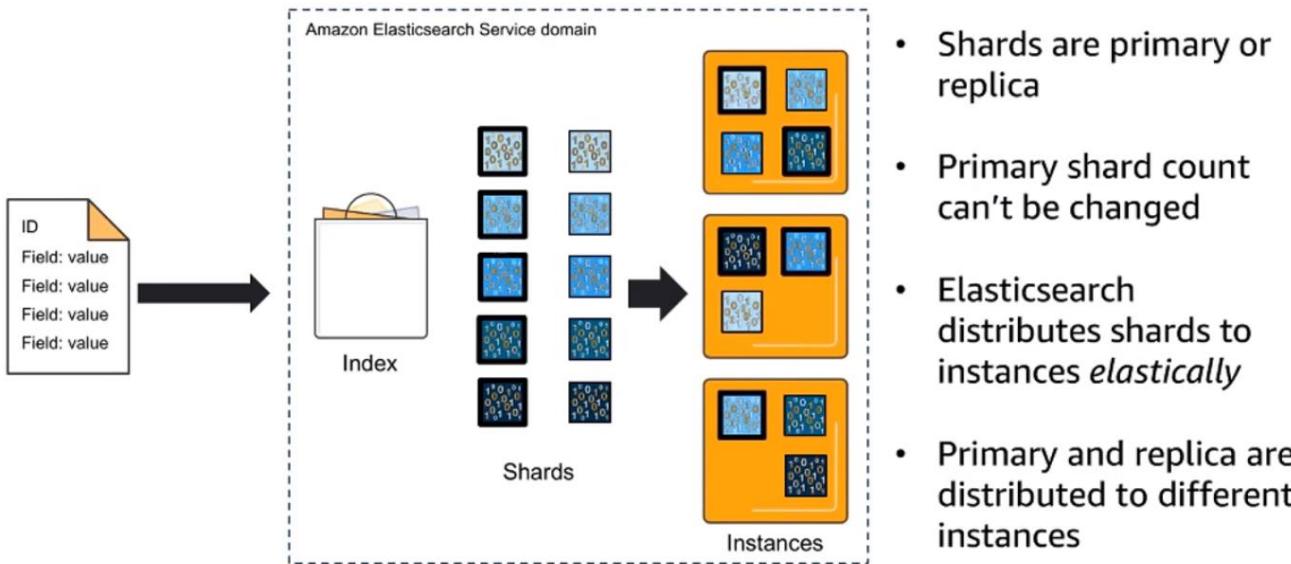
To store your data in ES, you need to send your data into ES as structured JSON with the fields and values as above. ES also supports a nested JSON structure natively, so you can send objects with a hierarchy.

# Elasticsearch creates an index for each field



At the field level, ES creates an index. We can picture our document as a set of fields with names and values. ES analyzes the values of each field in the document and then creates an index for that field. The index contains everything that are in that field for all the documents that have come into ES.

## Data is stored in indexes, distributed across shards



At the higher level we have an index that contains all of our documents, all organized within the index. We can have an index for movies that would contain all of our movie documents/objects. We can have an actors index that contains all the actors separately, we can have an index called users containing our user's objects. Each index is logically broken into shards, which has subsets of documents from the full set of documents in an index. Shards can be a primary shard or the first index of a shard, once we set our primary shard count it is fixed for that index forever and we can't change that. We can also have a replica shard which is a copy of the primary shard. The first replica shard gives us redundancy so that our data can be deployed to 2 different locations within the ES cluster, additional replicas provide us more query capacity.

# How many instances?

Instance count  ⓘ

Instance type  ⓘ  
m4.large.elasticsearch instance type needs EBS storage.

Storage type  ⓘ

EBS volume type\*  ⓘ

EBS volume size\*  ⓘ  
Total cluster size will be 20 GB (EBS volume size x Instance count).

- Index size is approximately source size
  - Double this if you are deploying an index replica
- Instance count based on storage requirements
  - Either local storage or up to 1.5 TB of Amazon Elastic Block Store (EBS) per instance

## Example: a 2 TB corpus will need 4 instances

Assuming a replica and using EBS

Given 1.5 TB of storage per instance, this gives 6TB of storage

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



You need to think about the number of instances you want to have in your ES cluster before creating the cluster, you can think about how much storage you will need to hold your data in the instances.

# Which instance type?

Instance	Max Storage*	Workload
T2	3.5TB	You want to do dev/QA
M3, M4	150TB	Your data and queries are “average”
R3, R4	150TB	You have higher request volumes, larger documents, or are using aggregations heavily
C4	150TB	You need to support high concurrency
I2, I3	1.5 PB	You have XL storage needs

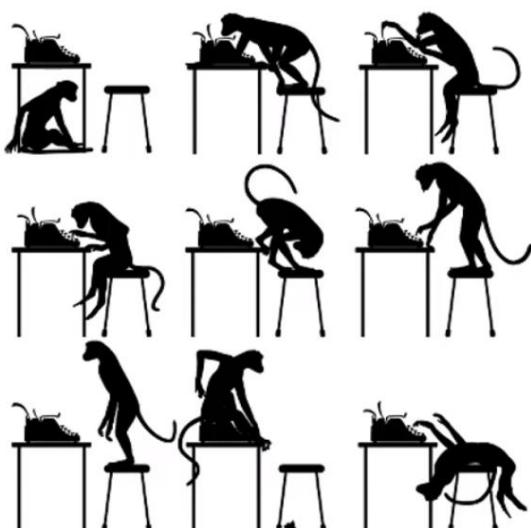
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.





## Shards are units of storage and work

- Set primary shard count based on storage, 40GB per primary
- Example: set shard count = 50 for a 2TB corpus ( $2\text{TB} / 40\text{GB} = 50$  shards)
- Always use at least 1 replica in production
- Set shard count so that shards per instance  $\approx$  number of CPUs
- Keep shard sizes as equivalent as possible



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

The shards are what actually holds/store the data and what does the work of ES regarding queries and updates. You need to get the best distribution of shards among the instances in your cluster.

## Use templates to set shard count

```
*PUT <endpoint>/_template/template1
{
    "index_patterns": ["movies*"],
    "settings": {
        "number_of_shards": 50,
        "number_of_replicas": 1
    }
}
```

- All new indexes which match the index pattern receive the settings
- You can also specify a mapping for all indexes

\*Note: ES 6.0+ syntax

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



You can use a template that is an ES API as a way of specifying the shard count automatically (also relevant for log analytics use case), a replica will then create an additional 50 shards. We can also put mapping information in the API call for the schema.

# Query your data

```
{  
  "query": {  
    "match": {  
      "title": "iron man"  
    }  
  }  
}
```



Title	Score
Iron Man	10.56436
Iron Man 2	8.631084
Iron Man 3	8.631084
Iron Sky	6.387543
The Man with the Iron Fists	6.1855173
The Man in the Iron Mask	6.1855173
The Iron Giant	5.218624
The Iron Lady	5.218624

77 hits

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



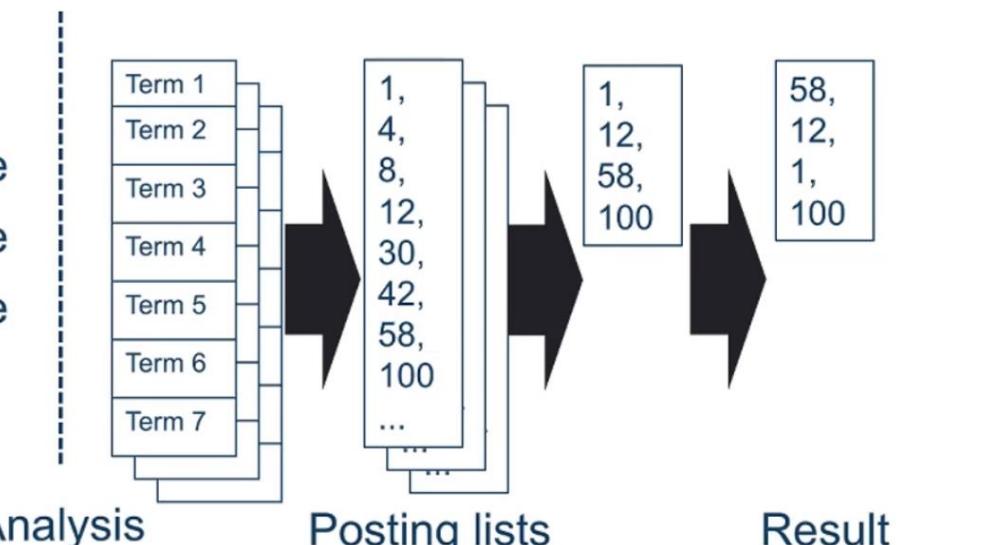
This is a sample of the Query API in ES, it is JSON as above. Note that the default operator is an 'OR' so that the above query will result in a search for the words 'iron' OR 'man'. ES has a notion of scoring for providing us a score sorted result set back for our queries.



**You can search for values by field, with Boolean expressions to get scored, sorted results**

Key Idea

Field1:value  
Field2:value  
Field3:value



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

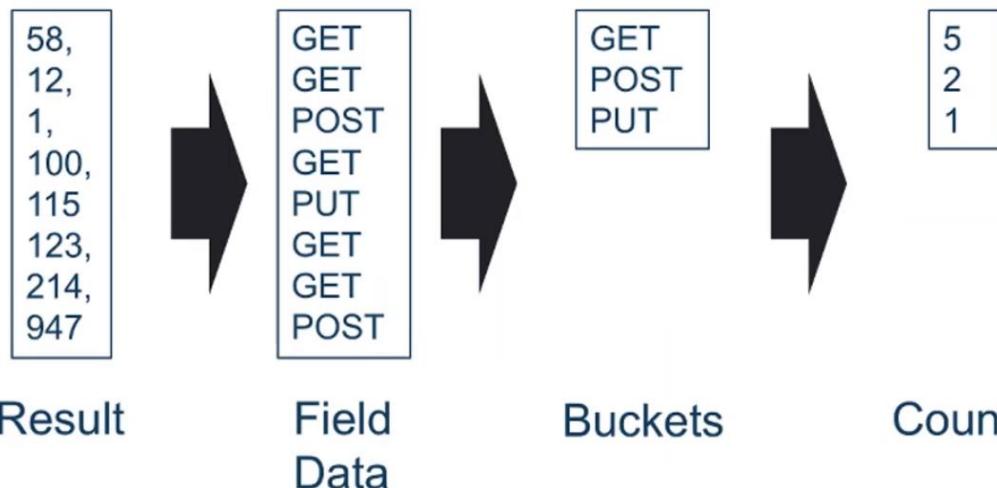


The Posting lists represent all the documents that have the search term(s) in them with the location\_ids of where the words are located within the documents.



Key Idea

## You can analyze field values to get statistics and build visualizations



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



Above is a log analysis example for HTTP data in our Apache web logs

### CASE STUDY: MirrorWeb

Full text search

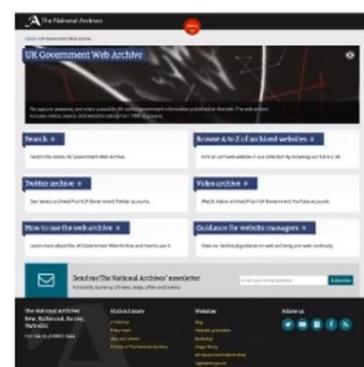
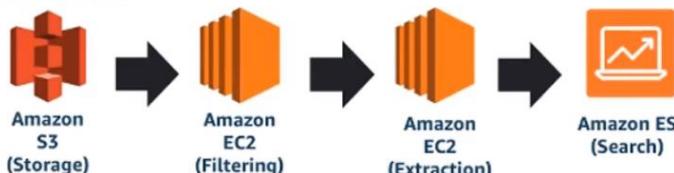


#### PROBLEM

Make the UK Government and UK Parliament's web archives searchable

Large scale ingestion scenario: 120 TB of data (1.2 MM 100MB files), duplicates and bad data, Warc format

#### SOLUTION



#### BENEFITS

**Scalability:** Started on a 9-node, R4.4Xlarge cluster for fast ingest, reduced to 6 R4.Xlarge instances for search. Able to reconfigure the cluster with no down time

**Cost effective:** Indexed 1.4 billion documents for \$337

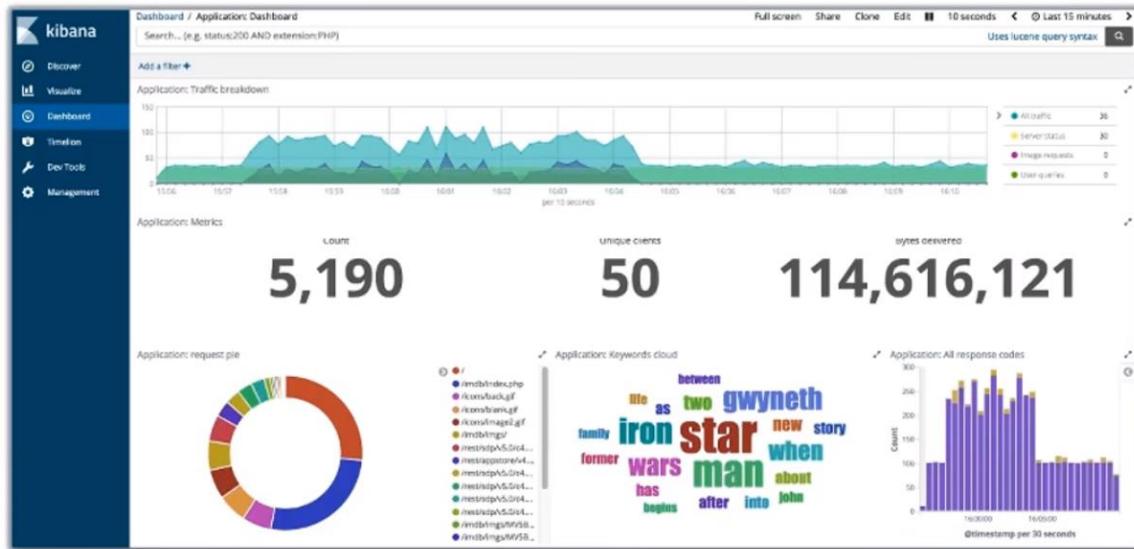
**Fast:** 146 MM docs per hour indexed. 14x faster than the previous best for this data set (using Hadoop)

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved. This document contains proprietary information of Amazon Web Services, Inc.

For more on this case, see <http://tinyurl.com/ybqwbolq>



# Log transport

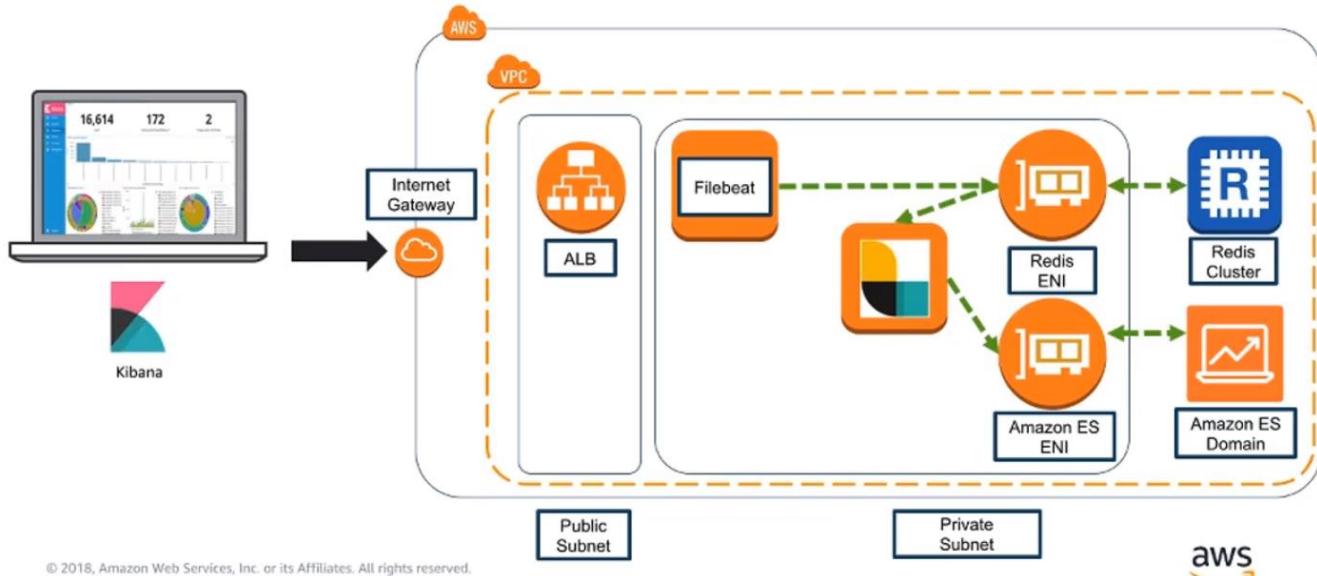


© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



The XYZ company now have their data in ES and their frontends can start using the search functionality from the web app. They now want to monitor the infrastructure using log analysis.

## Architecture: Log transport

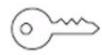


© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



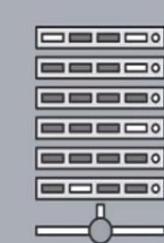
We are using an OSS tool called **Filebeat** to tail the log files and ship the log lines one at a time to a **Redis** buffer (not a cache) to receive the log lines and batch them up, sends/pulls the log lines to **Logstash** to be formatted and transformed, then sent to our **ES** ENI.





Key Idea

## Build an ingest pipeline that completes these tasks



Data source



Collect



Transform



Buffer



Deliver

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



### Collect

- Kinesis Agent
- CloudWatch Logs Agent
- Application
- Logstash
- Beats
- Fluentd

### Buffer

- Kinesis Firehose/Streams
- CloudWatch Logs
- Amazon S3
- Amazon Elasticache/Redis
- Logstash
- Kafka
- Rabbit MQ

### Transform

- Kinesis Firehose
- CloudWatch Logs
- Logstash
- Amazon Lambda

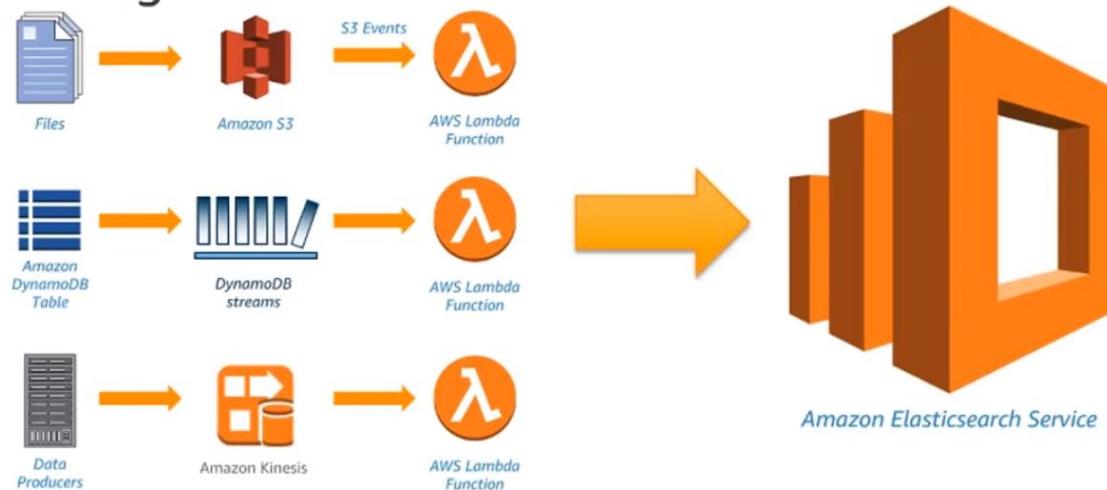
### Deliver

- Kinesis Firehose/Streams
- CloudWatch Logs
- Amazon Lambda
- Logstash
- Worker nodes

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



# Amazon Lambda architectures for dynamic updating



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



## When you use Elasticsearch for log analysis, split data into daily indexes

logs\_01.21.2018  
logs\_01.22.2018  
logs\_01.23.2018  
logs\_01.24.2018  
logs\_01.25.2018  
logs\_01.26.2018  
logs\_01.27.2018  
logs\_01.28.2018  
logs\_01.29.2018

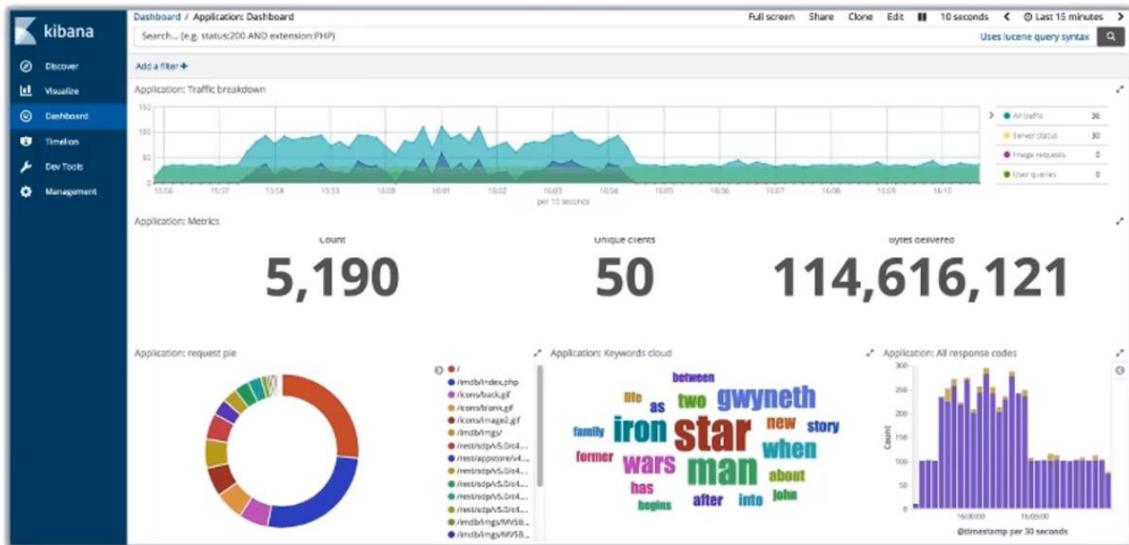
- On ingest, create indexes with a root string, e.g. logs\_
- Depending on volume, rotate at regular intervals – normally daily
- Daily indexes simplify index management. Delete the oldest index to create more space on your cluster

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



You usually should be creating a new index on a daily basis and use the timestamp index as a hash pattern. This will help with data lifecycle management in our ES cluster where we want to delete/expire older data as time passes from the cluster, we could archive the data before aging them out.

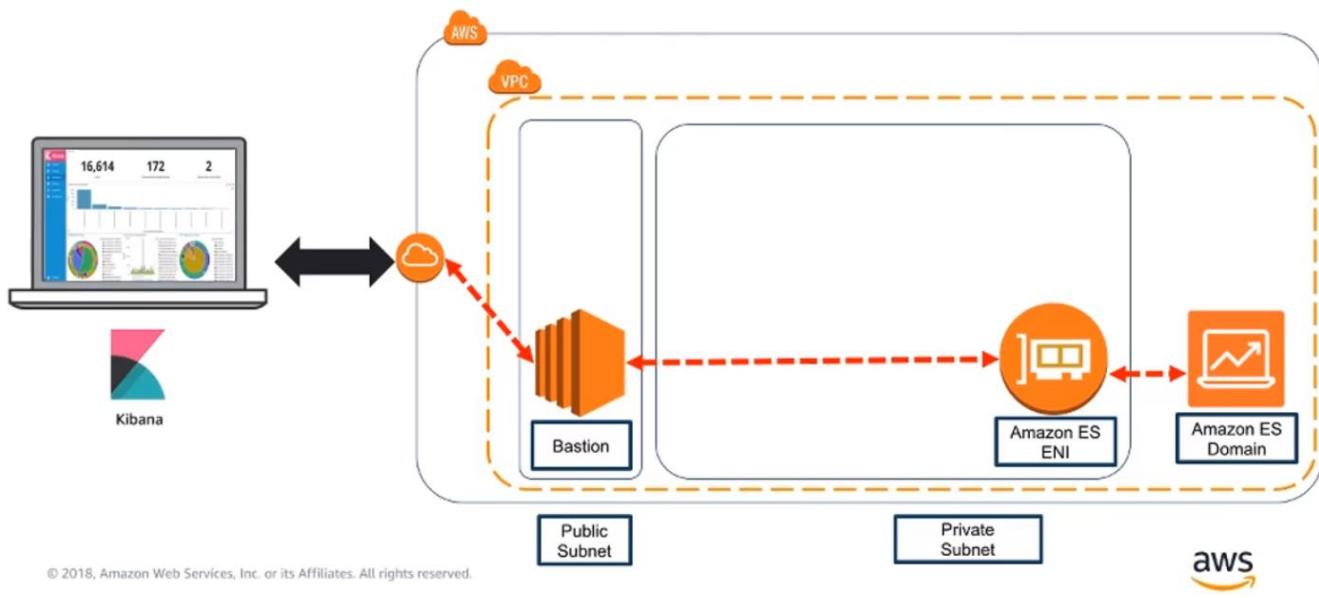
# Log analysis



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



## Architecture: Log analytics



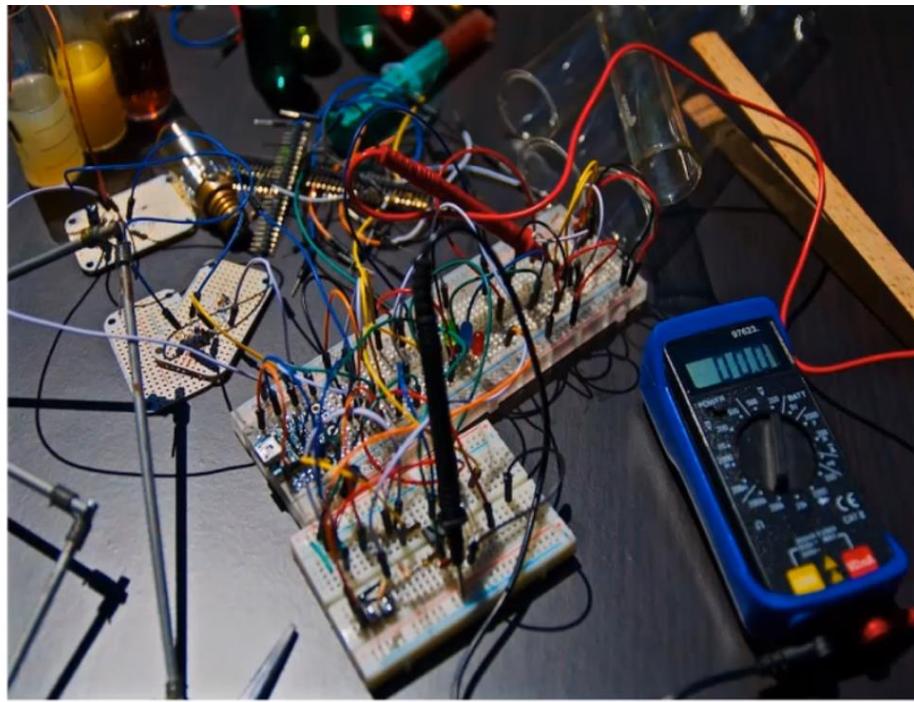
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



We will have Kibana talking to a public or bastion host within our public subnet of our VPC, the bastion then forwards the traffic to the ES service through the ENI and then provide the results back for visualization in Kibana

# Demo

## Log Analysis



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



Search 5000 IMDb titles

Search movies

star wars

Directors

Related actors

Actors

Genres

This is our search application running and we can search for movies

Screenshot of a web browser showing the search results for "iron man" on an Elasticsearch-powered IMDb clone. The UI includes sections for Directors, Related actors, and Actors, each with a list of names and counts. On the right, movie details for "Iron Man" and "Iron Man 2" are displayed, including titles, IDs, scores, ratings, and plot summaries.

We then get a list of results back that we can display on the UI as above

Screenshot of a code editor showing the PHP file "index.php". The code defines a function "format\_hit" that takes an Elasticsearch hit object and formats it into an HTML row structure. It extracts fields like title, ID, score, rating, and plot from the hit and constructs a responsive grid row with image placeholders and bolded text for key information.

```

FOLDERS
Lab - Search Movies
CFN Templates
imdb-for-es
2013imdb.txt
convert-imdb-data.py
imdb_mapping.json
jmeter
Old and unused
re-invent-cfn-templates
upload-to-s3
filebeat.yml
heartbeat.yml
index.php
jmeter.log
Kibana Instructions.docx
logstash.conf
Metricbeat Install Instructions
metricbeat.yml
virtualhosts.conf

index.php
index.php

1 <?php
2 require 'vendor/autoload.php';
3
4 use Elasticsearch\ClientBuilder;
5
6 function format_hit($hit) {
7     $parsed = parse_url($hit['_source']['image_url']);
8     $imgsrc = "imgs/".$basename($parsed['path']);
9
10    $result = '<div class="row"><div class="col-md-8"><hr></div></div>';
11
12    $result .= '<div class="row">';
13    $result .= '<div class="col-md-3">';
14    $result .= '';
15    $result .= '</div>';
16
17    $result .= '<div class="col-md-9">';
18    $result .= '<div class="row">';
19    $result .= '<div class="col-md-4"><span style="font-weight: bold">Title: </span>'.$hit["_source"]["title"].'</div>';
20    $result .= '<div class="col-md-4"><span style="font-weight: bold">ID: </span>'.$hit["_id"].'</div>';
21    $result .= '</div>';
22    $result .= '<div class="row">';
23    $result .= '<div class="col-md-4"></div>';
24    $result .= '<div class="col-md-3"><span style="font-style: italic">Score: </span>'.$hit["_score"].'</div>';
25    $result .= '</div>';
26    $result .= '<div class="row">';
27    $result .= '<div class="col-md-4"></div>';
28    $result .= '<div class="col-md-3"><span style="font-style: italic">Rating: </span>'.$hit["_source"]["rating"].'</div>';
29    $result .= '</div>';
30    $result .= '<div class="row">';
31    $result .= '<div class="col-md-8"><span style="font-weight: bold">Plot: </span>'.$hit["_source"]["plot"].'</div>';
32    $result .= '</div>';
33    $result .= '</div>';
34    $result = $result . "</div>&nbsp;&nbsp;";
35    return $result;
36 }

```

Here is the PHP code for developing the UI, we are using the Elasticsearch client from elastic.

**FOLDERS**

- Lab - Search Movies
  - CFN Templates
  - imdb-for-es
    - 2013imdb.txt
    - convert-imdb-data.py
    - imdb\_mapping.json
  - jmeter
  - Old and unused
  - re-invent-cfn-templates
  - upload-to-s3
  - filebeat.yml
  - heartbeat.yml
- index.php
- jmeter.log
- Kibana instructions.docx
- logstash.conf
- Metricbeat Install instructions
- metricbeat.yml
- virtualhosts.conf

```

index.php
35     return $result;
36 }
37
38 $query_json = '{
39     "query" : {
40         "simple_query_string" : {
41             "query" : "'.$_REQUEST["keyword"].'",
42             "fields" : ["title^3", "plot", "actors", "directors"],
43             "default_operator" : "AND"
44         }
45     },
46     "aggs" : {
47         "Genres" : {
48             "terms" : {
49                 "field" : "genres.keyword",
50                 "size" : 5
51             }
52         },
53         "Actors" : {
54             "terms" : {
55                 "field" : "actors.keyword",
56                 "size" : 5
57             }
58         },
59         "Directors" : {
60             "terms" : {
61                 "field" : "directors.keyword",
62                 "size" : 5
63             }
64         },
65         "Related actors" : {
66             "significant_terms" : {
67                 "field" : "actors.keyword"
68             }
69         }
70     }
71 }';
72
73 $params = [
74     'index' => 'movies',
75     'type' => 'movie',
76 ];

```

This is the simple querystring that we are using for the search button functionality, it takes in a keyword from the user via the input element on the UI. We are also doing aggregations on some other fields like Genres, Actors, Directors, and Related actors.

**FOLDERS**

- Lab - Search Movies
  - CFN Templates
  - imdb-for-es
    - 2013imdb.txt
    - convert-imdb-data.py
    - imdb\_mapping.json
  - jmeter
  - Old and unused
  - re-invent-cfn-templates
  - upload-to-s3
  - filebeat.yml
  - heartbeat.yml
- index.php
- jmeter.log
- Kibana instructions.docx
- logstash.conf
- Metricbeat Install instructions
- metricbeat.yml
- virtualhosts.conf

```

index.php
55     "field" : "actors.keyword",
56     "size" : 5
57 }
58 },
59 "Directors" : {
60     "terms" : {
61         "field" : "directors.keyword",
62         "size" : 5
63     }
64 },
65 "Related actors" : {
66     "significant_terms" : {
67         "field" : "actors.keyword"
68     }
69 }
70 };
71 }
72
73 $params = [
74     'index' => 'movies',
75     'type' => 'movie',
76     'body' => $query_json
77 ];
78
79 $hosts = ['search-bootcamp1-5s5jafofxnm4kf64ihl75era4m.us-east-1.es.amazonaws.com:80'];
80 $client = ClientBuilder::create()->setHosts($hosts)->build();
81 $results = $client->search($params);
82 ?>
83
84 <!DOCTYPE html>
85 <html>
86 <head>
87     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
88     <title>Simple Search Page</title>
89     <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/
bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg320mUcw7on3RYdg4Va+PmSTsz
/K68vbdEjh4u" crossorigin="anonymous">
90 </head>
91 <body style='margin: 20px;'>
92     <div id="wrapper">
93         <h2>Search 2000 IMDB Titles</h2>
94     </div>
95 </body>
96 </html>

```

**FOLDERS**

- Lab - Search Movies
- CFN Templates
- imdb-for-es
  - 2013imdb.txt
  - convert-imdb-data.py
  - imdb\_mapping.json
- jmeter
- Old and unused
- re-invent-cfn-templates
- upload-to-s3
- filebeat.yml
- heartbeat.yml

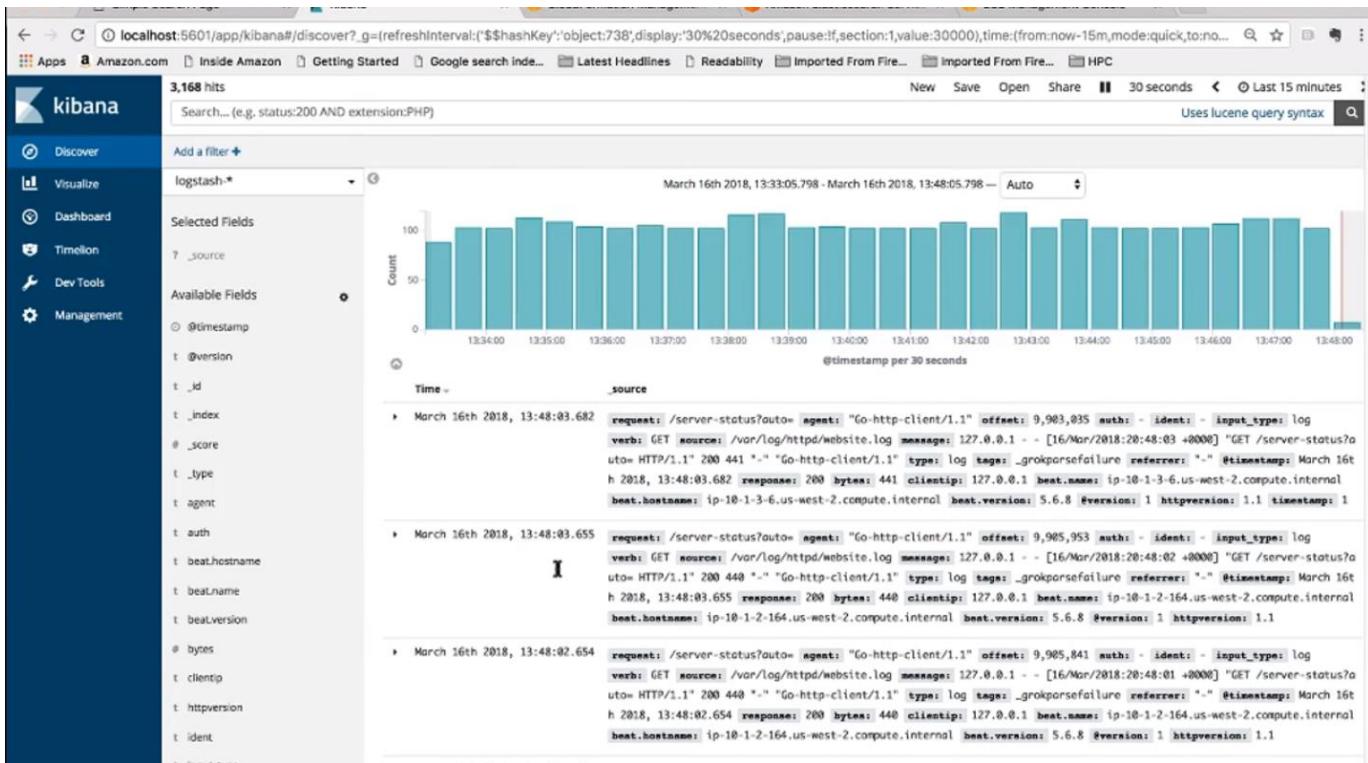
**index.php**

```

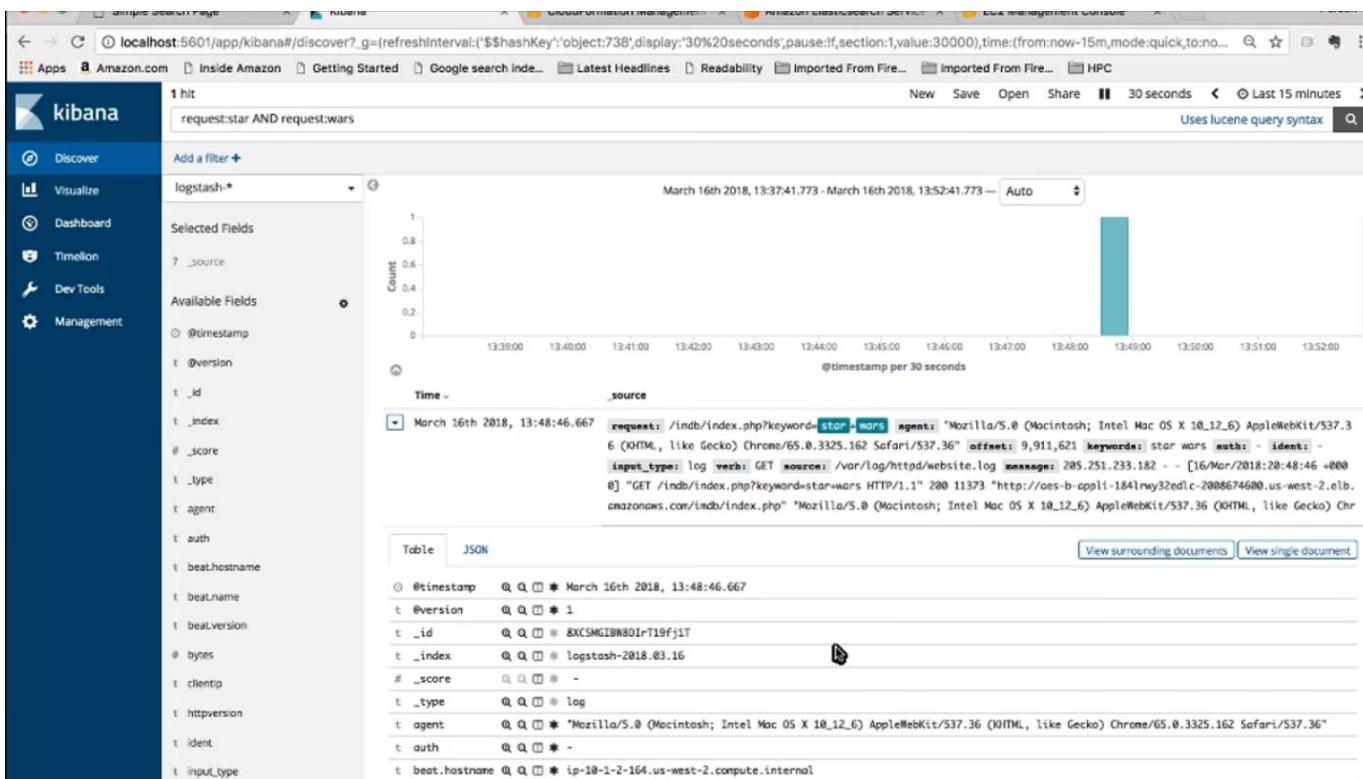
88      <title>Simple Search Page</title>
89      <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/
bootstrap.min.css" integrity="sha384-BVYiiSIFeKIdGmRAkyuHAHrg320mUcw7on3RYdg4Va+PmSTsz
/K68vbdEjh4u" crossorigin="anonymous">
90  </head>
91  <body style="margin: 20px;">
92    <div id="wrapper">
93      <h1>Search 5000 IMDb titles</h1>
94      <div id="search_section" style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px">
95        <form style="display:inline" action="index.php">
96          <label for="Search field" style="font-size: 16pt">Search movies</label>
97          <input style="font-size: 16pt" name="keyword" type="text" id="author" value="?php echo $_REQUEST['keyword'] ??" size="30" maxlength="100" class="search_field" />
98          <input type="submit" value="Search"/>
99        </form>
100      </div>
101    </div>
102    <div class="container-fluid">
103      <div class="row">
104        <div class="col-md-3">
105          <?php
106            foreach ($results['aggregations'] as $aggName => $agg) {
107              echo '<div class="row"><h2>' . $aggName . '</h2></div>';
108              foreach ($agg['buckets'] as $bucket) {
109                echo '<div class="row">';
110                echo '<div class="col-md-6">' . $bucket['key'] . '</div>';
111                echo '<div class="col-md-6">' . $bucket['doc_count'] . '</div>';
112                echo '</div>';
113              }
114            }
115          ?>
116        </div>
117        <div class="col-md-9">
118          <?php
119            foreach ($results["hits"]["hits"] as $hit) {
120              echo format_hit($hit);
121            }
122          ?>
123        </div>
124      </div>

```

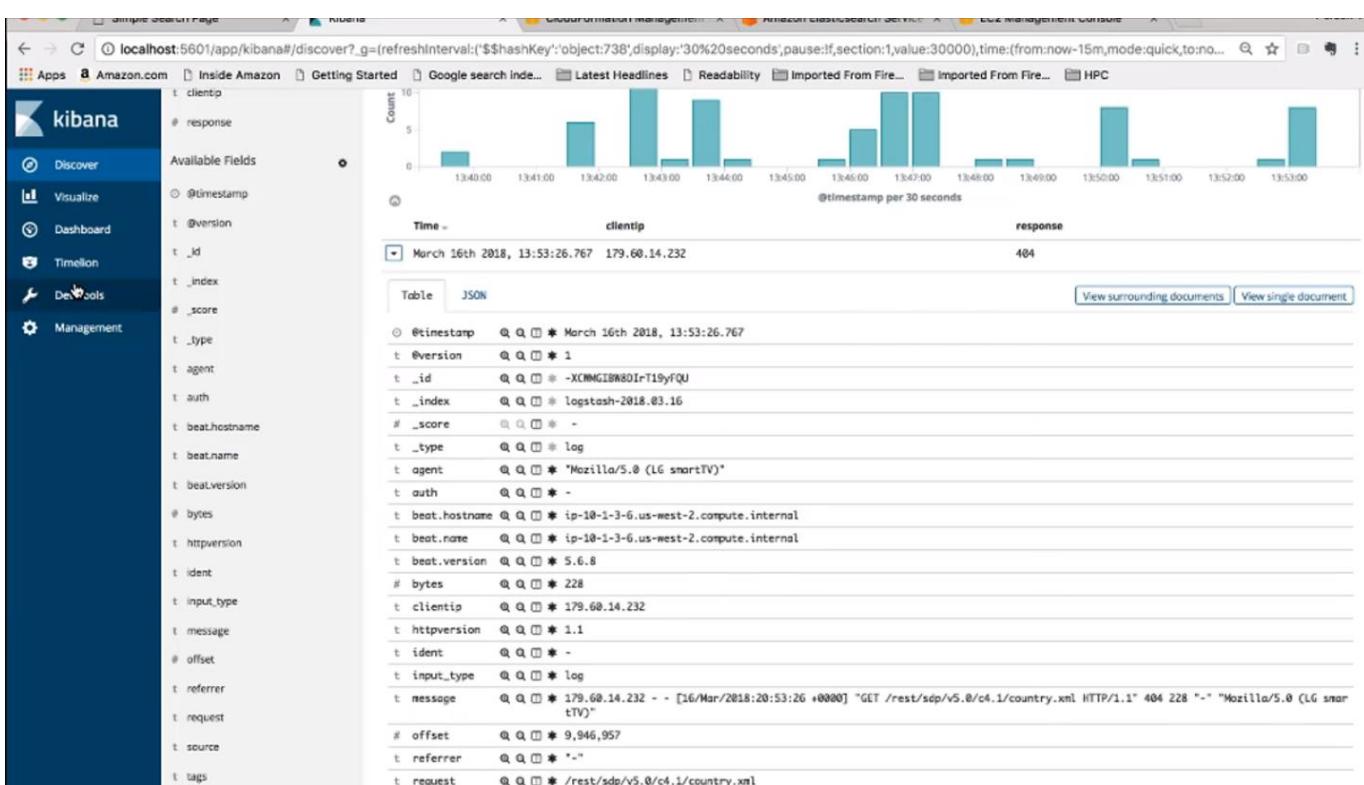
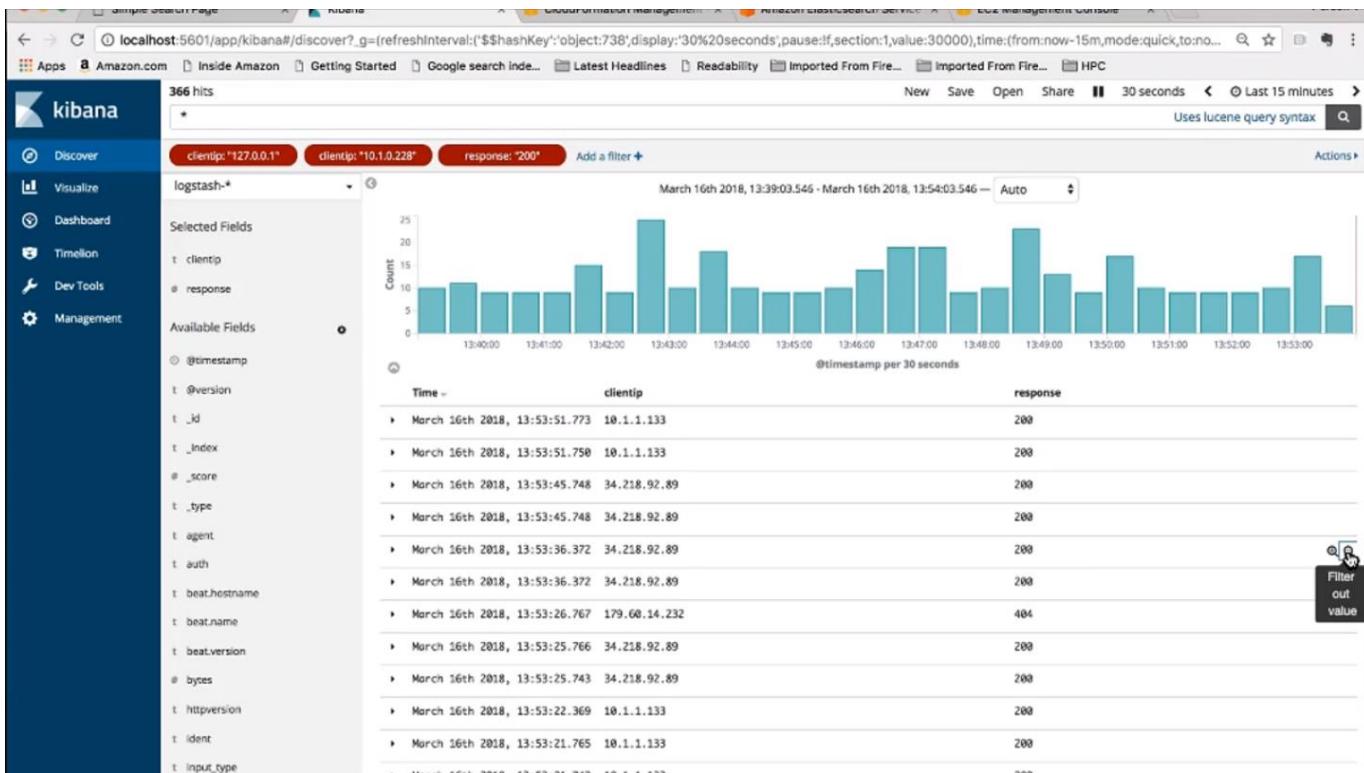
We then have the main body with the form in it as above. We already have Filebeat running on our HTTP servers that help to ship out the logs as they get created. We can then visualize the logs in Kibana

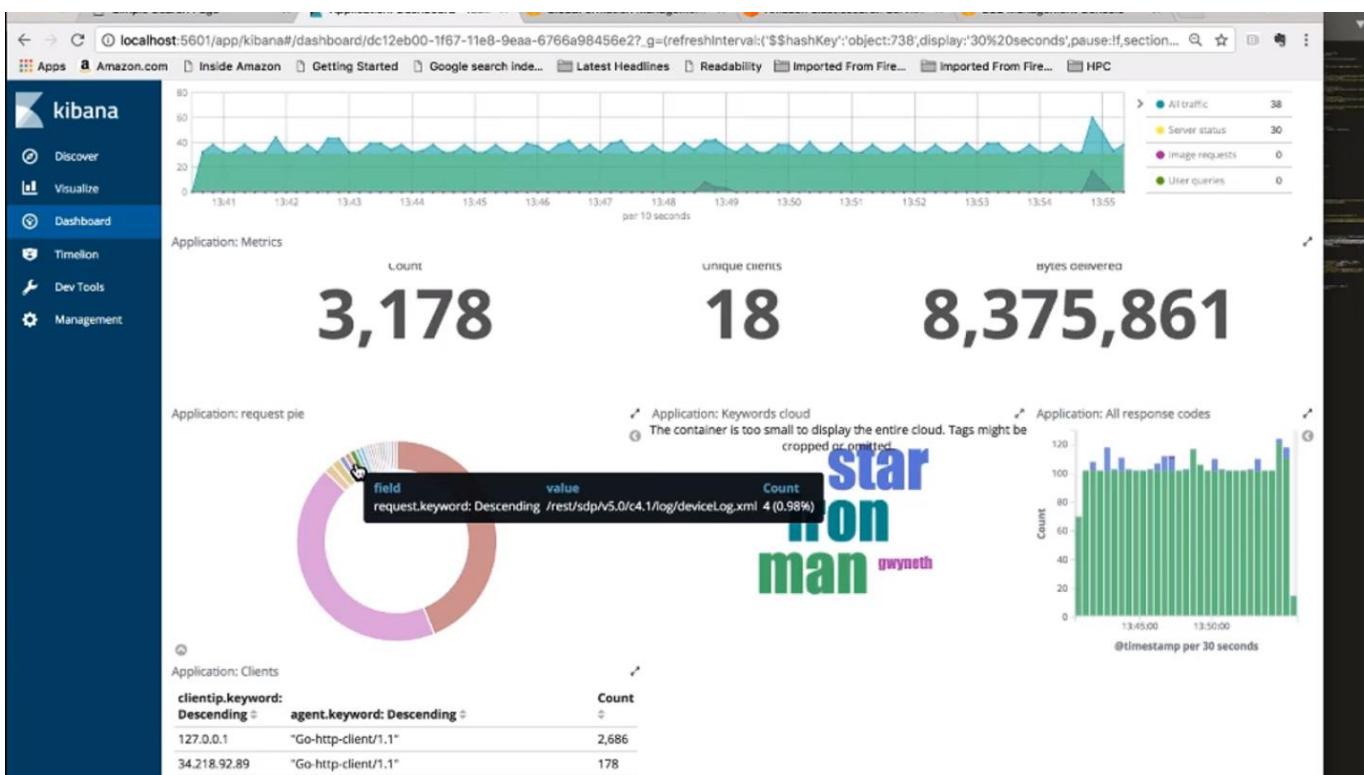
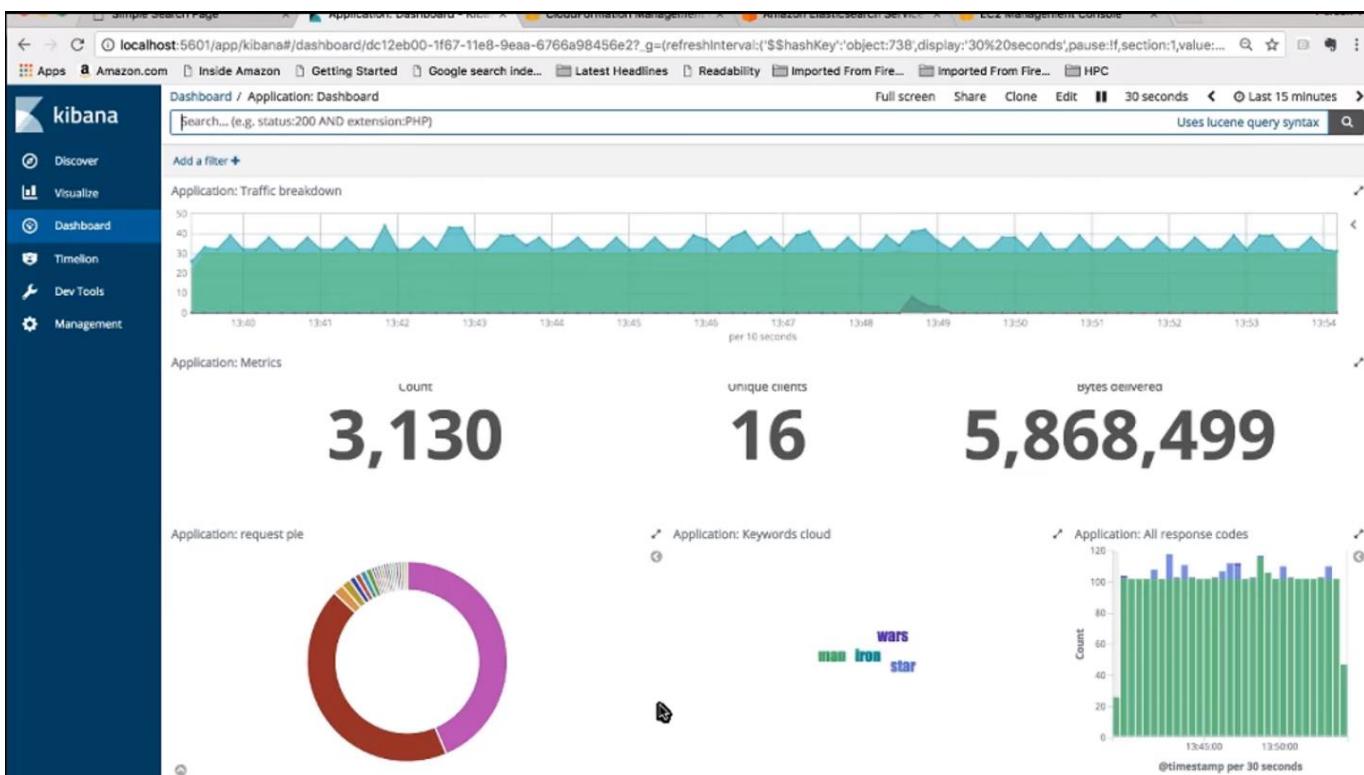


This is our main Kibana dashboard with all the outputs from Logstash









## CASE STUDY: EXPEDIA

### Application Monitoring & Root-cause Analysis

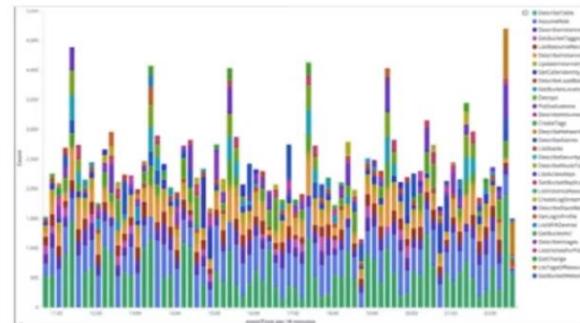


#### PROBLEM

Logs, lots and lots of logs. How to cost effectively monitor logs?

Require centralized logging infrastructure

Did not have the man power to manage infrastructure



#### SOLUTION

Streaming AWS CloudTrail logs, application logs, and Docker startup logs to Elasticsearch

Created centralized logging service for all team members

Using Kibana for visualizations and for Elasticsearch queries

#### BENEFITS

**Quick insights:** Able to identify and troubleshoot issues in real-time

**Secure:** Integrated w/ AWS IAM

**Scalable:** Cluster sizes are able to grow to accommodate additional log sources

© 2016 Amazon Web Services, Inc. or its affiliates. All rights reserved.



# Loose Ends





Key Idea

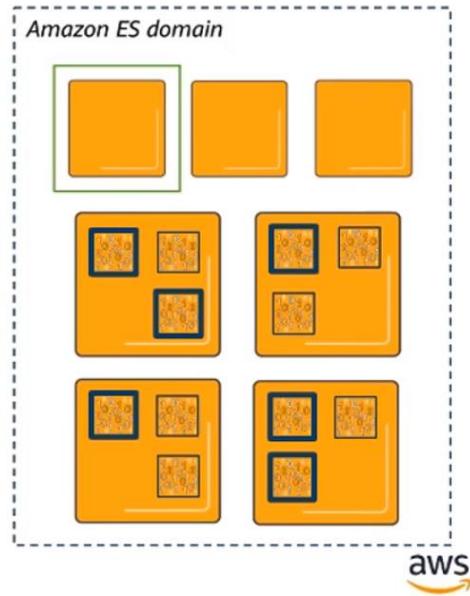
## Use 3 dedicated master instances in production

Enable dedicated master

Dedicated master instance type: m4.large.elasticsearch (default)

Dedicated master instance count: 3 (default)

Master instances orchestrate and make your cluster more stable



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Every ES cluster has a master node, AWS provides a feature that allows you to deploy a dedicated master node instances in production to orchestrate the cluster, keep track of the number of instances within the cluster, the indexes, shards, and schema for all the instances. That master function resides on a data node within the cluster, if your data node becomes overloaded then your master function gets degraded and your ES cluster can be lost.

## Dedicated master node recommendations

Number of data nodes	Master node instance type
< 10	m3.medium+
< 20	m4.large+
<= 50	c4.xlarge+
50-100	c4.2xlarge+

- In production, use 3 dedicated masters
- Use smaller instances than data instances
- Size based on instance count, index count, and schema size

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.





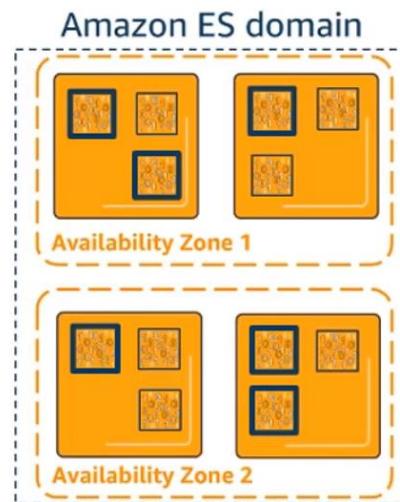
Key Idea

## Use zone awareness in production

Enable zone awareness



100% data redundancy in 2 zones makes your cluster more highly available



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



This checkbox provides HA feature to your ES cluster

## Set CloudWatch alarms for at least these metrics

Name	Metric	Threshold	Periods
ClusterStatus.red	Maximum	$\geq 1$	1
ClusterIndexWritesBlocked	Maximum	$\geq 1$	1
CPUUtilization/MasterCPUUtilization	Average	$\geq 80\%$	3
JVMMemoryPressure/Master...	Maximum	$\geq 80\%$	3
FreeStorageSpace	Minimum	$\leq (25\% \text{ of avail space})$	1
AutomatedSnapshotFailure	Maximum	$\geq 1$	1

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



## Wrap up

- Whether you're monitoring your infrastructure, or searching your application data, Amazon Elasticsearch Service makes it easy to locate the information you need
- The service offers ease-of-use, scalability, security, high availability, and compatibility with existing open source software
- With its tight integration with other AWS services you can get your data in and start working

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



# Thank you!

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

