Amazon Elasticsearch Service

Deep Dive

Jon Handler – Principal Solutions Architect, Search Services

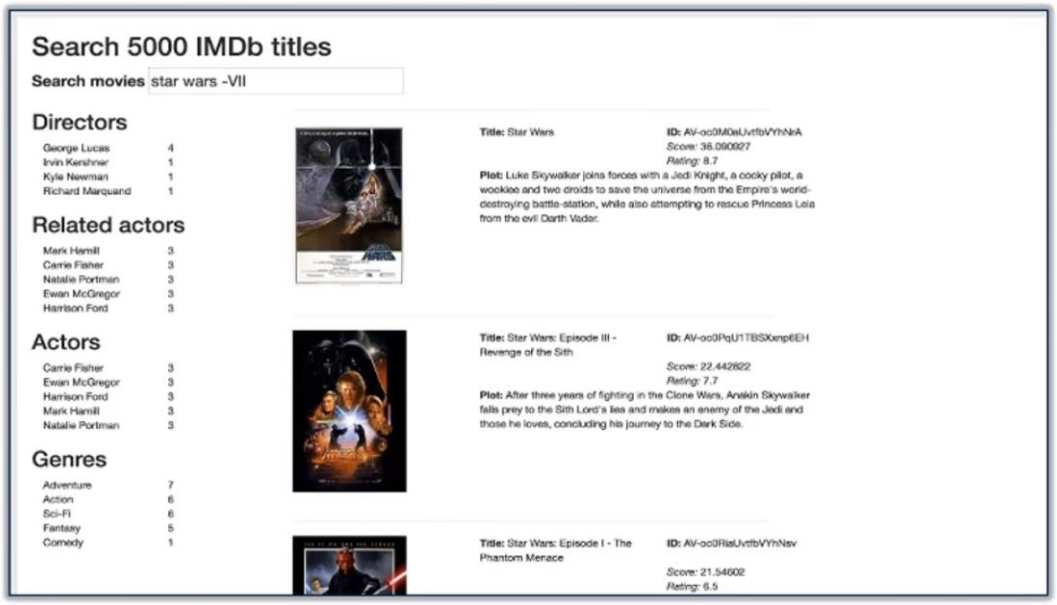
March 21, 2018

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

aws

"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



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

aws

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

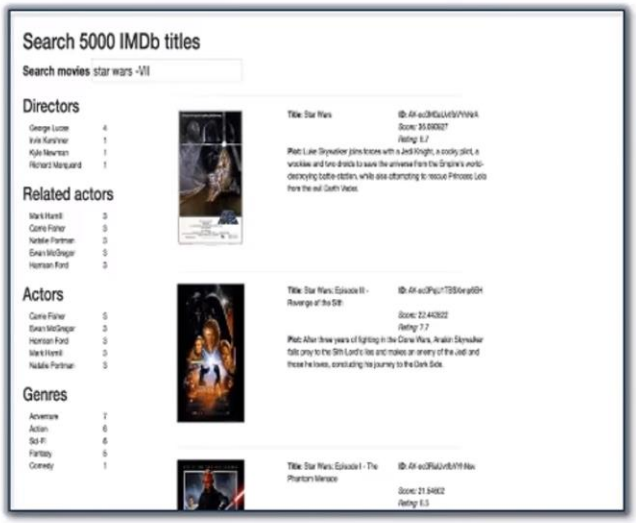
Rank	Project Name	Overall Project Rating
1	Linux	100.00
2	Git	31.10
3	MySQL	25.23
4	Node.js	22.75
5	Docker	22.61
6	Hadoop	16.19
7	**Elasticsearch**	**15.72**
8	Spark	14.99
9	MongoDB	14.68
10	Selenium	12.81
11	NPM	12.31
12	Redis	11.61

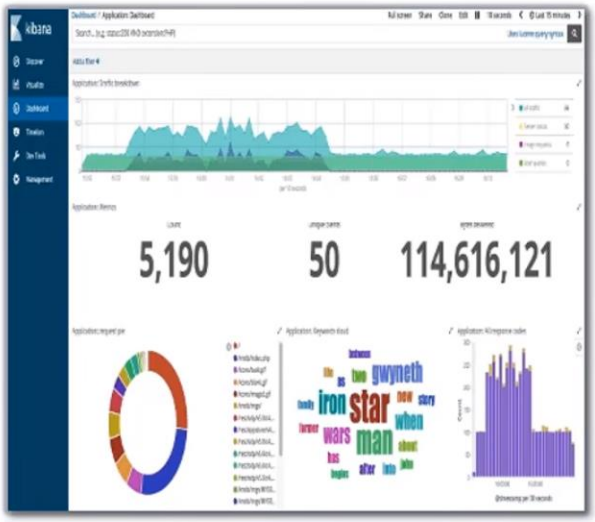
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

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

aws

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



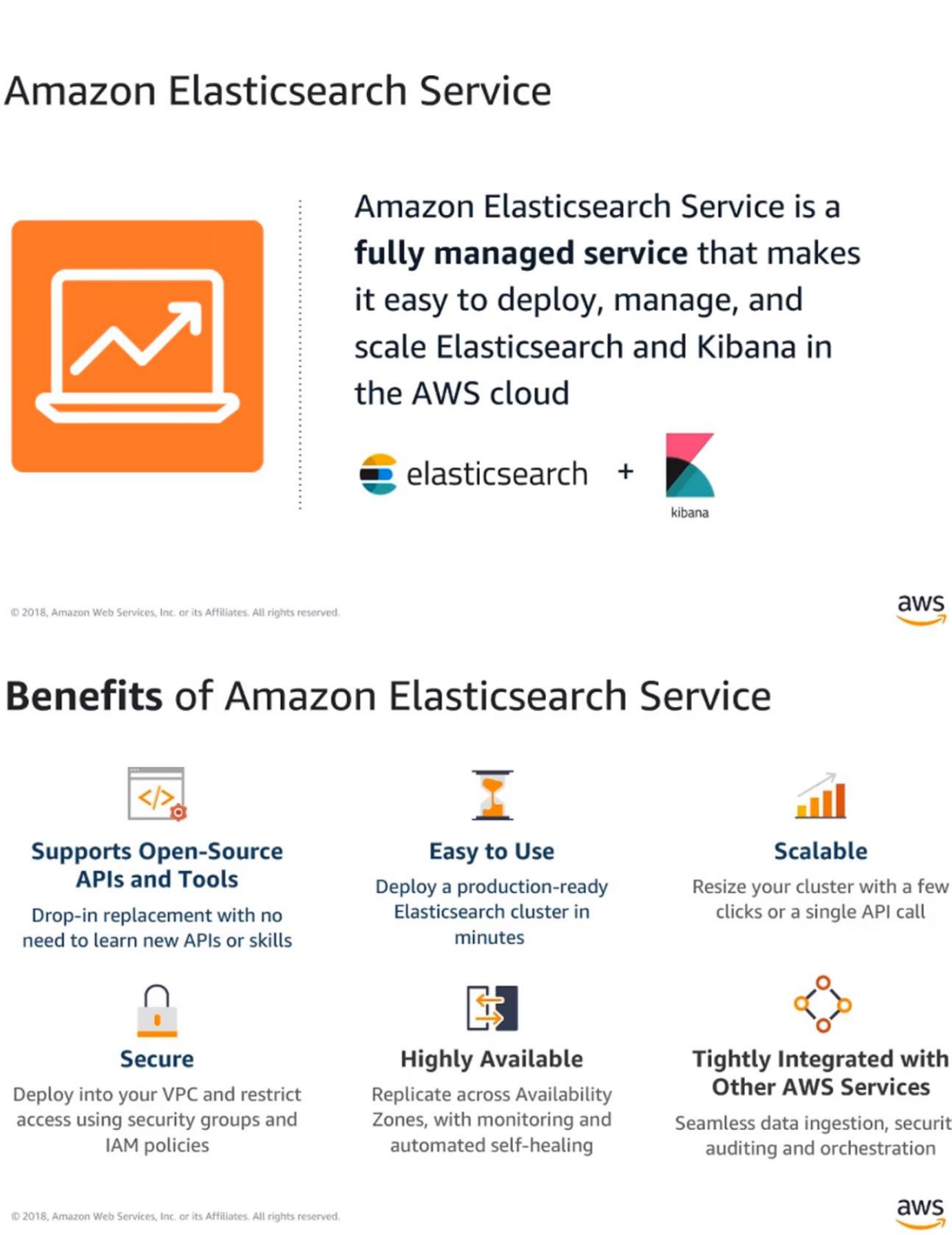


With Elasticsearch, you can run your business AND monitor it as well

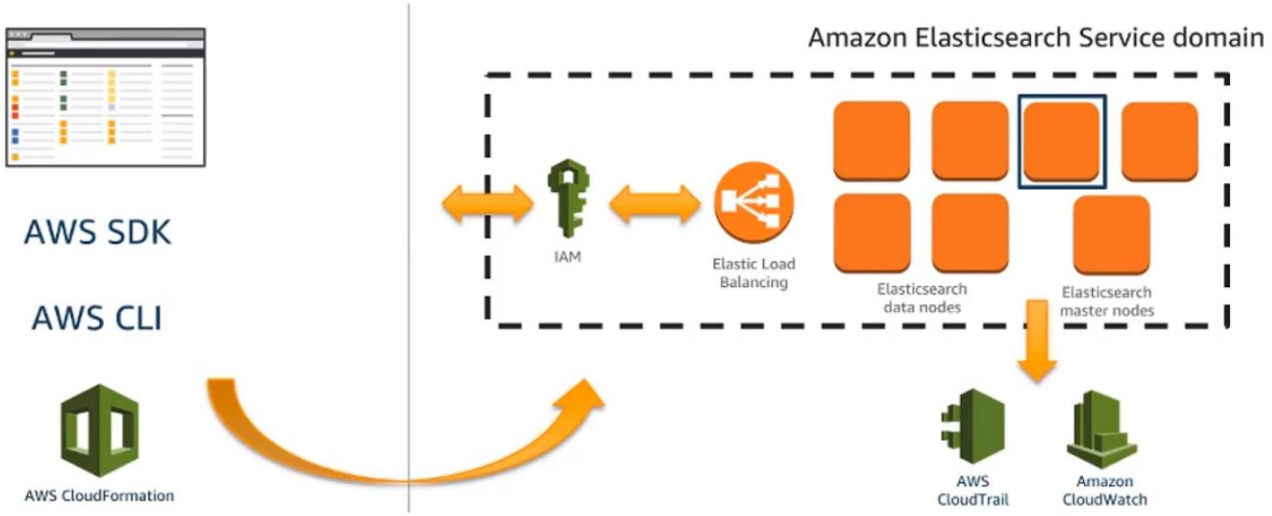
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

aws

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



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.



Key Idea

Service Architecture

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

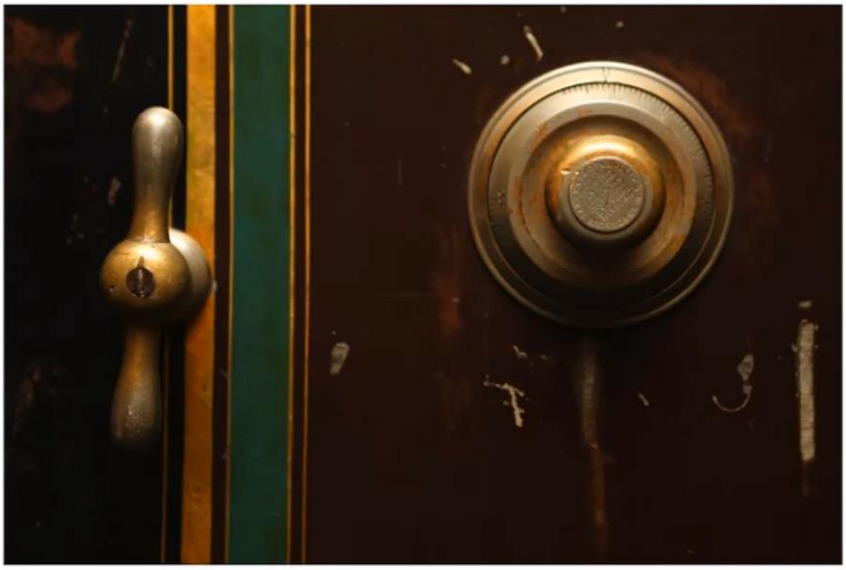
aws

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.

aws

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/*"
        }
    ]
}
```

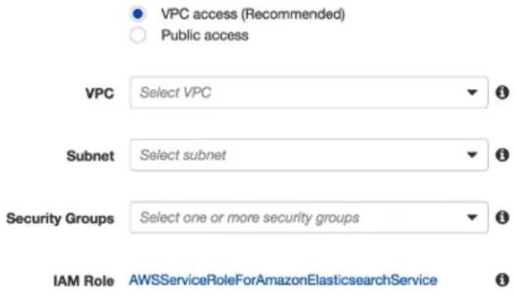
- To grant access for Kibana, use a CIDR **Condition**
- To grant read-only access to an account or role, specify an es:HttpGet **Action**
- To limit a user to a specific index, or API add it to the **Resource**
- For Admin access, specify administrative es:* **Actions**

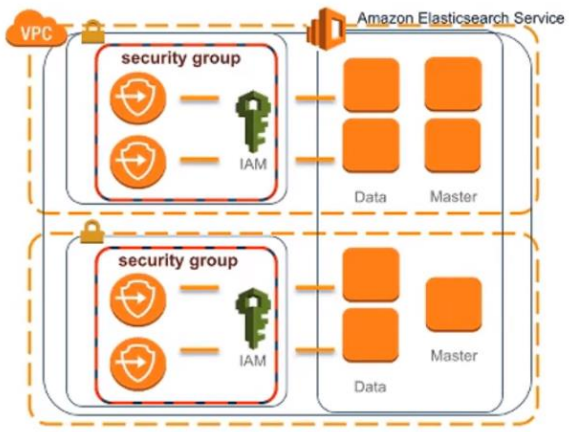
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

aws

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





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

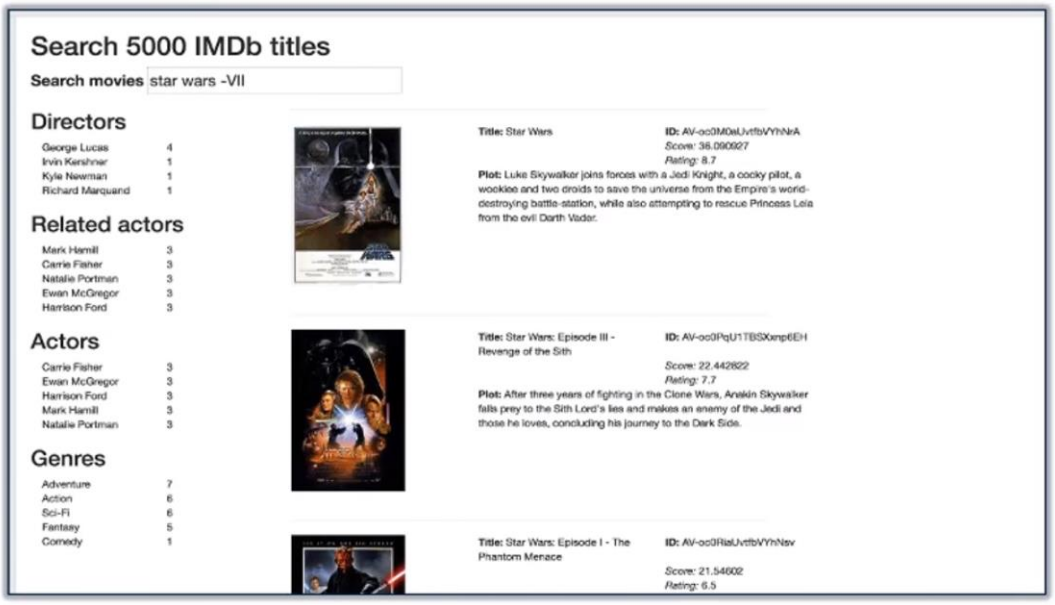
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

aws

Encrypt your data

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

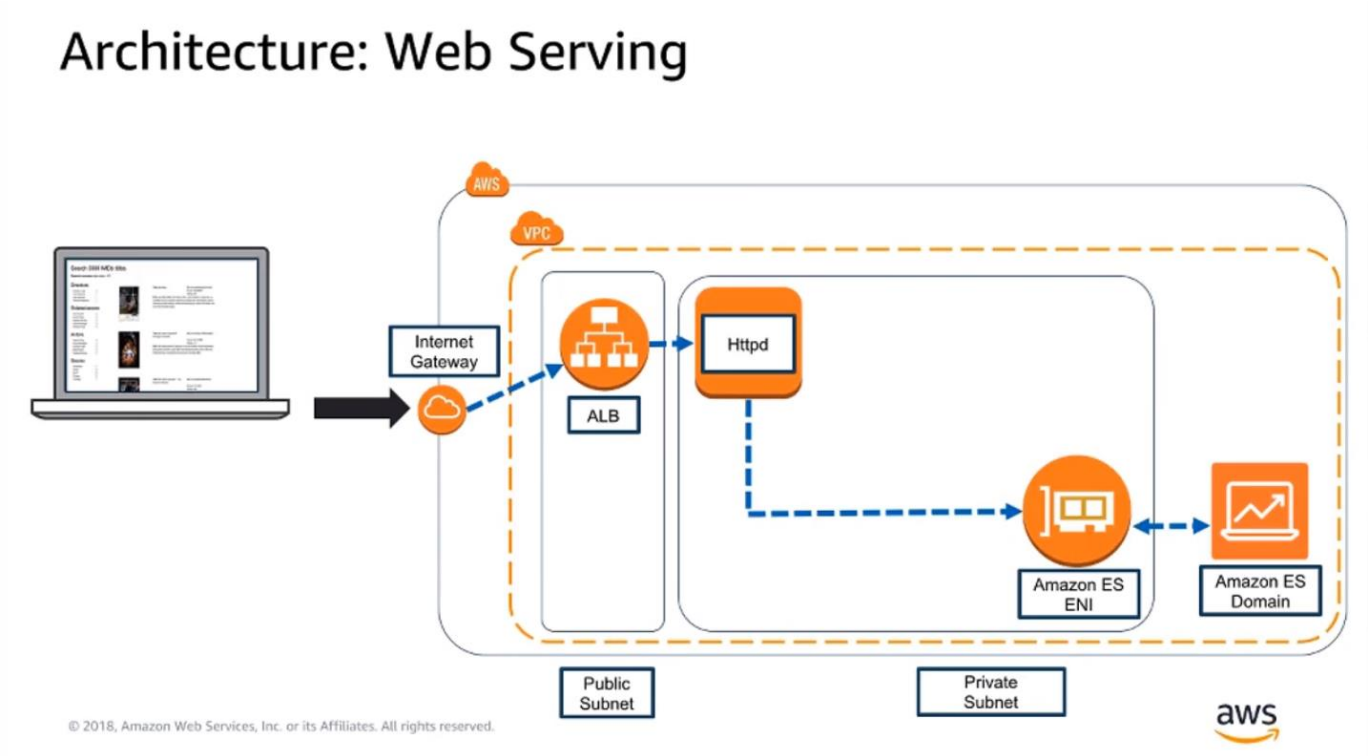
The front end



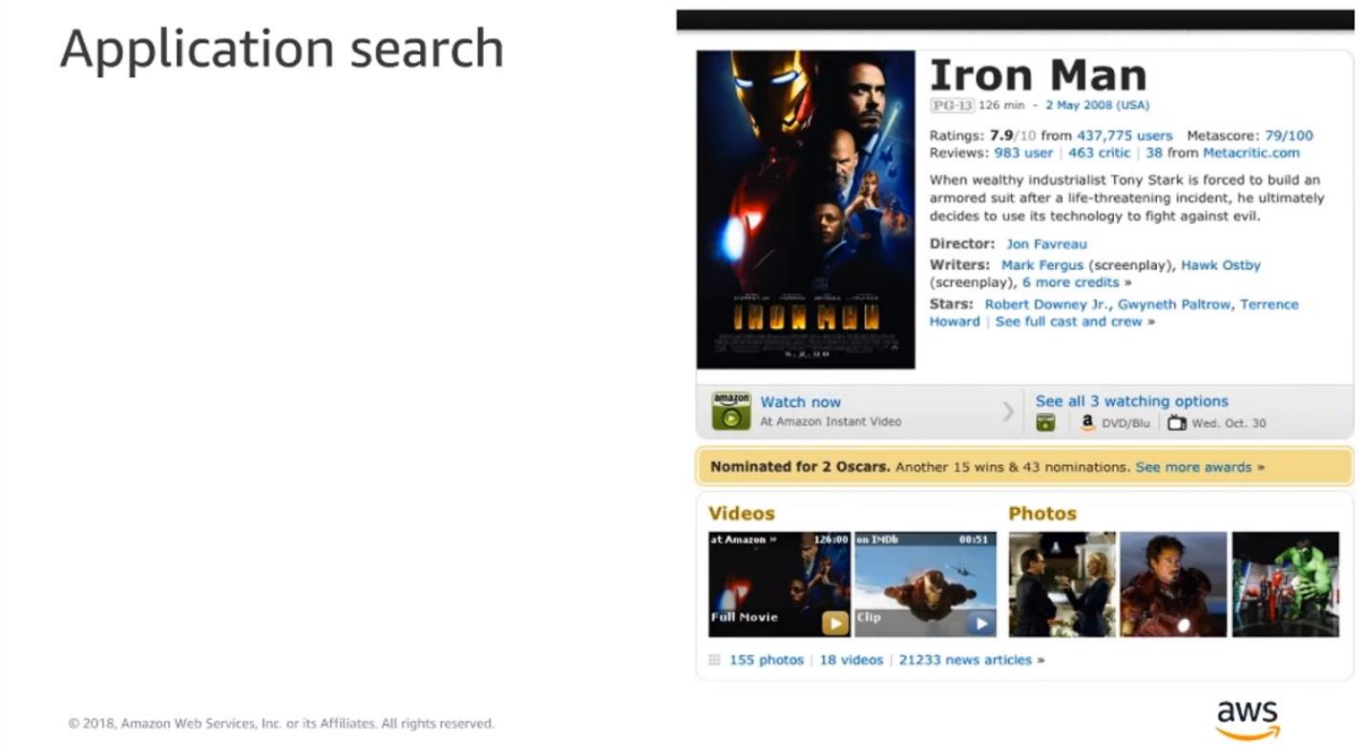
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

aws

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



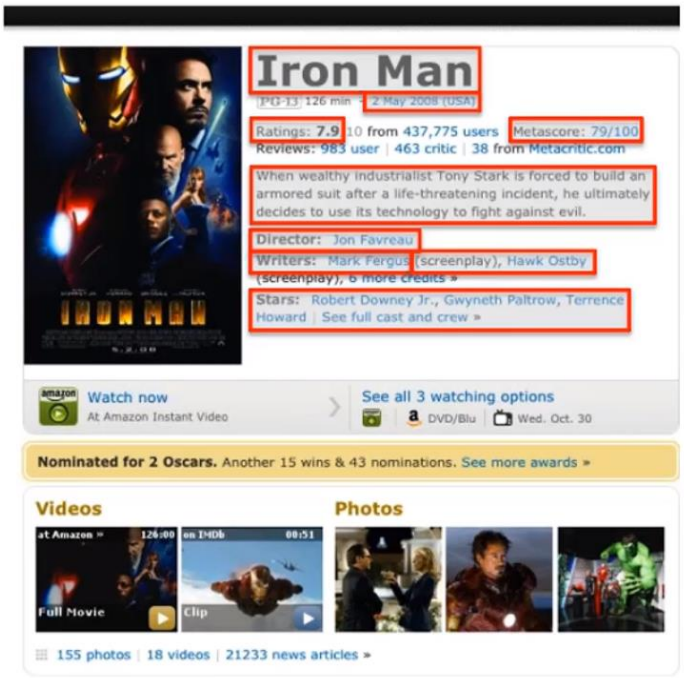
The web app takes a couple of search key words from the customer's client, formulates a query for ES, sends the query to the Elastic Network Interface ENI, this gets forwarded to the ES cluster running in the ES Domain that replies with the search results, the web app then formats the search results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



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.

aws

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 filed level, you start off with the single entity that you want to retrieve and the specify the set of fields that you want to comprise the values you want to get.

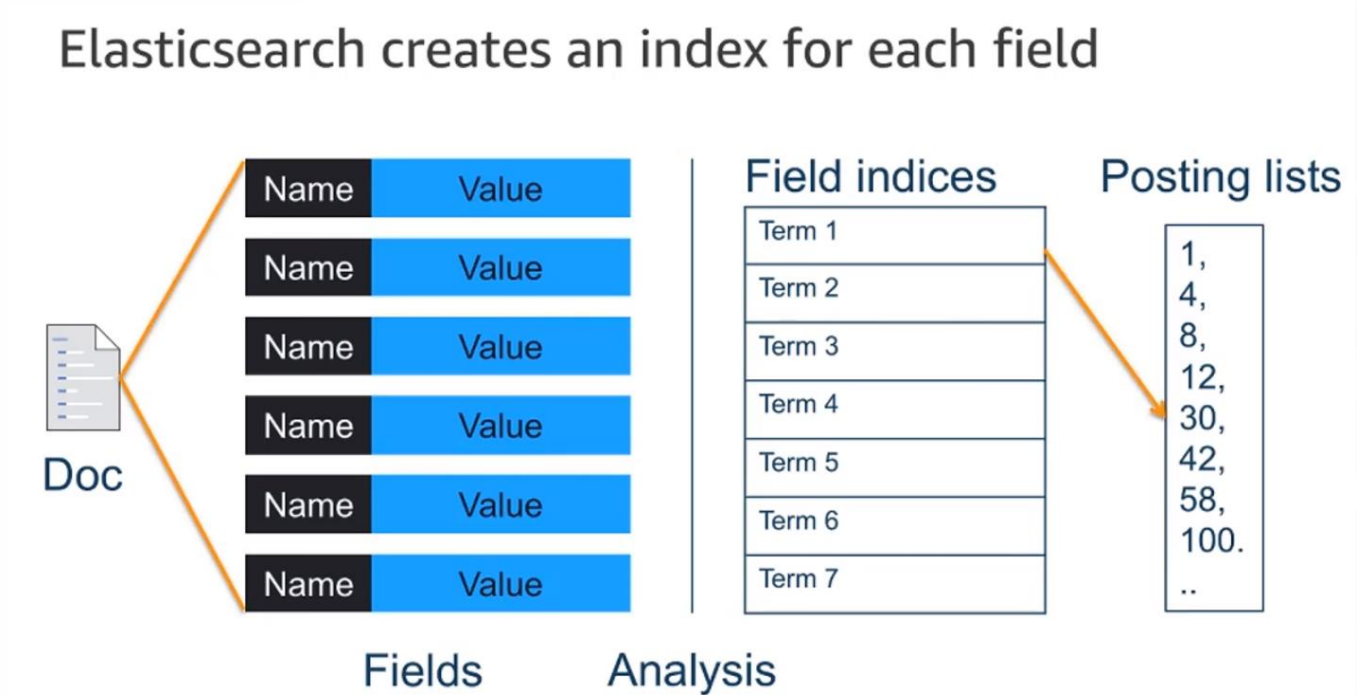
Elasticsearch works with structured JSON containing fields and values

```
{   "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"
    ]
}
```

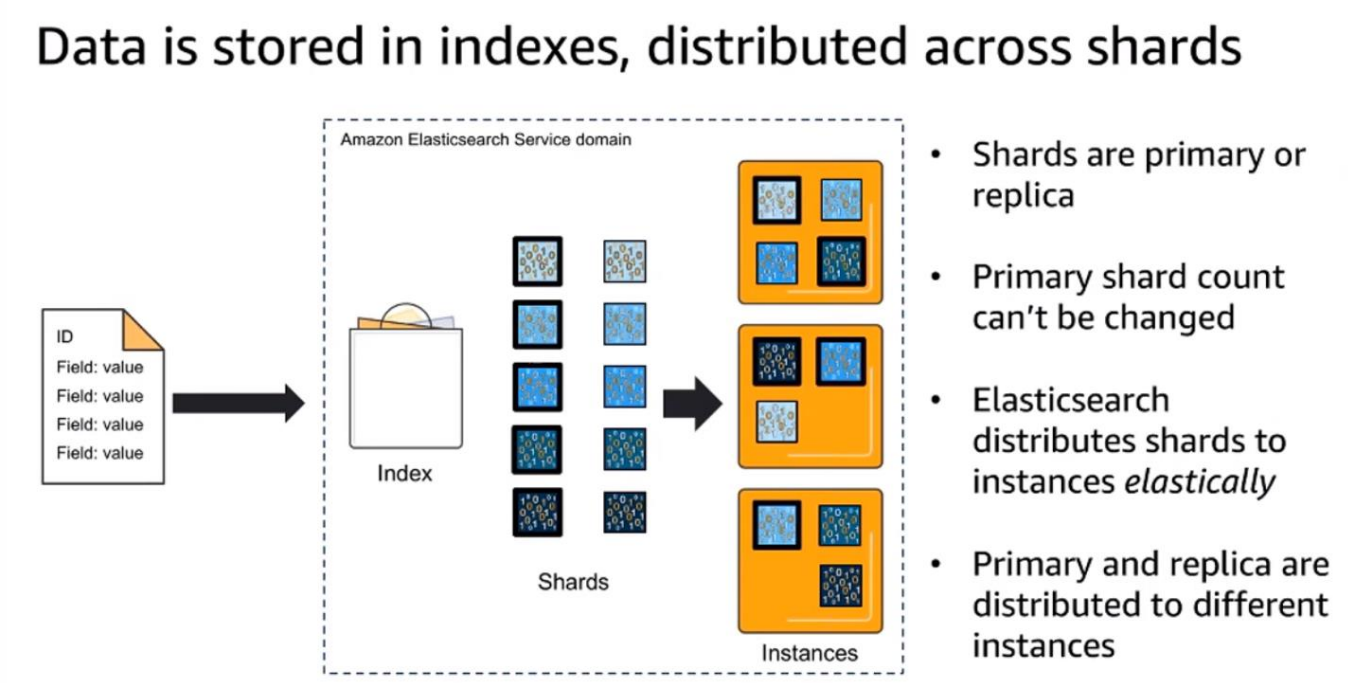
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

aws

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.

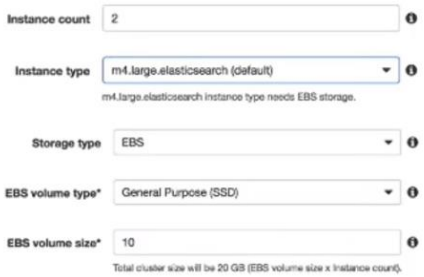


At the field level, ES creates an index. We can picture our document as a set of fields with names and values. ES analyzes the values of each field in the document and then creates an index for that field. The index contains everything that are in that field for all the documents that have come into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



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?



- 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.

aws

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

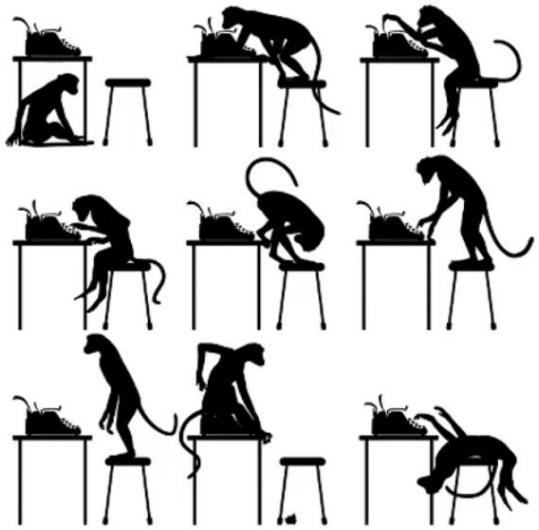
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

aws

Key Idea

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 (2TB / 40GB = 50 shards)
- Always use at least 1 replica in production
- Set shard count so that shards per instance ~= number of CPUs
- Keep shard sizes as equivalent as possible



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

aws

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
    }
}
```

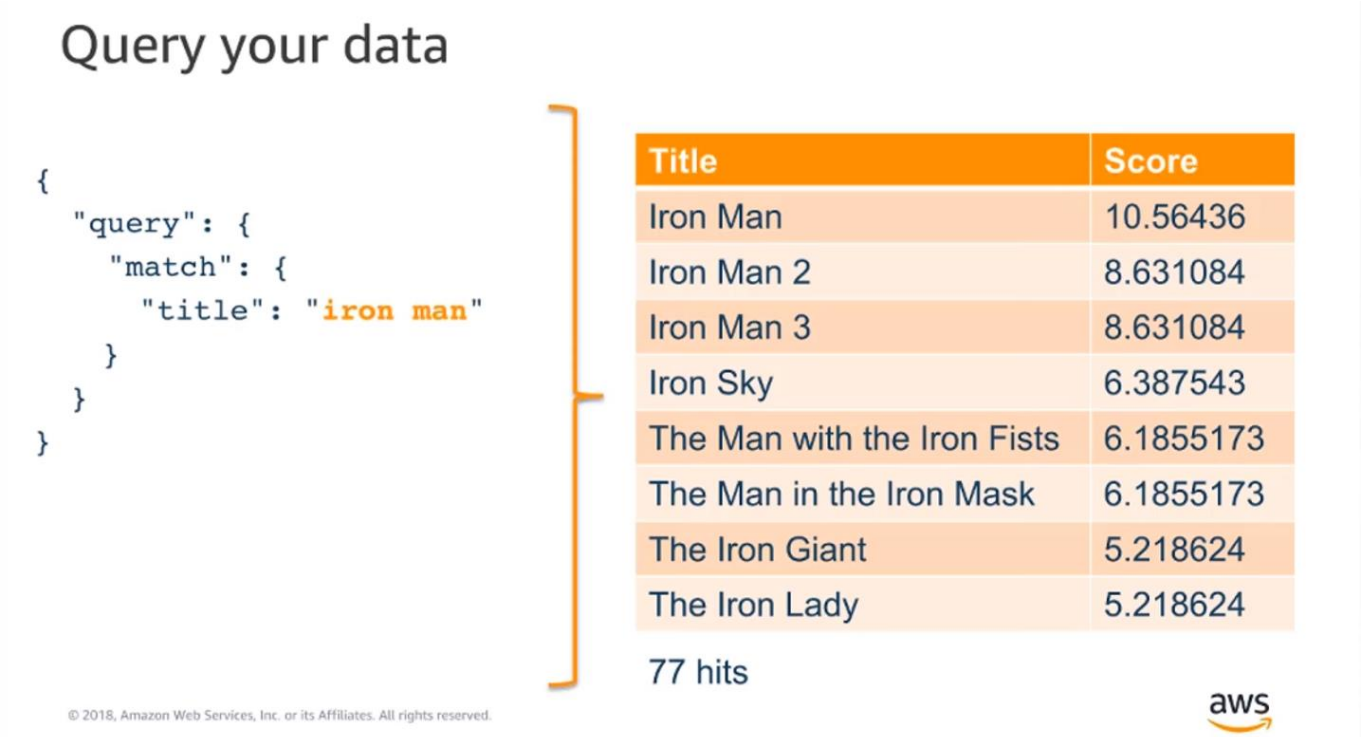
- All new indexes which match the index pattern receive the settings
- You can also specify a mapping for all indexes

*Note: ES 6.0+ syntax

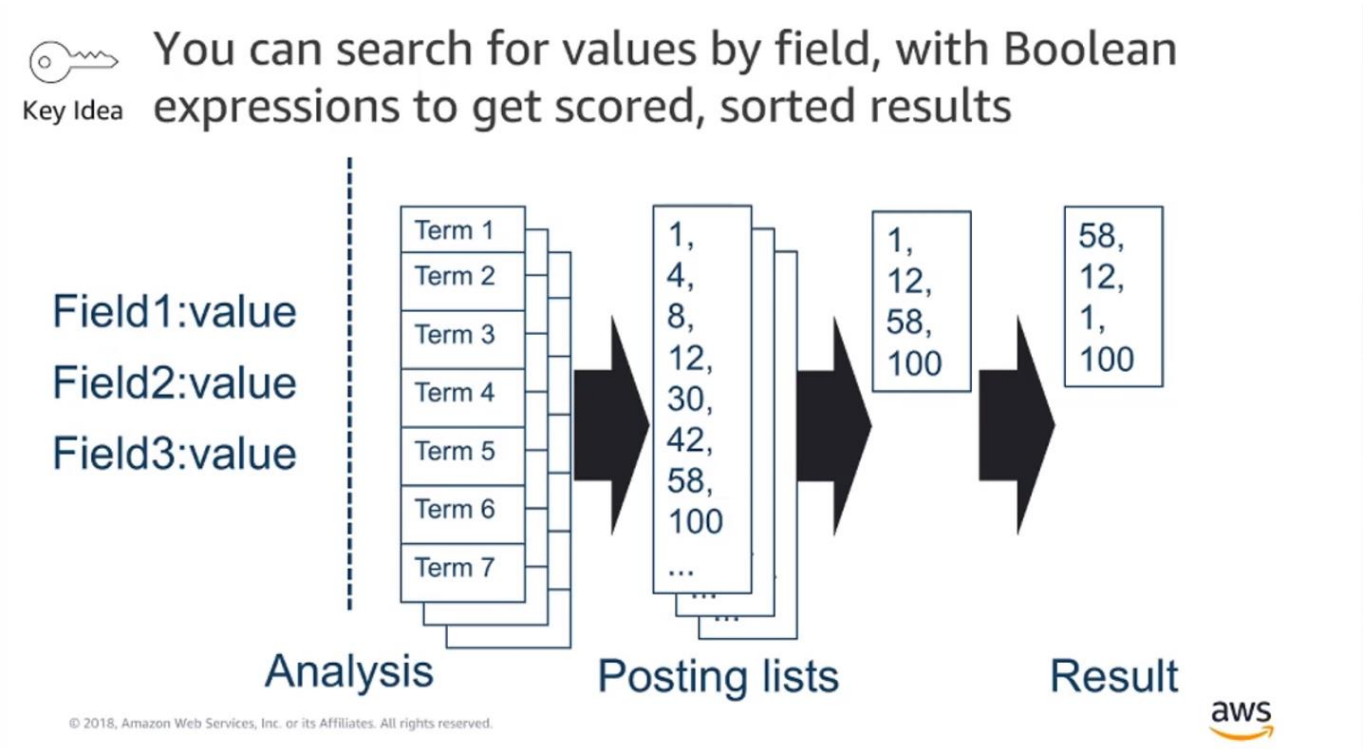
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

aws

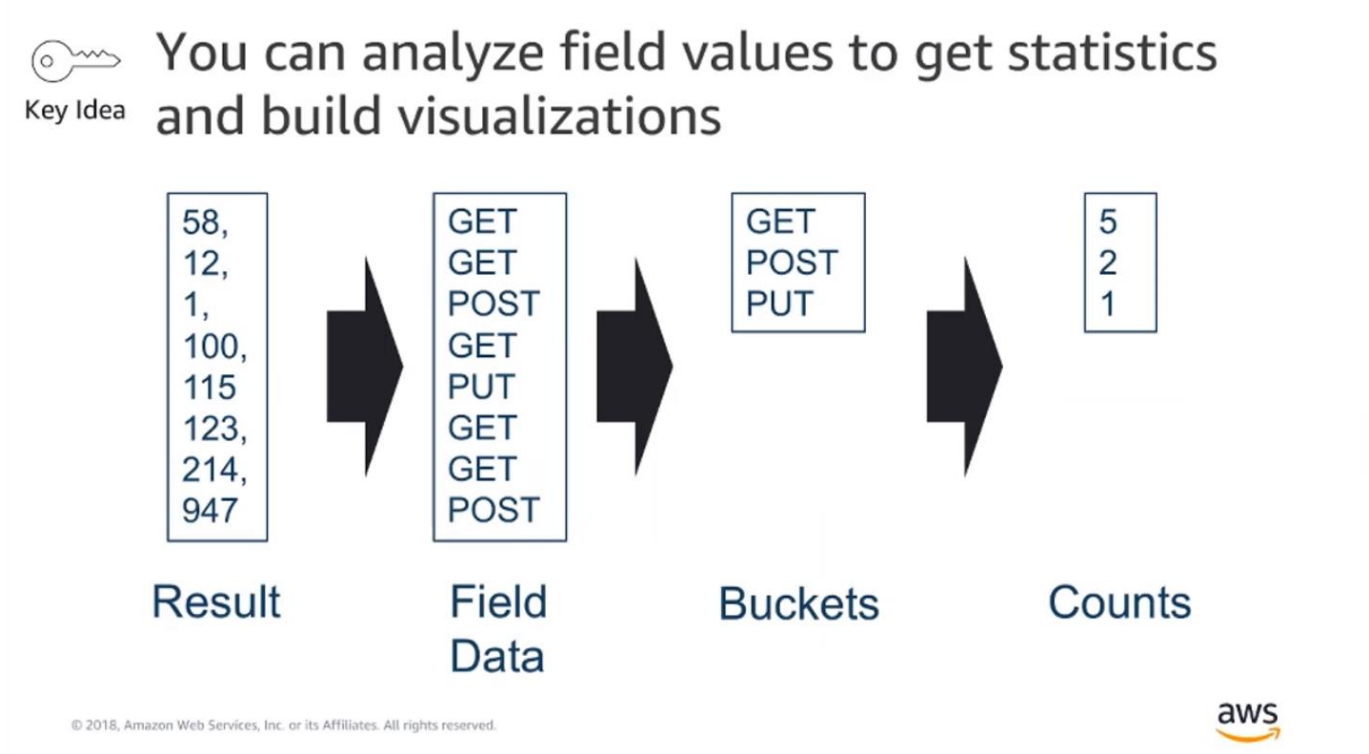
You can use a template that is an ES API as a way of specifying the shard count automatically (also relevant for log analytics use case), a replica will then create an additional 50 shards. We can also put mapping information in the API call for the schema.



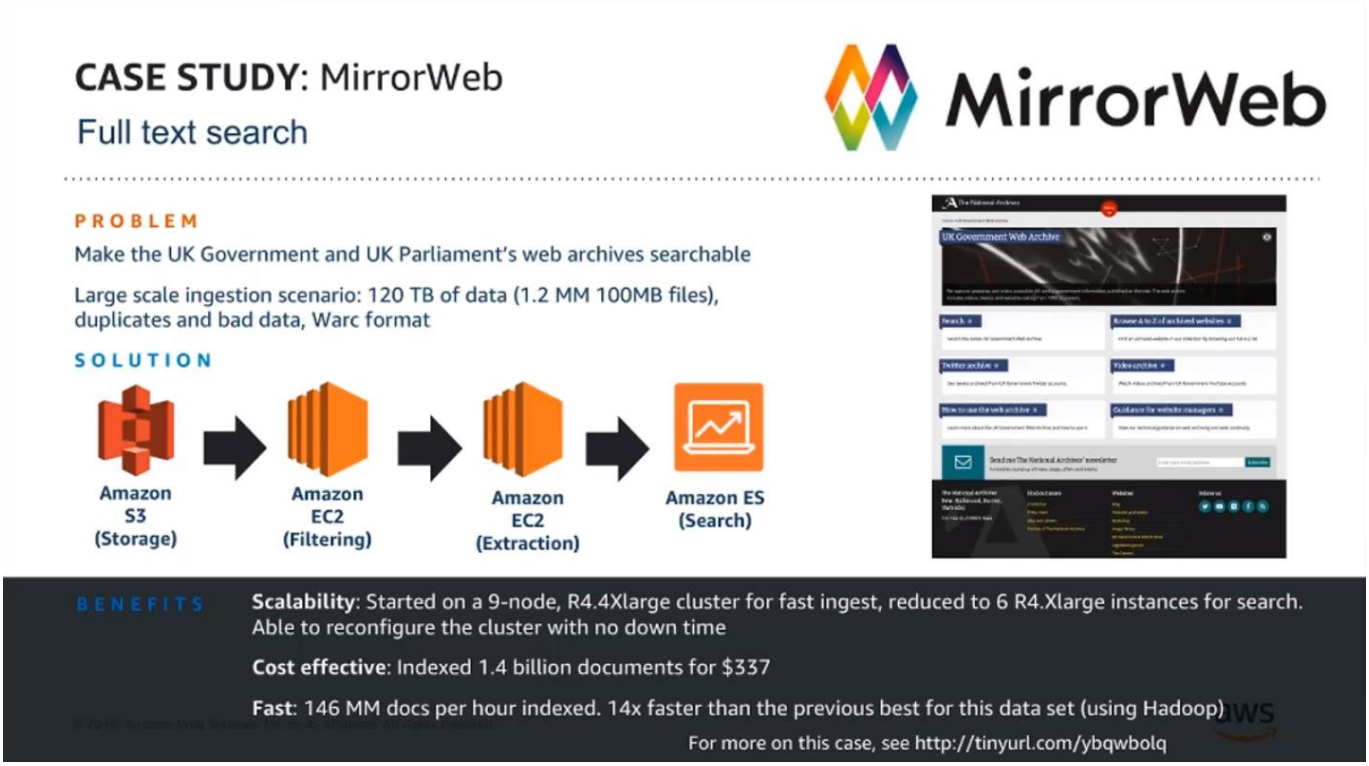
This is a sample of the Query API in ES, it is JSON as above. Note that the default operator is an 'OR' so that the above query will result in a search for the words 'iron' OR 'man'. ES has a notion of scoring for providing us a score sorted result set back for our queries.

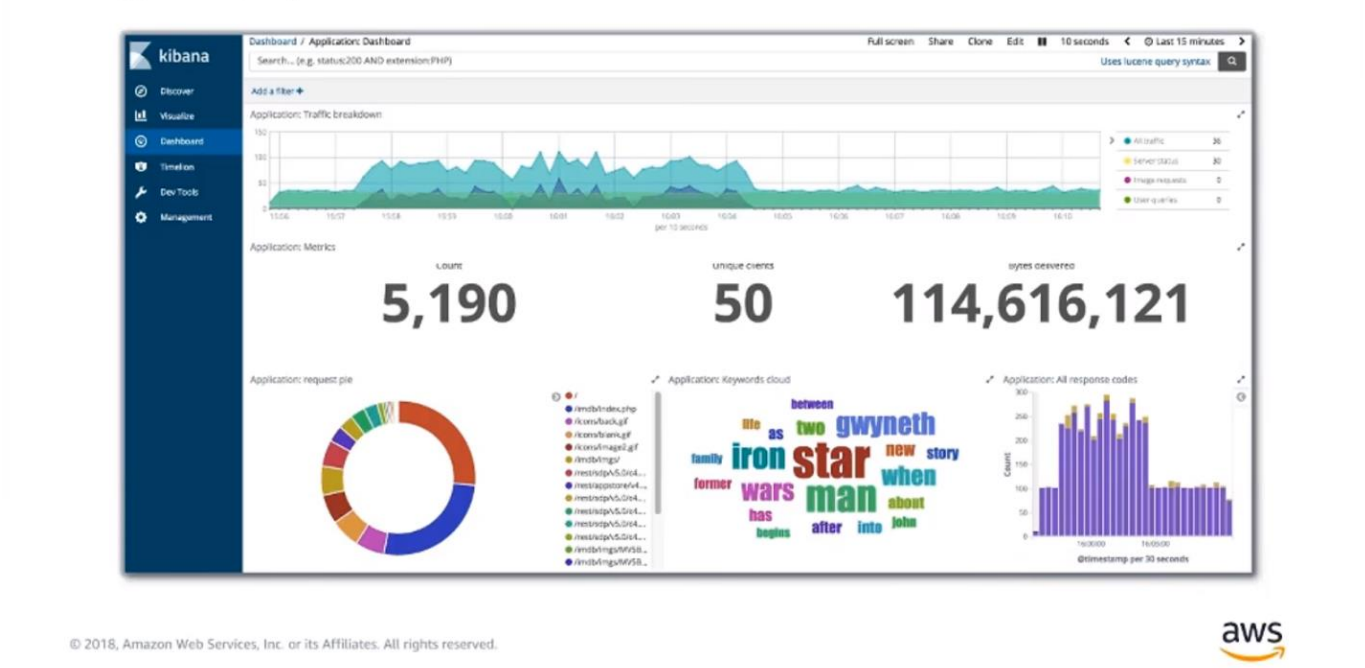


The Posting lists represent all the documents that have the search term(s) in them with the location_ids of where the words are located within the documents.

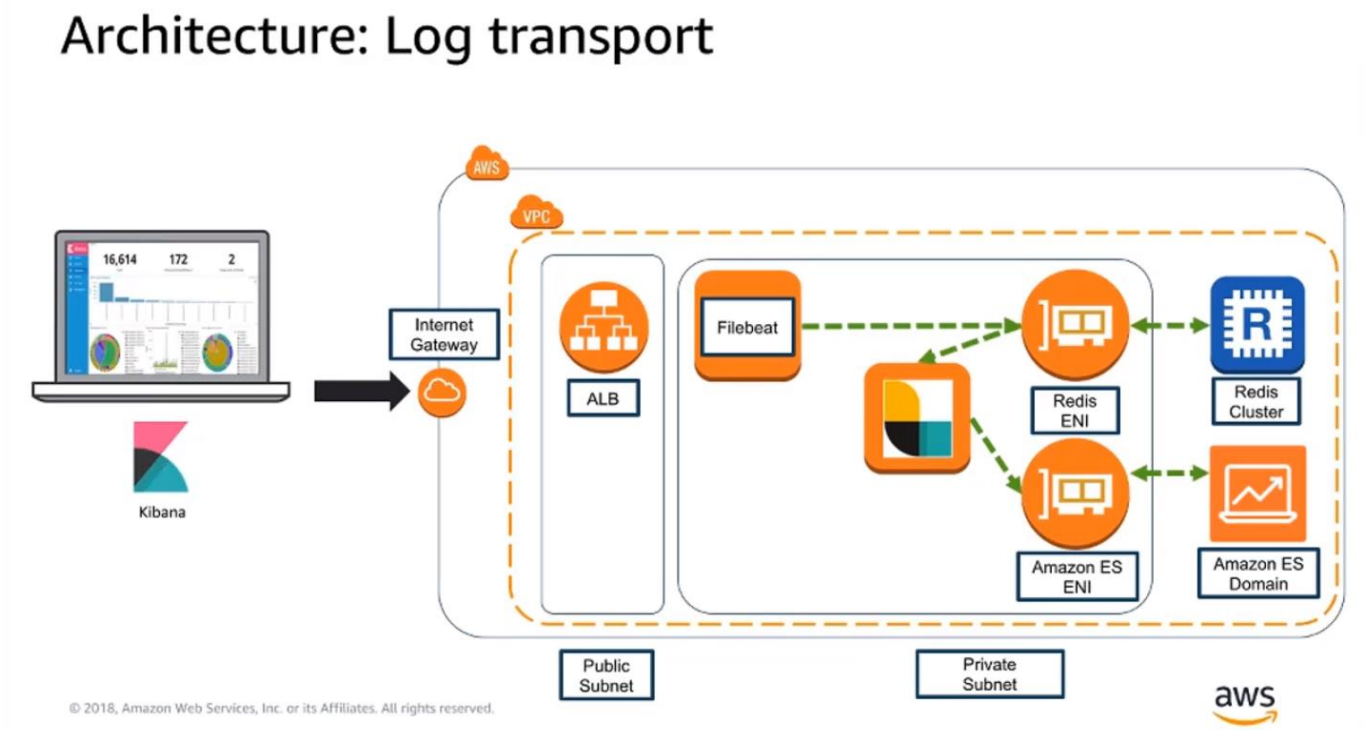


Above is a log analysis example for HTTP data in our Apache web logs

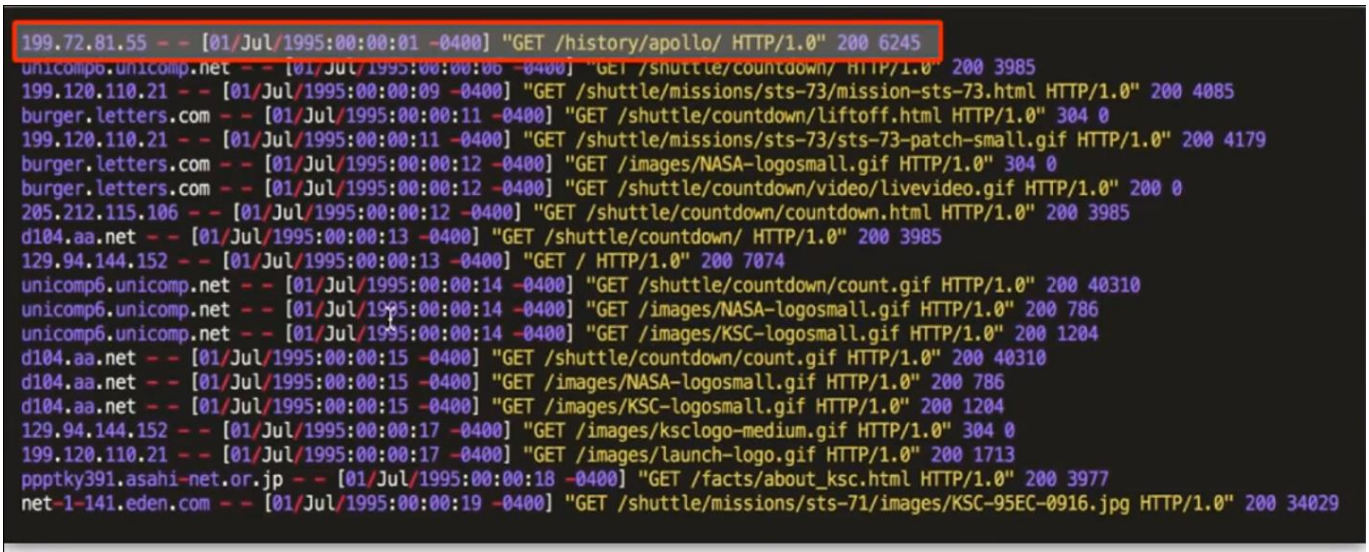




The XYZ company now have their data in ES and their frontends can start using the search functionality from the web app. The now want to monitor the infrastructure using log analysis.



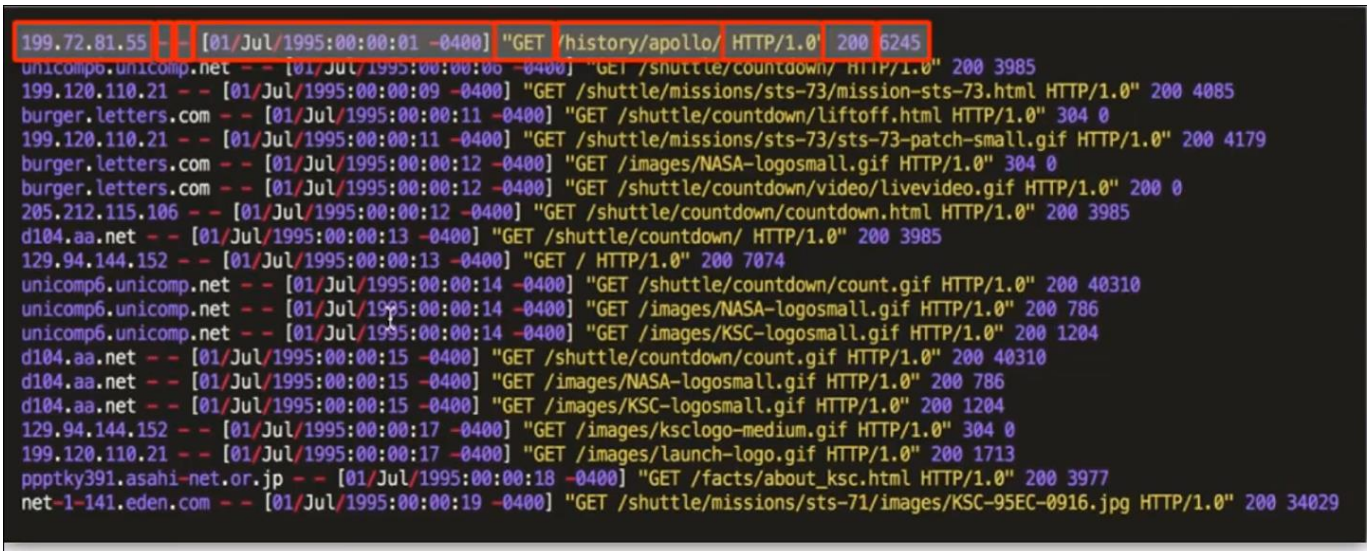
We are using an OSS tool called ***Filebeat*** to tail the log files and ships the log lines one at a time to a ***Redis*** buffer (not a cache) to receive the log lines and batch them up, sends/pulls the log lines to ***Logstash*** to be formatted and transformed, then sent to our ***ES*** ENI.



Each log line or other event constitutes a search *document*

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

aws



Key Idea

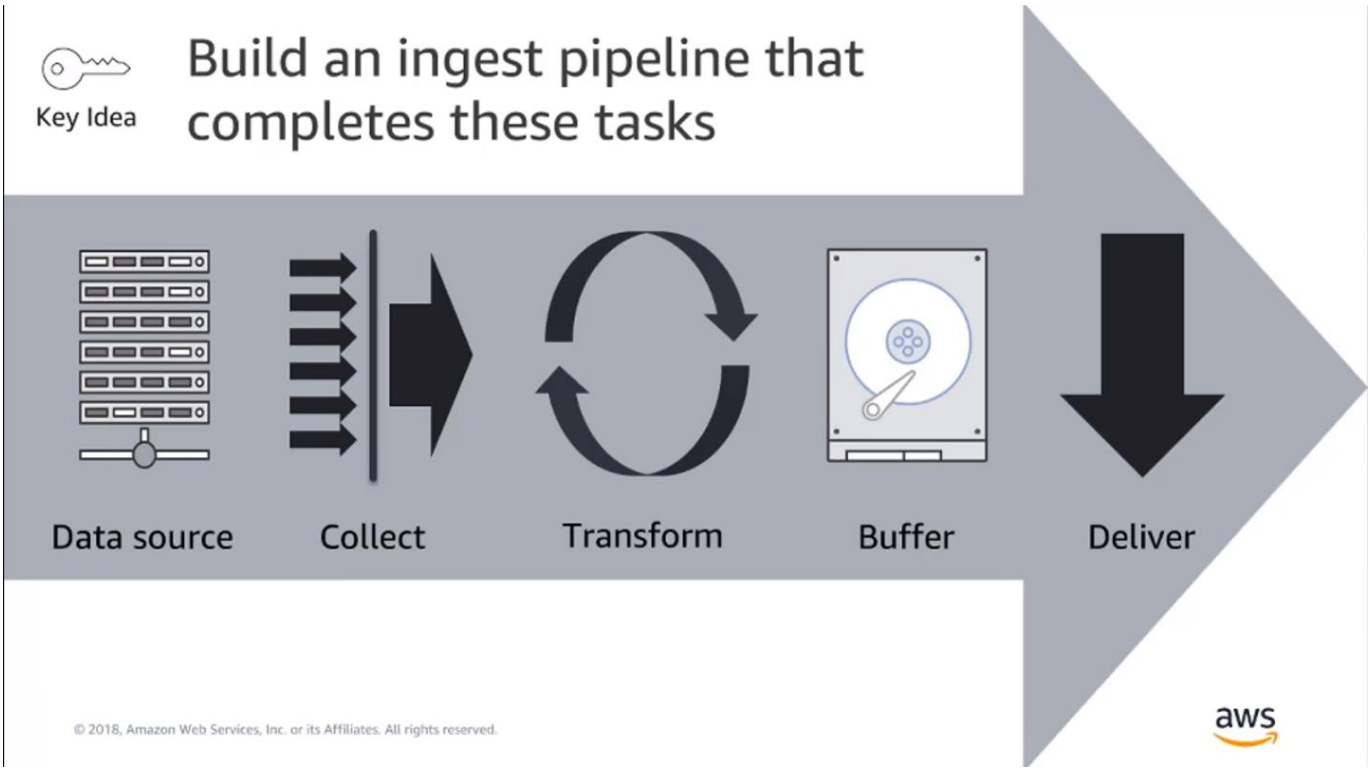
Each log line has various *fields* that contain data you can search and analyze

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

aws

Key Idea

Build an ingest pipeline that completes these tasks



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

aws

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

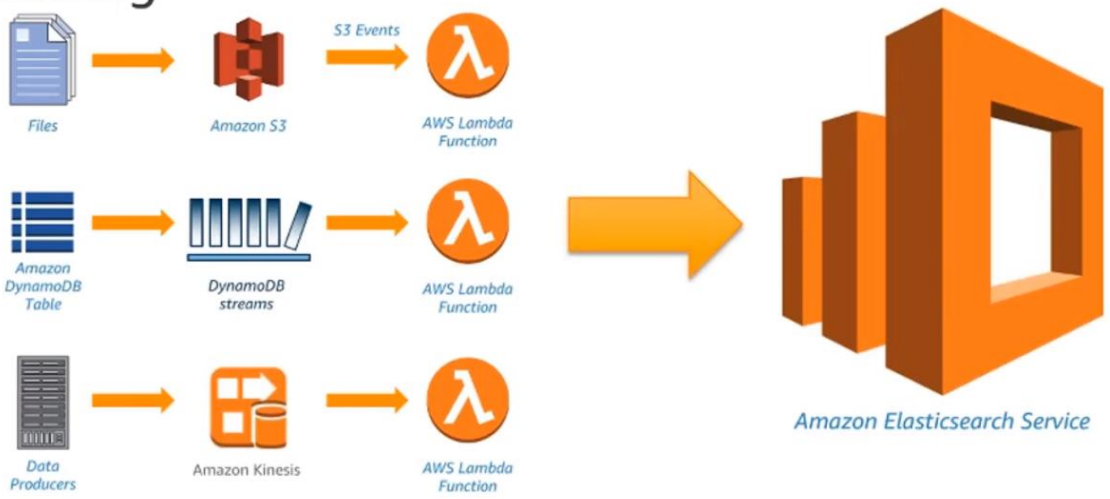
Deliver

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

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

aws

Amazon Lambda architectures for dynamic updating



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

aws

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

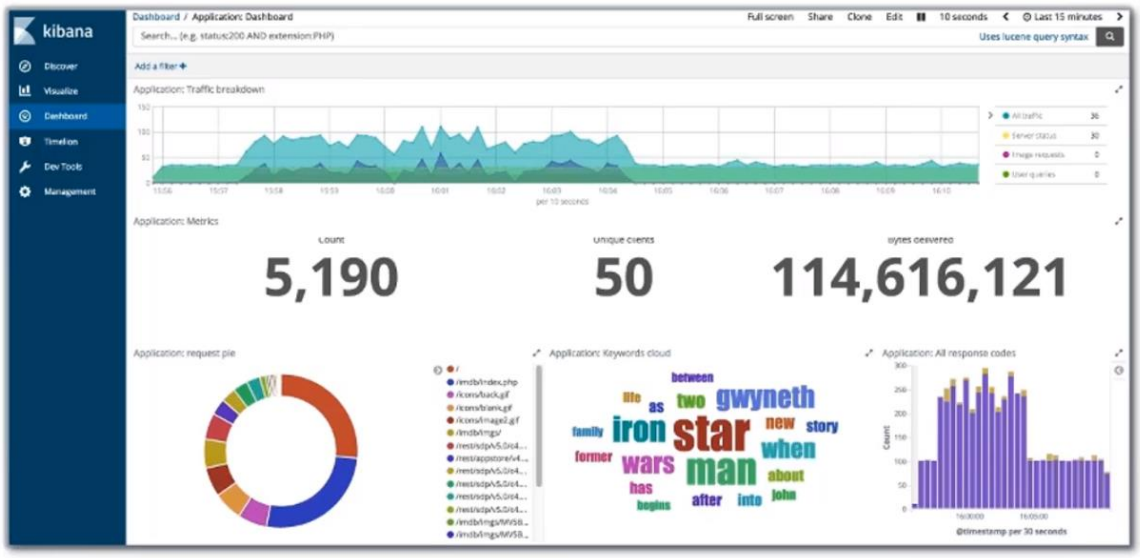
- On ingest, create indexes with a root string, e.g. logs_
- Depending on volume, rotate at regular intervals – normally daily
- Daily indexes simplify index management. Delete the oldest index to create more space on your cluster

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

aws

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.

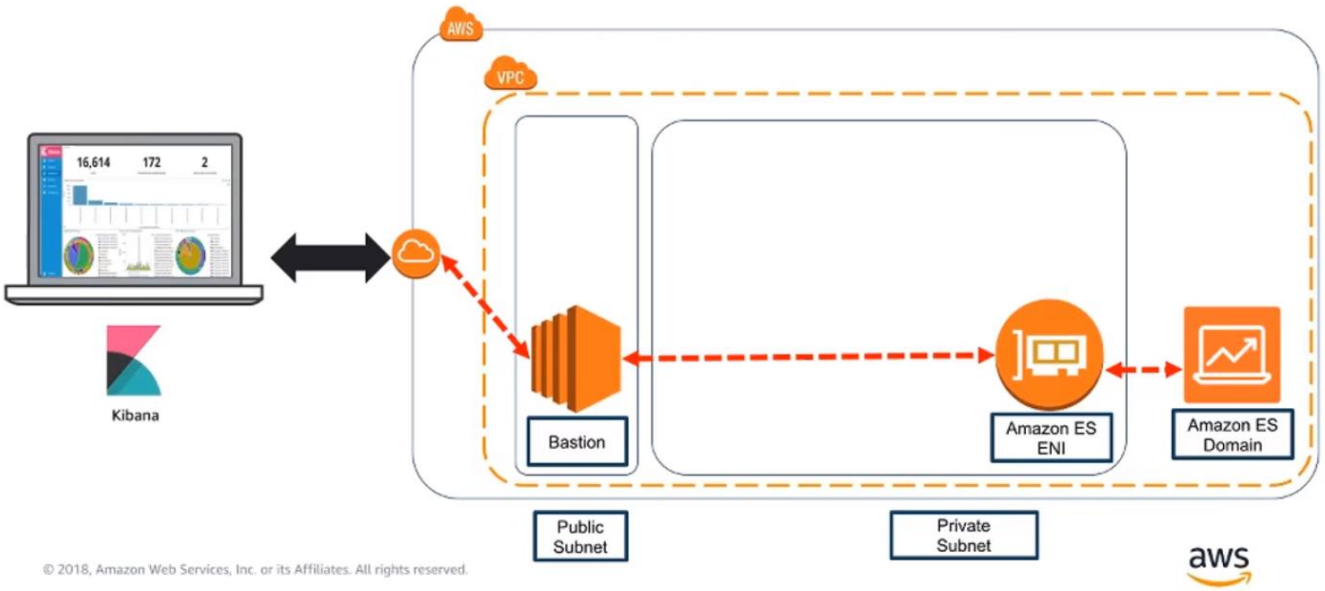
Log analysis



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

aws

Architecture: Log analytics

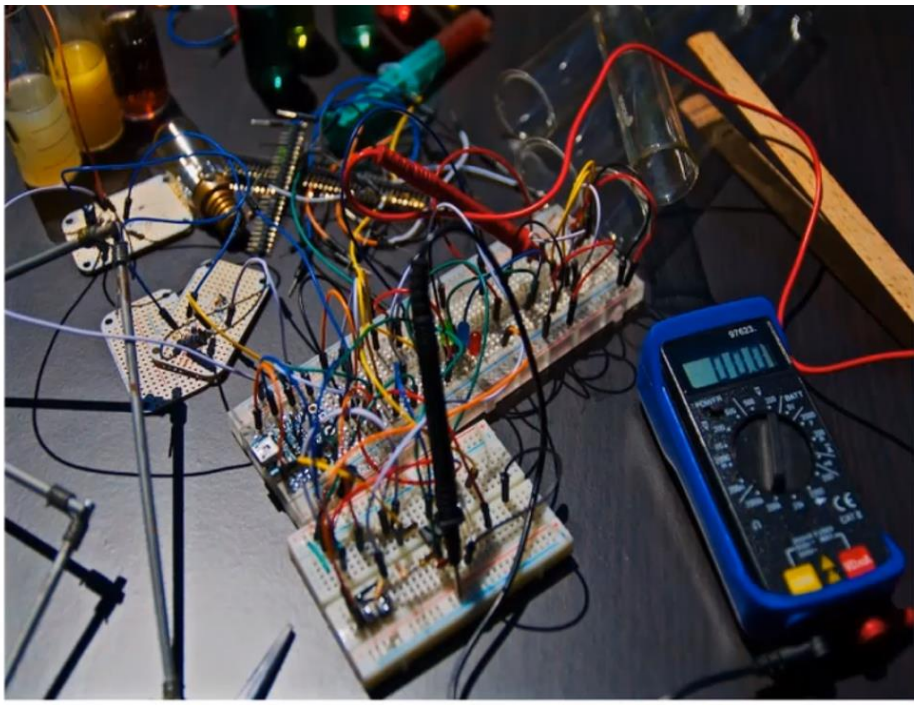


© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

aws

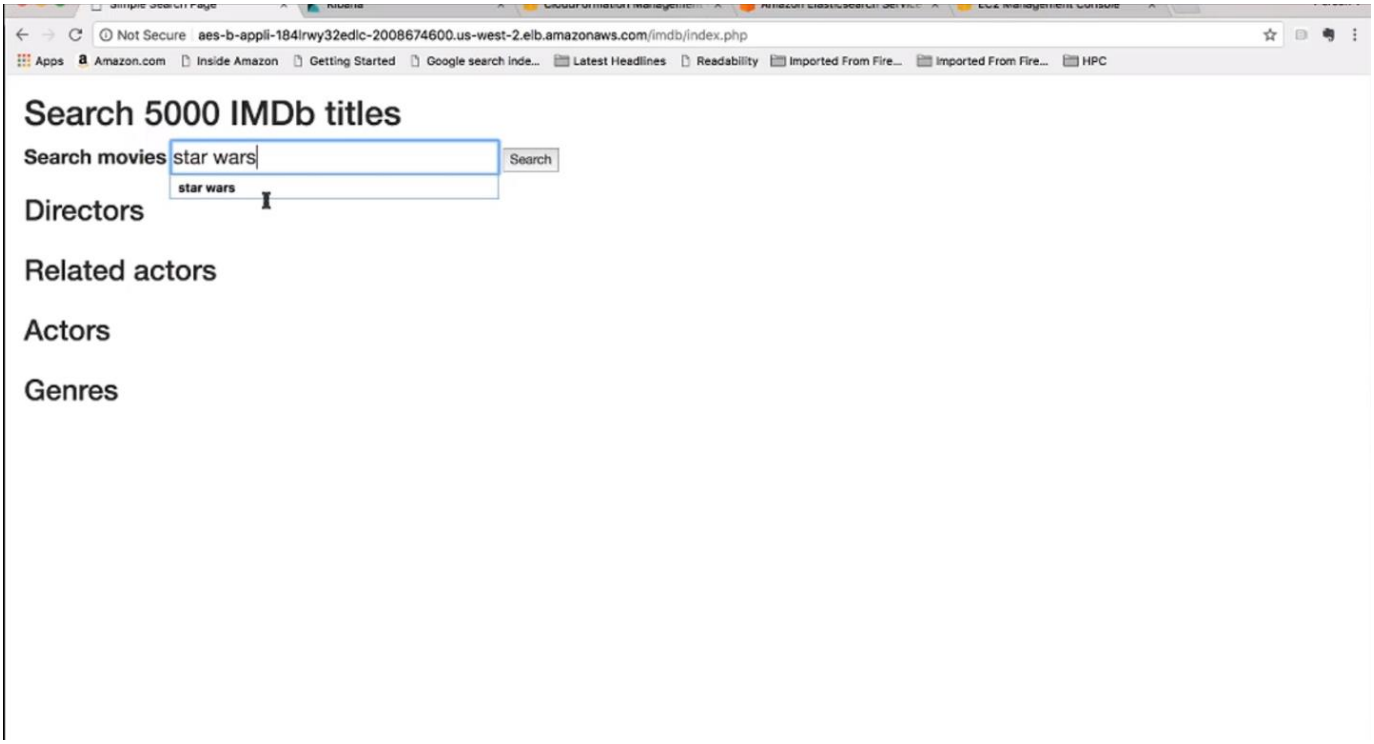
We will have Kibana talking to a public or bastion host within our public subnet of our VPC, the bastion then forwards the traffic to the ES service through the ENI and then provide the results back for visualization in Kibana

Demo
Log Analysis

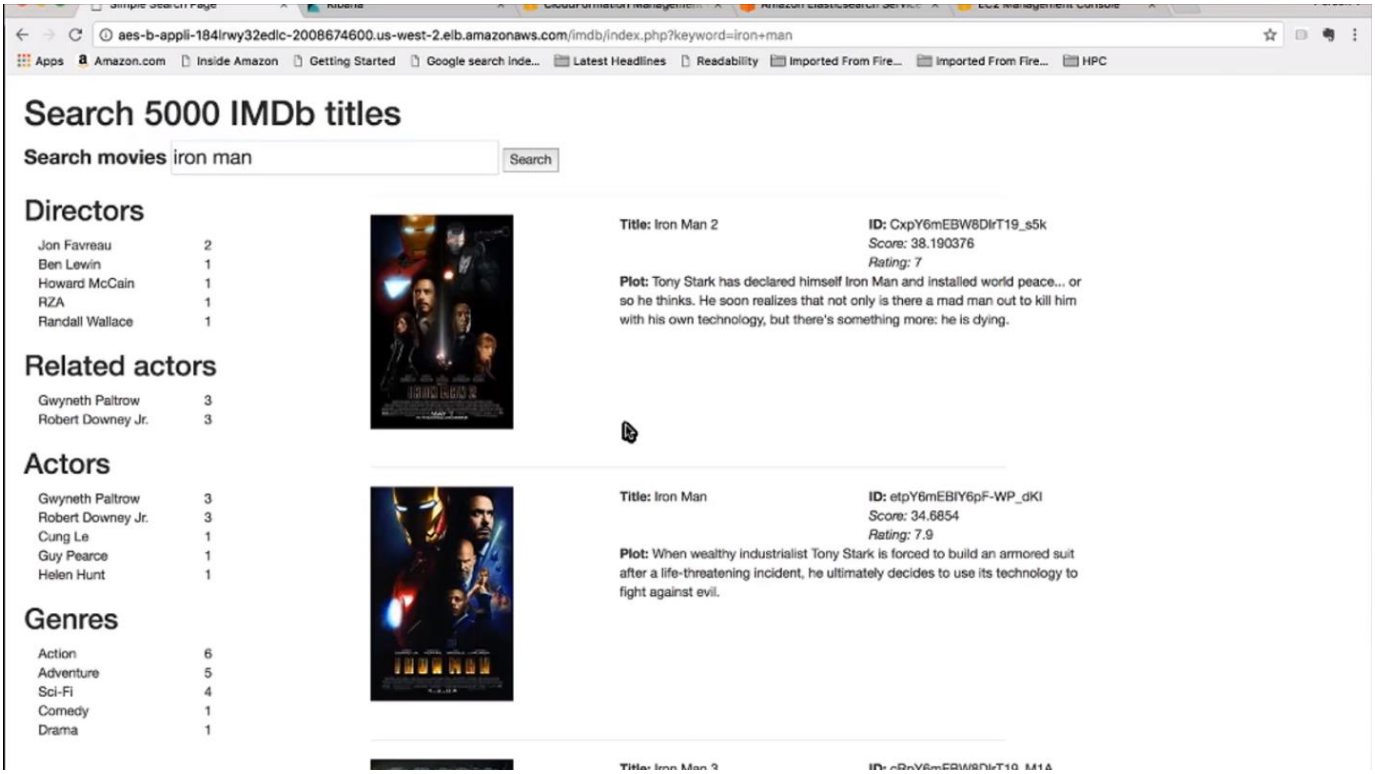


© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

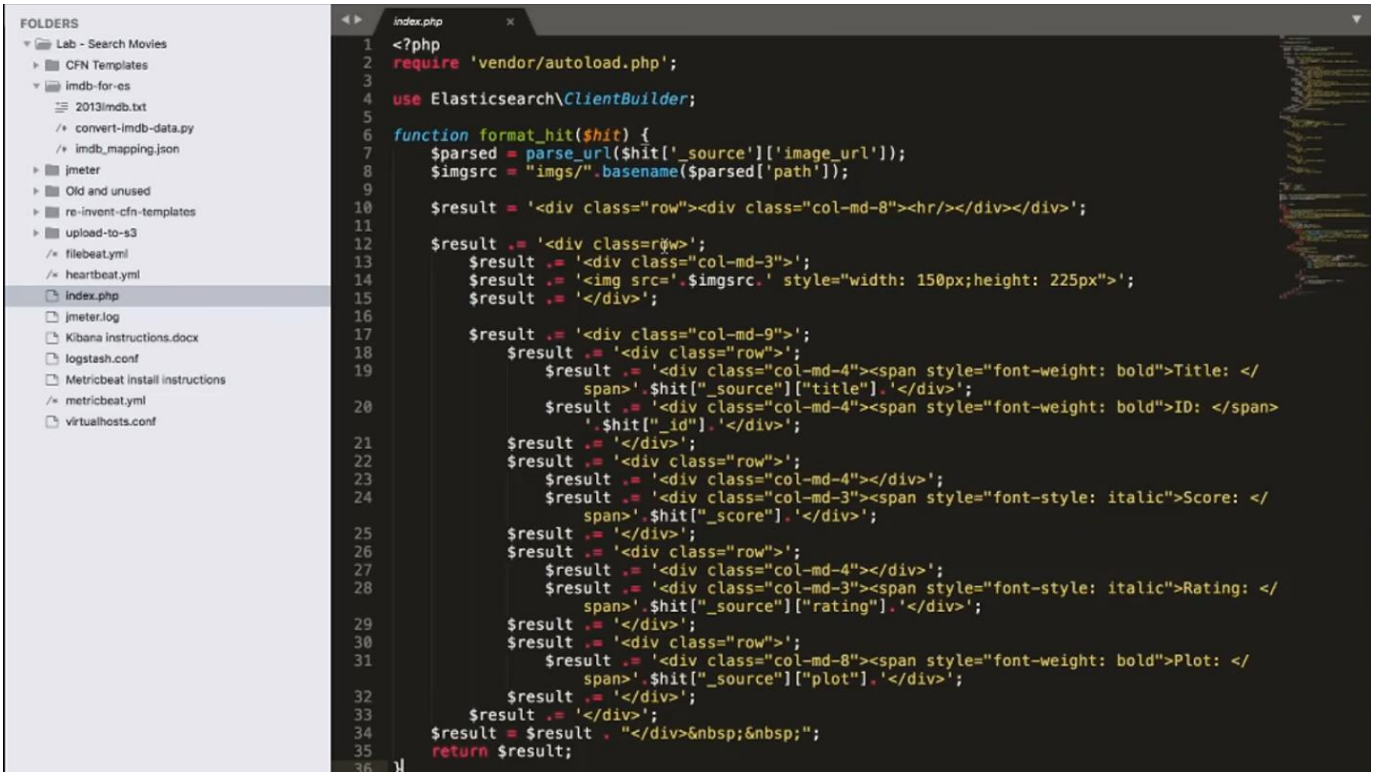
aws



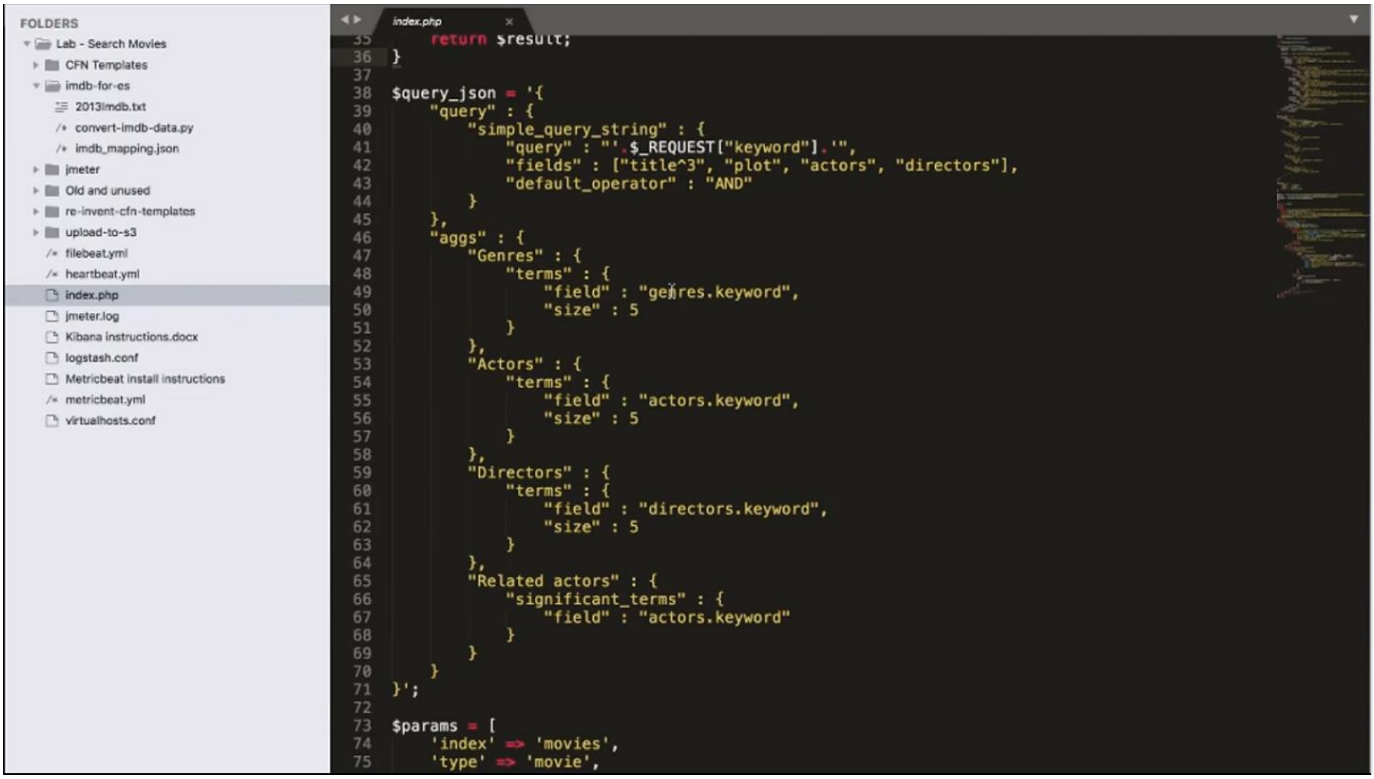
This is our search application running and we can search for movies



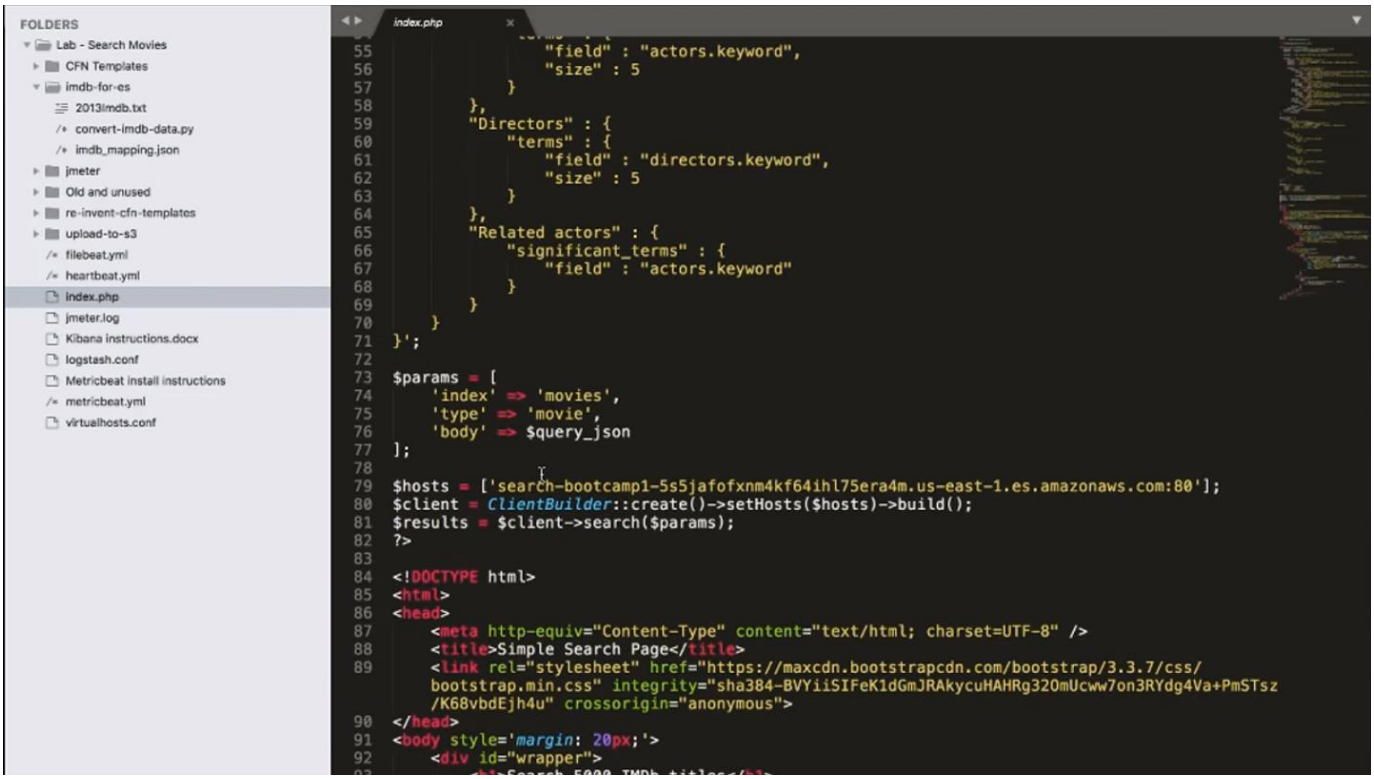
We then get a list of results back that we can display on the UI as above

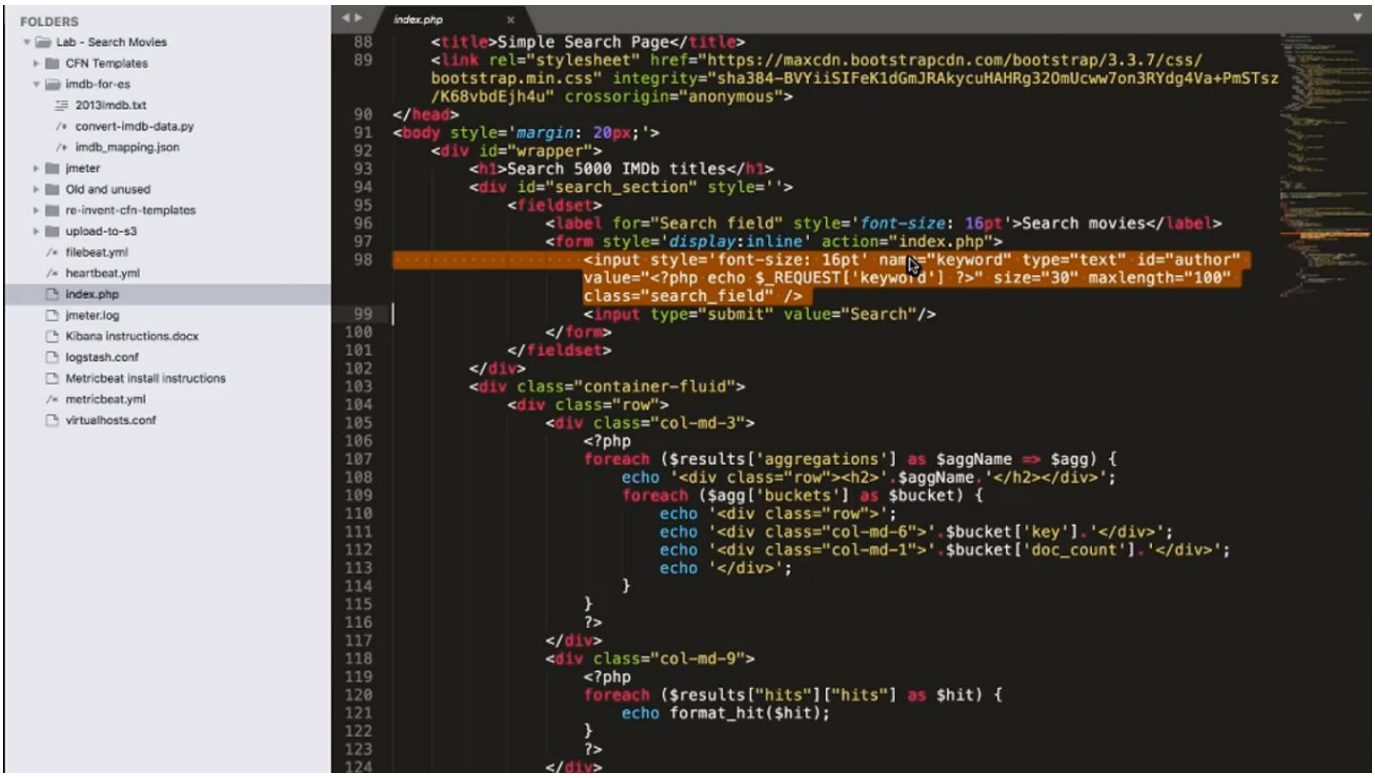


Here is the PHP code for developing the UI, we are using the Elasticsearch client from elastic.

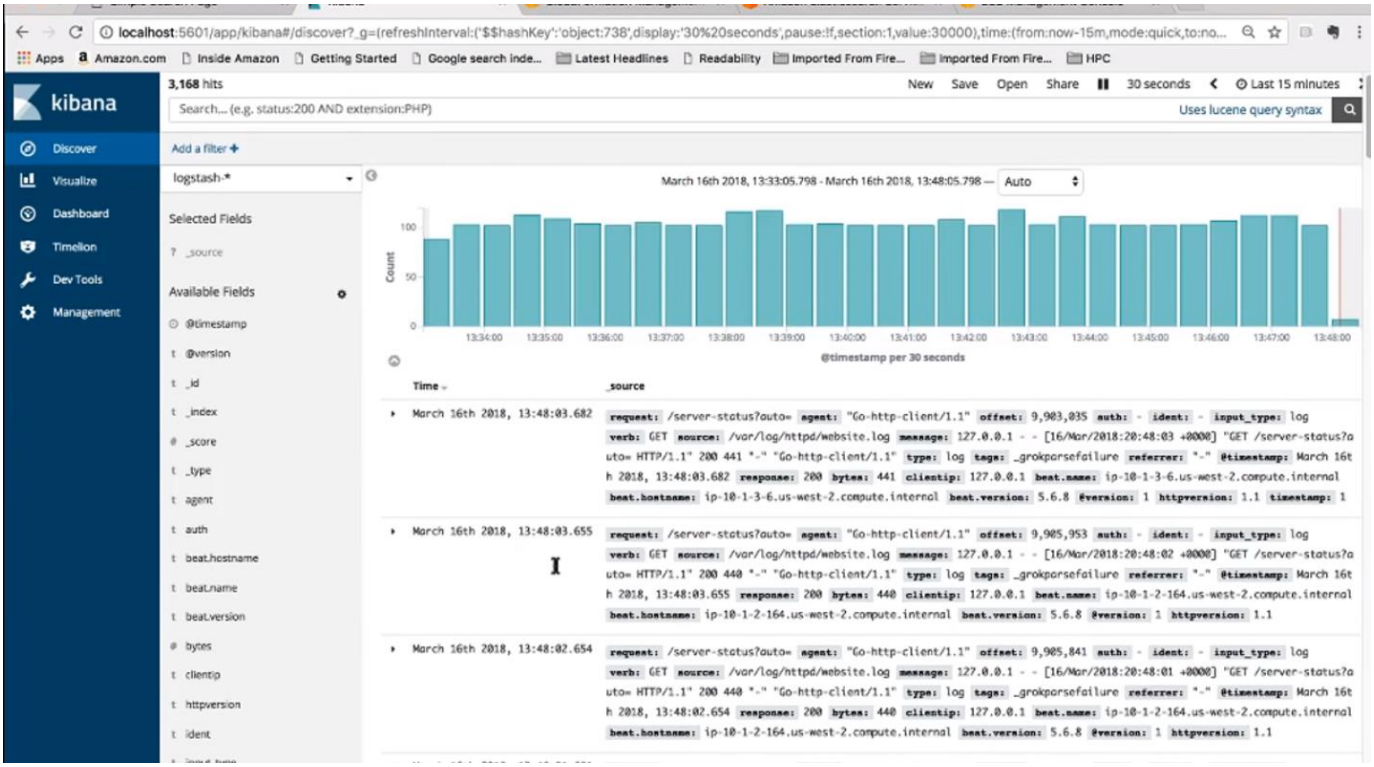


This is the simple querystring that we are using for the search button functionality, it takes in a keyword from the user via the input element on the UI. We are also doing aggregations on some other fields like Genres, Actors, Directors, and Related actors.

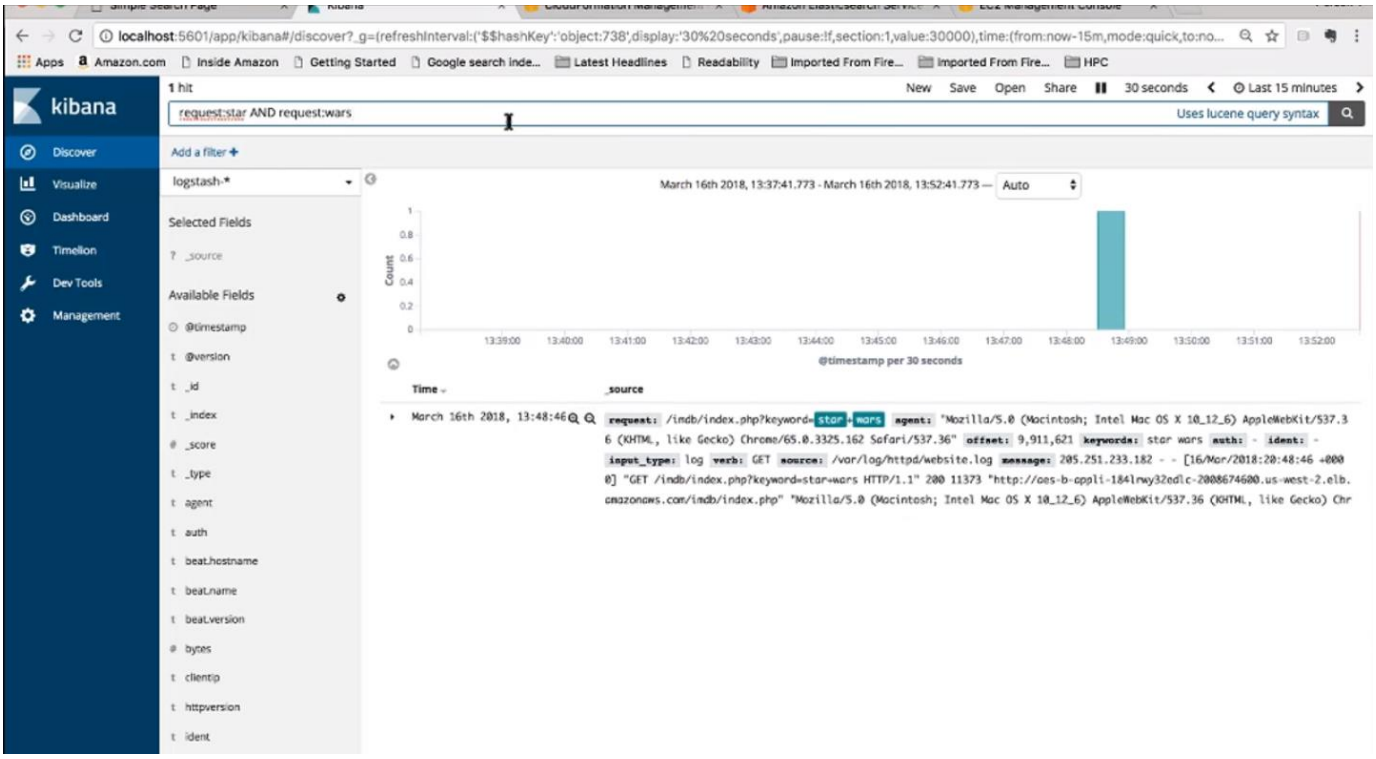


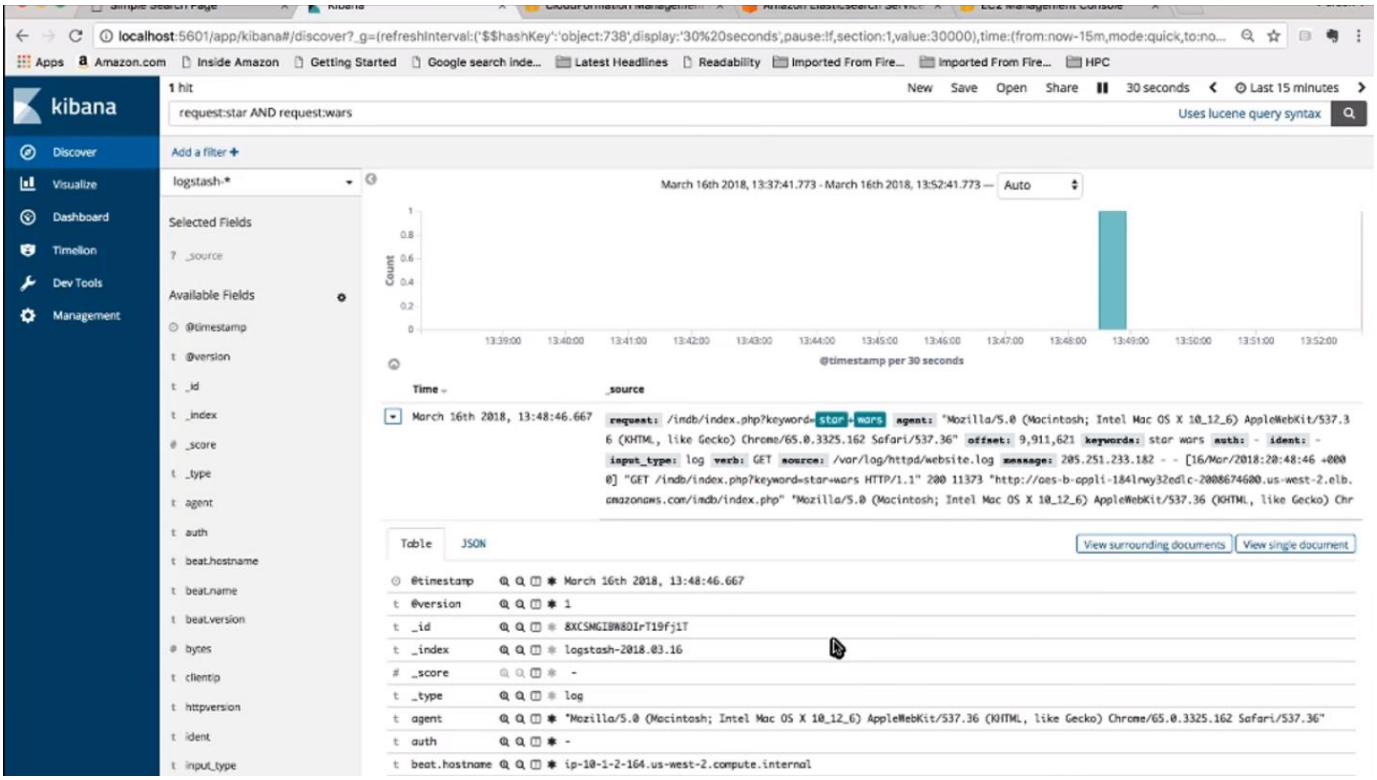


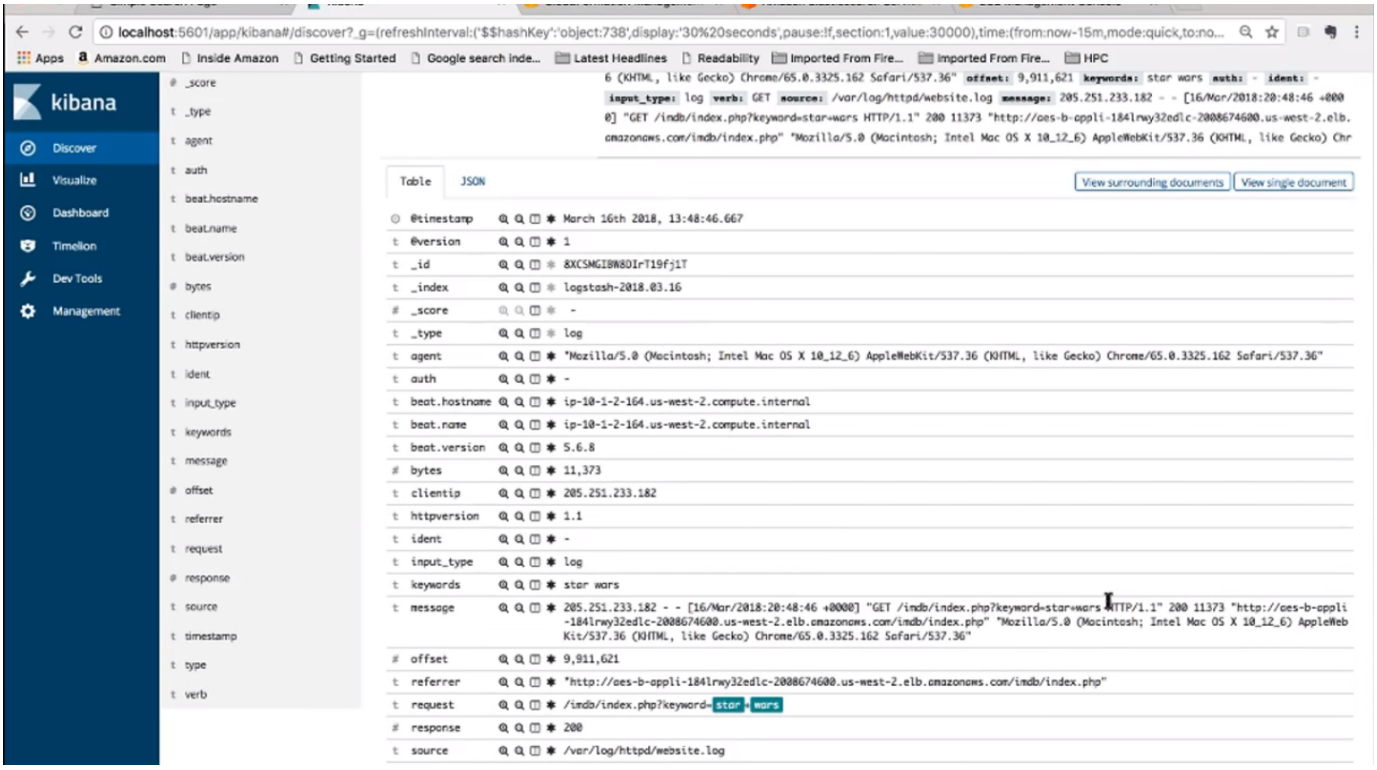
We then have the main body with the form in it as above. We already have Filebeat running on our HTTP servers that help to ship out the logs as they get created. We can then visualize the logs in Kibana

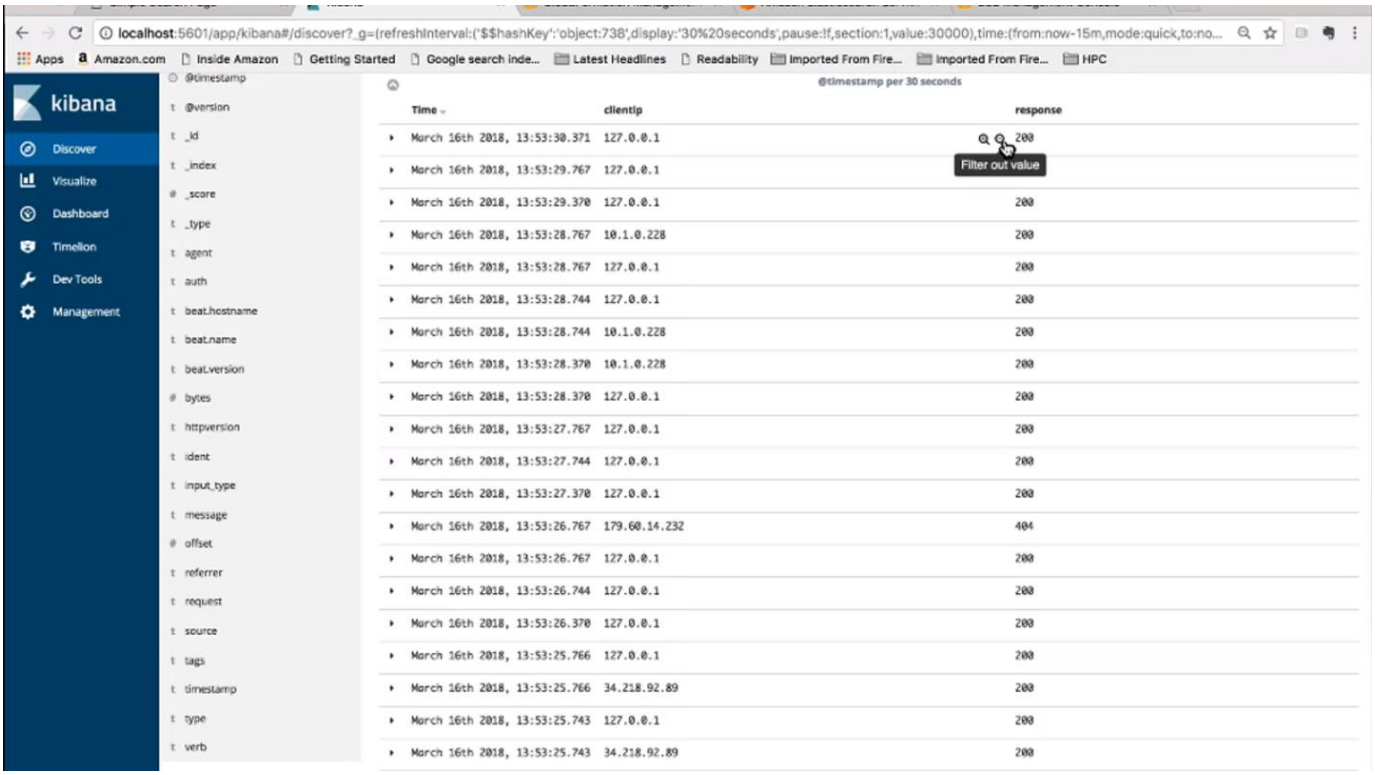


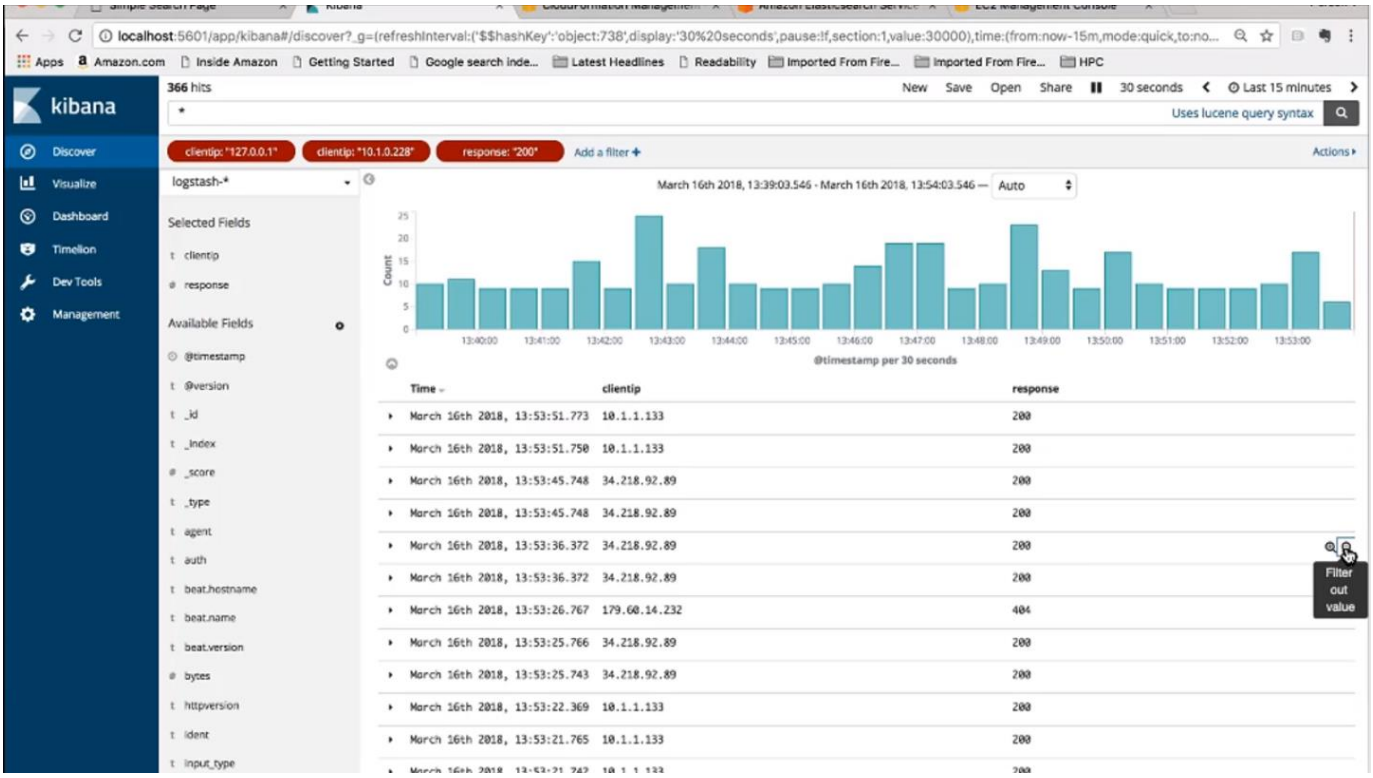
This is our main Kibana dashboard with all the outputs from Logstash

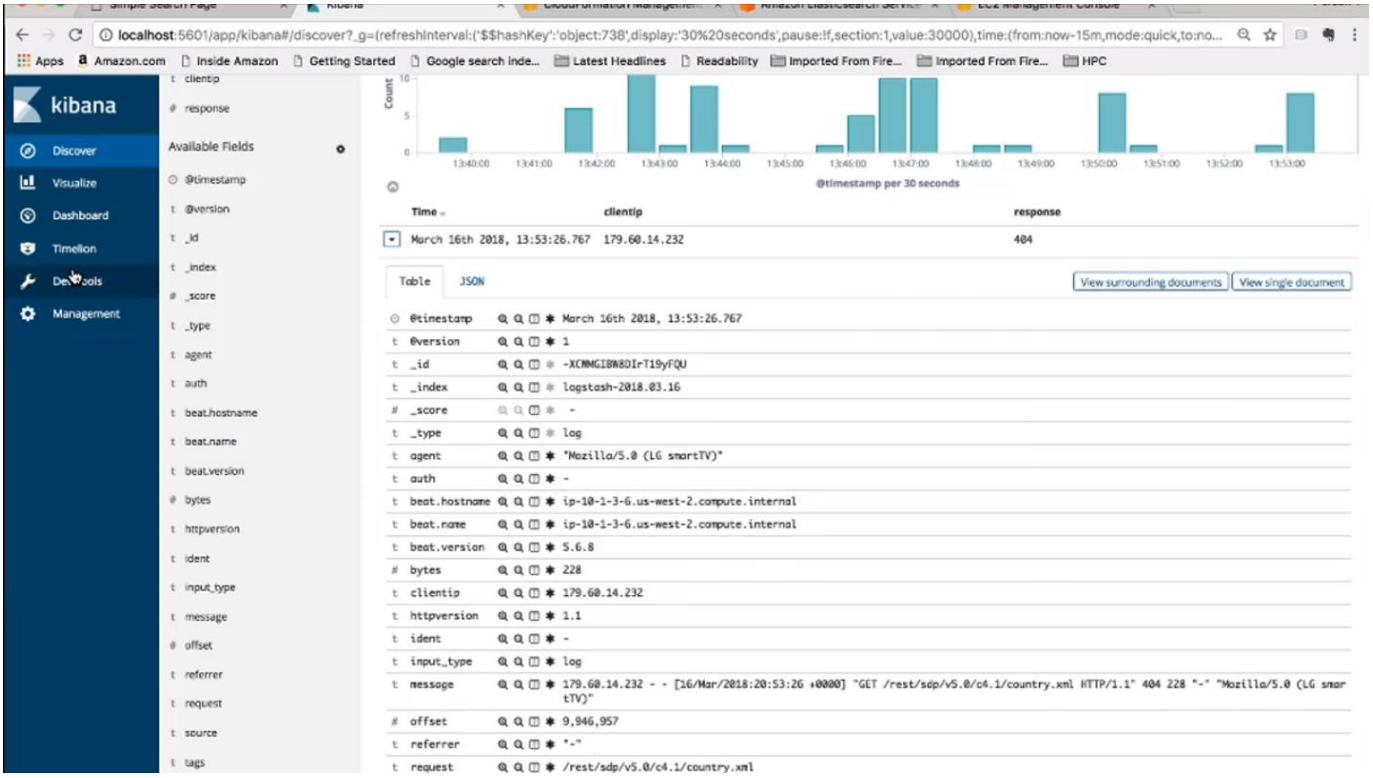

kibana
1 hit
request:star AND request:wars
Uses lucene query syntax
New Save Open Share 30 seconds Last 15 minutes
Discover
Visualize
Dashboard
Timelion
Dev Tools
Management
Add a filter
logstash-*
March 16th 2018, 13:37:41.773 - March 16th 2018, 13:52:41.773 — Auto
Selected Fields
_source
Available Fields
@timestamp
@version
_id
_index
_score
_type
agent
auth
beat.hostname
beat.name
beat.version
bytes
clientip
httpversion
ident
Count
@timestamp per 30 seconds
Time
_source
March 16th 2018, 13:48:46

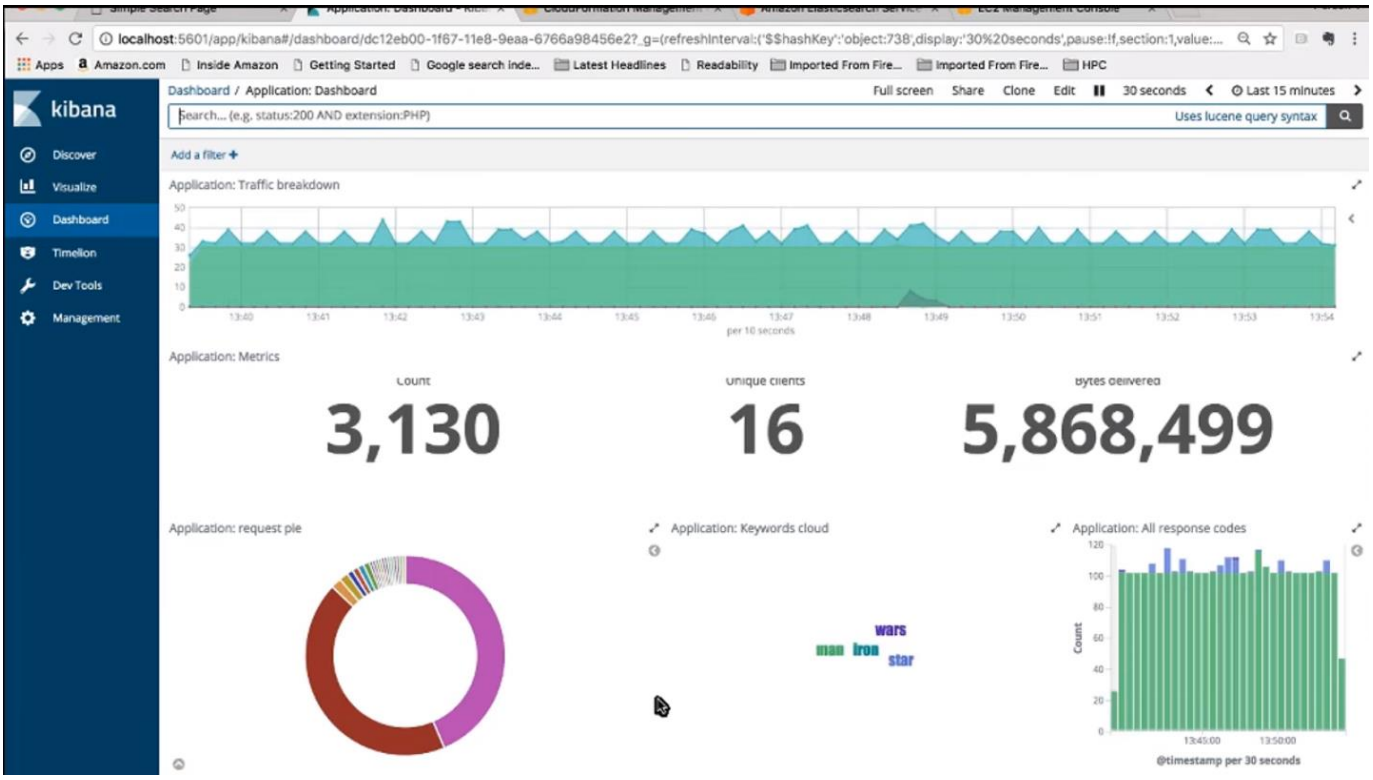

kibana
1 hit
request:star AND request:wars
Uses lucene query syntax
New Save Open Share 30 seconds Last 15 minutes
Discover
Visualize
Dashboard
Timelion
Dev Tools
Management
Add a filter
logstash-*
March 16th 2018, 13:37:41.773 - March 16th 2018, 13:52:41.773 — Auto
Selected Fields
_source
Available Fields
@timestamp
@version
_id
_index
_score
_type
agent
auth
beat.hostname
beat.name
beat.version
bytes
clientip
httpversion
ident
input_type
Count
@timestamp per 30 seconds
Time
_source
March 16th 2018, 13:48:46.667
Table
JSON
View surrounding documents
View single document
@timestamp March 16th 2018, 13:48:46.667
@version 1
_id 8XCSMGIBW8OIrT19fj1T
_index logstash-2018.03.16
_score -
_type log
agent "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/65.0.3325.162 Safari/537.36"
auth -
beat.hostname ip-10-1-2-164.us-west-2.compute.internal

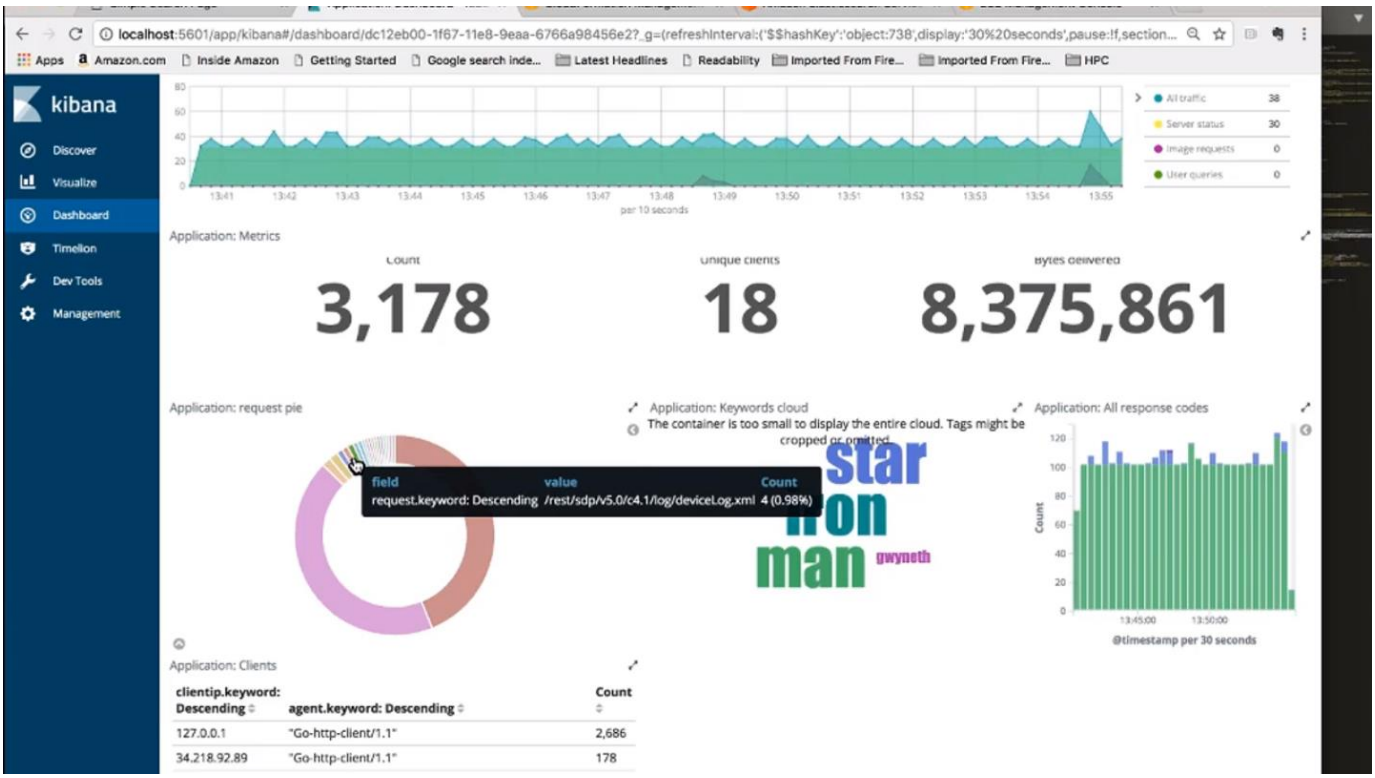

kibana
Discover
Visualize
Dashboard
Timelion
Dev Tools
Management
Table
JSON
View surrounding documents
View single document


kibana
Discover
Visualize
Dashboard
Timelion
Dev Tools
Management
@timestamp per 30 seconds
Time
clientip
response
Filter out value


kibana
366 hits
New Save Open Share 30 seconds Last 15 minutes
Uses lucene query syntax
clientip: "127.0.0.1"
clientip: "10.1.0.228"
response: "200"
Add a filter
Actions
Discover
Visualize
Dashboard
Timelion
Dev Tools
Management
logstash-*
March 16th 2018, 13:39:03.546 - March 16th 2018, 13:54:03.546 — Auto
Selected Fields
clientip
response
Available Fields
@timestamp
@version
_id
_index
_score
_type
agent
auth
beat.hostname
beat.name
beat.version
bytes
httpversion
ident
input_type
Count
@timestamp per 30 seconds
Time
clientip
response
March 16th 2018, 13:53:51.773 10.1.1.133 200
March 16th 2018, 13:53:51.750 10.1.1.133 200
March 16th 2018, 13:53:45.748 34.218.92.89 200
March 16th 2018, 13:53:45.748 34.218.92.89 200
March 16th 2018, 13:53:36.372 34.218.92.89 200
March 16th 2018, 13:53:36.372 34.218.92.89 200
March 16th 2018, 13:53:26.767 179.60.14.232 404
March 16th 2018, 13:53:25.766 34.218.92.89 200
March 16th 2018, 13:53:25.743 34.218.92.89 200
March 16th 2018, 13:53:22.369 10.1.1.133 200
March 16th 2018, 13:53:21.765 10.1.1.133 200
Filter out value


kibana
Discover
Visualize
Dashboard
Timelion
Dev Tools
Management
Available Fields
@timestamp
@version
_id
_index
_score
_type
agent
auth
beat.hostname
beat.name
beat.version
bytes
httpversion
ident
input_type
message
offset
referrer
request
source
tags
Count
@timestamp per 30 seconds
Time
clientip
response
March 16th 2018, 13:53:26.767 179.60.14.232 404
Table
JSON
View surrounding documents
View single document
@timestamp March 16th 2018, 13:53:26.767
@version 1
_id -XCMMGIBW8OIrT19yFQU
_index logstash-2018.03.16
_score -
_type log
agent "Mozilla/5.0 (LG smartTV)"
auth -
beat.hostname ip-10-1-3-6.us-west-2.compute.internal
beat.name ip-10-1-3-6.us-west-2.compute.internal
beat.version 5.6.8
bytes 228
clientip 179.60.14.232
httpversion 1.1
ident -
input_type log
message 179.60.14.232 - - [16/Mar/2018:20:53:26 +0000] "GET /rest/sdp/v5.0/c4.1/country.xml HTTP/1.1" 404 228 "-" "Mozilla/5.0 (LG smartTV)"
offset 9,946,957
referrer "-"
request /rest/sdp/v5.0/c4.1/country.xml


kibana
Discover
Visualize
Dashboard
Timelion
Dev Tools
Management
Dashboard / Application: Dashboard
Full screen
Share
Clone
Edit
30 seconds
Last 15 minutes
Search... (e.g. status:200 AND extension:PHP)
Uses lucene query syntax
Add a filter
Application: Traffic breakdown
per 10 seconds
Application: Metrics
Count
3,130
Unique clients
16
Bytes delivered
5,868,499
Application: request pie
Application: Keywords cloud
wars
man
iron
star
Application: All response codes
Count
@timestamp per 30 seconds


kibana
Discover
Visualize
Dashboard
Timelion
Dev Tools
Management
All traffic 38
Server status 30
Image requests 0
User queries 0
per 10 seconds
Application: Metrics
Count
3,178
Unique clients
18
Bytes delivered
8,375,861
Application: request pie
field value Count
request.keyword: Descending /rest/sdp/v5.0/c4.1/log/deviceLog.xml 4 (0.98%)
Application: Keywords cloud
The container is too small to display the entire cloud. Tags might be cropped or omitted.
star
iron
man
gwyneth
Application: All response codes
Count
@timestamp per 30 seconds
Application: Clients
clientip.keyword: Descending
agent.keyword: Descending
Count
127.0.0.1 "Go-http-client/1.1" 2,686
34.218.92.89 "Go-http-client/1.1" 178

CASE STUDY: EXPEDIA

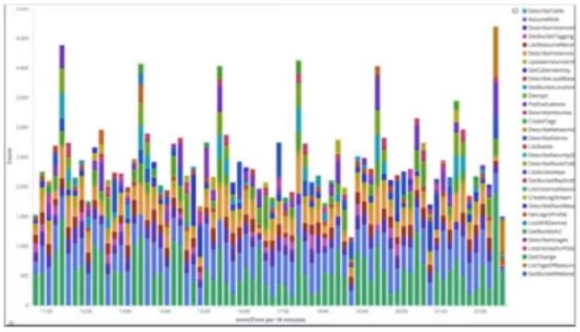
Application Monitoring & Root-cause Analysis

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

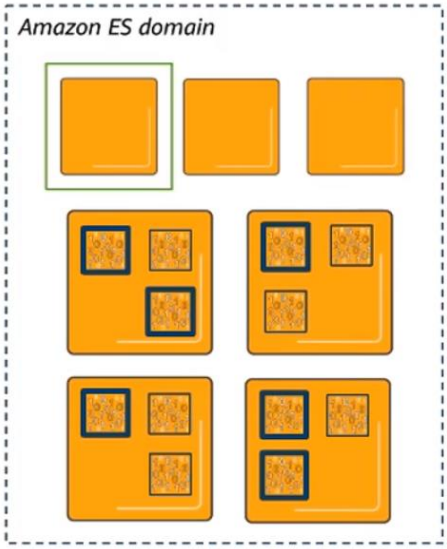
Scalable: Cluster sizes are able to grow to accommodate additional log sources

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Key Idea

Use 3 dedicated master instances in production

Master instances orchestrate and make your cluster more stable



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

aws

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.

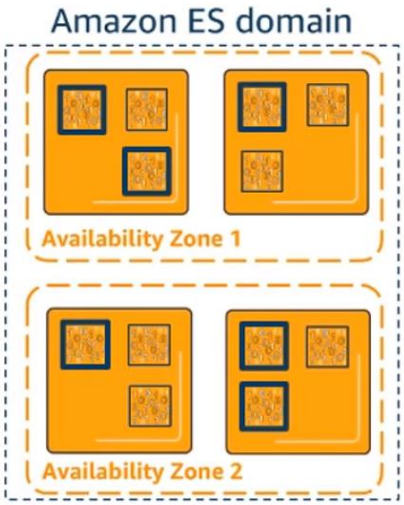
aws

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.

aws

This checkbox provides HA feature to your ES cluster

Set CloudWatch alarms for at least these metrics

Name	Metric	Threshold	Periods
ClusterStatus.red	Maximum	>= 1	1
ClusterIndexWritesBlocked	Maximum	>= 1	1
CPUUtilization/MasterCPUUtilization	Average	>= 80%	3
JVMMemoryPressure/Master...	Maximum	>= 80%	3
FreeStorageSpace	Minimum	<= (25% of avail space)	1
AutomatedSnapshotFailure	Maximum	>= 1	1

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

aws

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.

aws

Thank you!

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

aws