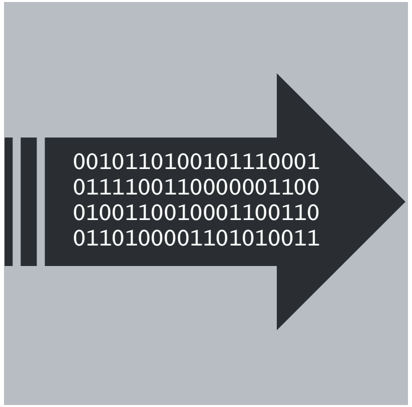
Secure your Amazon Elasticsearch Service Domain

Jon Handler, Principal Solutions Architect

September 17, 2018

Elasticsearch: Purpose built for search and analysis



Text search

Natural language

Boolean queries

Relevance

Streaming data

High-volume ingest

Near real time

Distributed storage

Analysis

Time-based visualizations

Nestable statistics

Time series tools

Amazon Elasticsearch Service

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

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

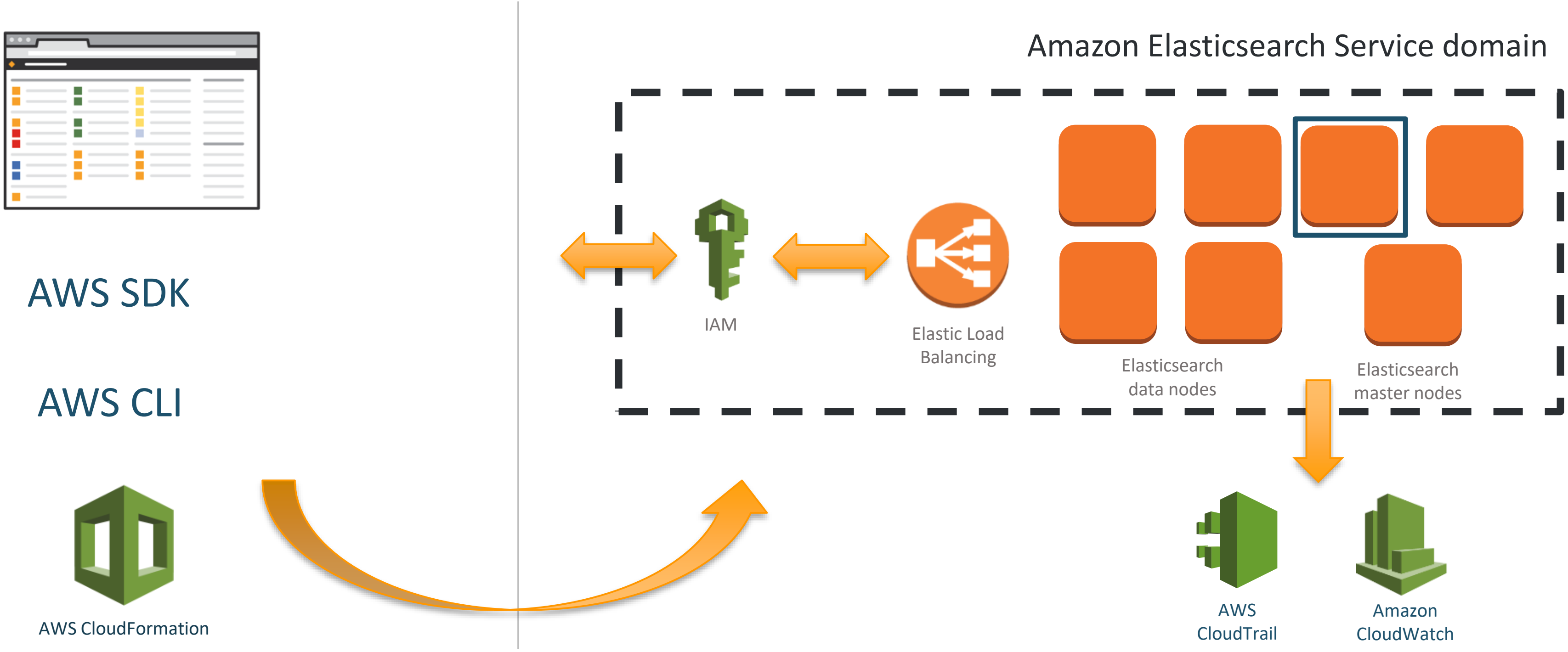
Highly Available

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

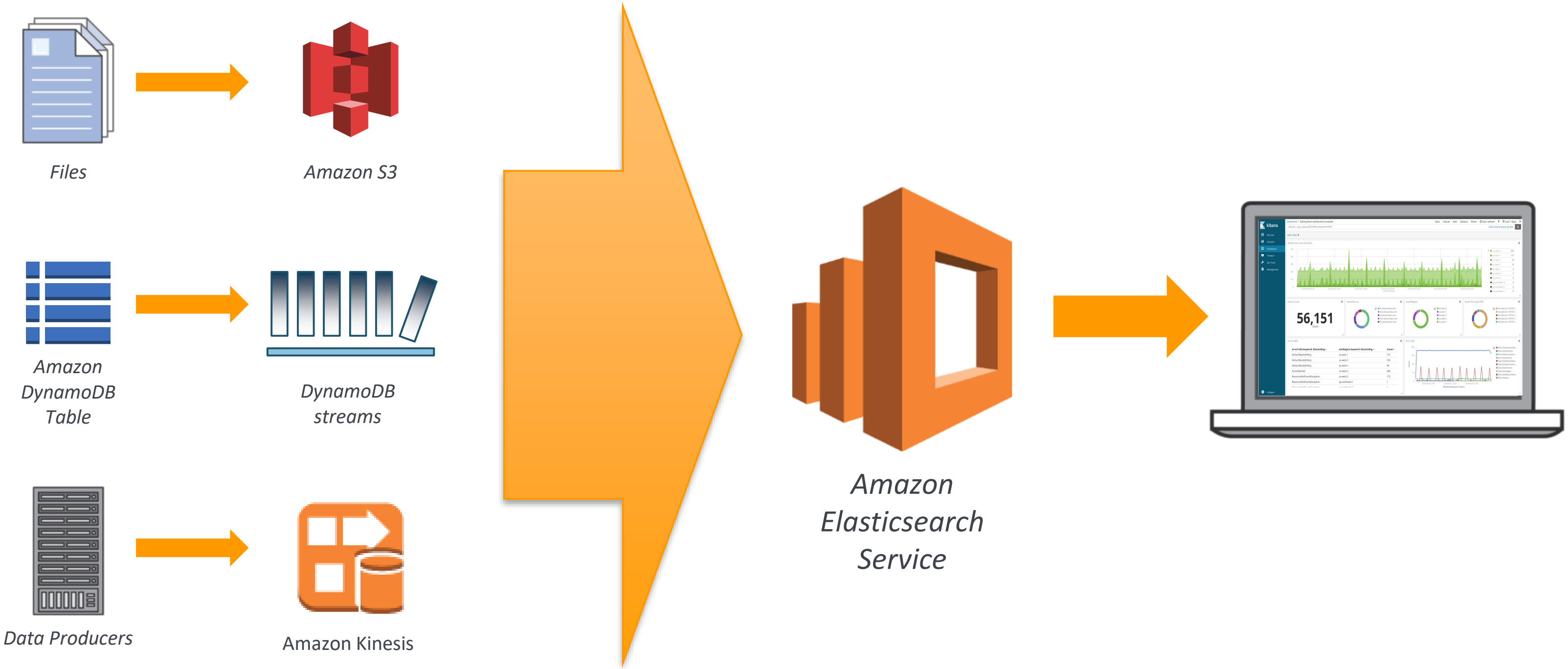
Tightly Integrated with Other AWS Services

Seamless data ingestion, security, auditing and orchestration

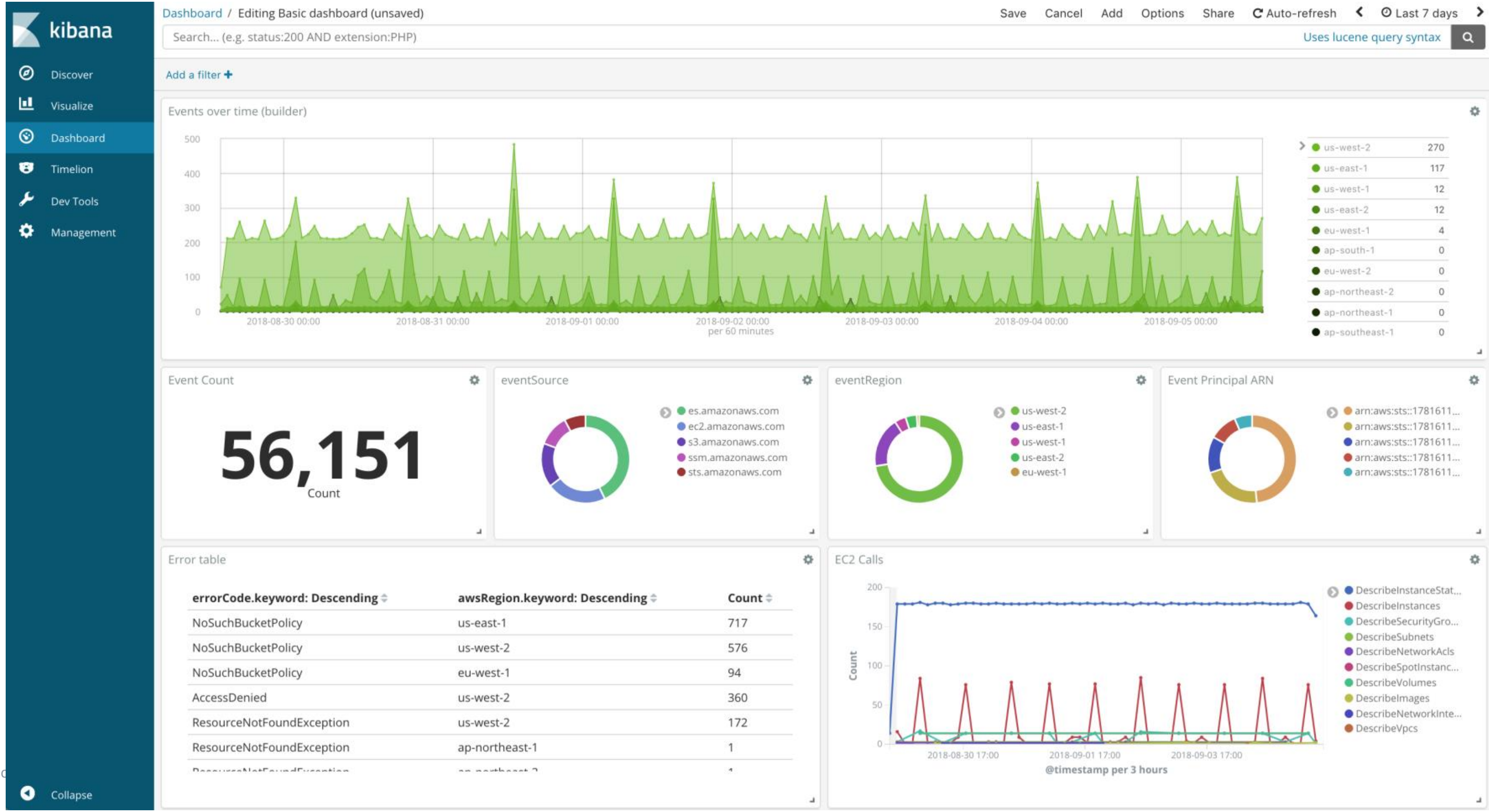
Service architecture



Service usage patterns



Example – AWS CloudTrail



errorCode.keyword: Descending	awsRegion.keyword: Descending	Count
NoSuchBucketPolicy	us-east-1	717
NoSuchBucketPolicy	us-west-2	576
NoSuchBucketPolicy	eu-west-1	94
AccessDenied	us-west-2	360
ResourceNotFoundException	us-west-2	172
ResourceNotFoundException	ap-northeast-1	1
ResourceNotFoundException	ap-northeast-2	1

Elasticsearch leading use cases

Application Monitoring & Root-cause Analysis

Provides developers with a high performance, self-service operational monitoring and analytics platform

Security Information and Event Management (SIEM)

Enables security practitioners to centralize and analyze events from across the entire organization

IoT & Mobile

Gives developers and lines of business users real-time location-aware insights into their device fleets

Business & Clickstream Analytics

Provides business users with a real-time view of the performance of their web content and e-commerce platforms

Amazon Elasticsearch Service customers

Software & Internet

Adobe

IAC

AUTODESK

Education Technology

Mc Graw Hill Education

INSTRUCTURE

Blackboard

Financial Services

THOMSON REUTERS

stripe

KICKSTARTER

BioTech and Pharma

MONSANTO

Bristol-Myers Squibb

Media and Entertainment

mlbam

NETFLIX

FOX

Social Media

cookpad

ancestry

Nextdoor

Telecommunications

Jio

COMCAST

T · · Mobile ·

Travel & Transportation

Expedia

UBER

Real Estate

Zillow

move

Logistics & Operations

here

elementum

infor

Publishing

FT FINANCIAL TIMES

ELSEVIER

The Washington Post

Other

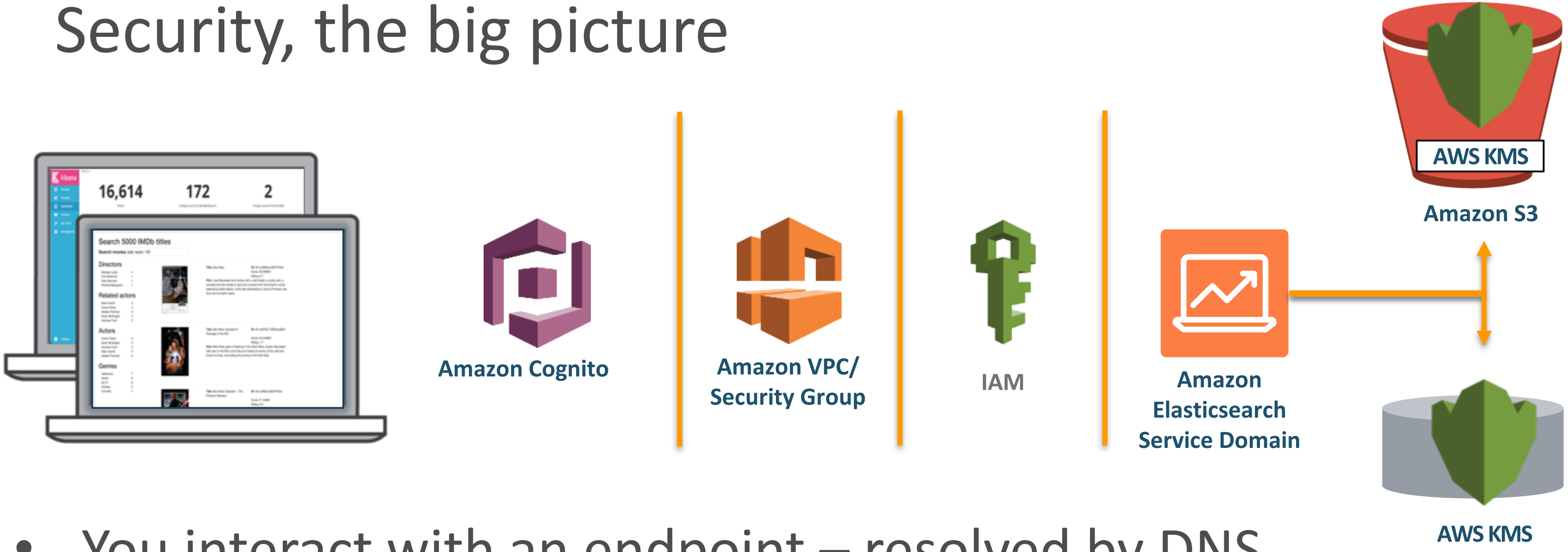
Canon

SAMSUNG

British Gas

aws

Security, the big picture

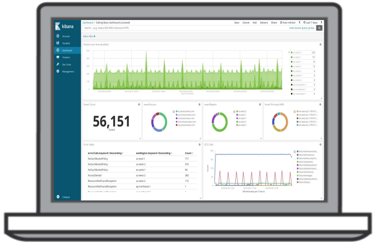


- You interact with an endpoint – resolved by DNS
- Cognito for authentication and external identities.
- VPC for restricting to your IP address space
- IAM to control actions
- KMS for encryption at rest

IAM Policies

Public endpoints use IAM policies exclusively

Kinesis Data Firehose



Ad-hoc, local proxy

Kibana with proxy

Lambda with signing

Code access

Policy skeleton

```
{
    "Version": "2012-10-17",
    "Statement": [ {
        "Effect":...
        "Principal": [...]
        "Action": [...],
        "Resource": [...],
        "Condition": [...]
        } ]
}
```

- Effect: Allow or Deny
- Principal: AWS account ID
- Action
  - HTTP verbs
  - Service actions
- Resource: Amazon ES domain/index
- Condition: IP Address

Policy 1: public access, signed requests

- Requests must be signed. User-name-1 can run all actions against all indices

```
{
    "Sid": "",
    "Effect": "Allow",
    "Principal": {
        "AWS": [
            "arn:aws:iam::12345678910:user/user-name-1”
        ]
    },
    "Action": "es:*",
    "Resource": "arn:aws:es:us-east-1:12345678910:domain/test/*”
}
```

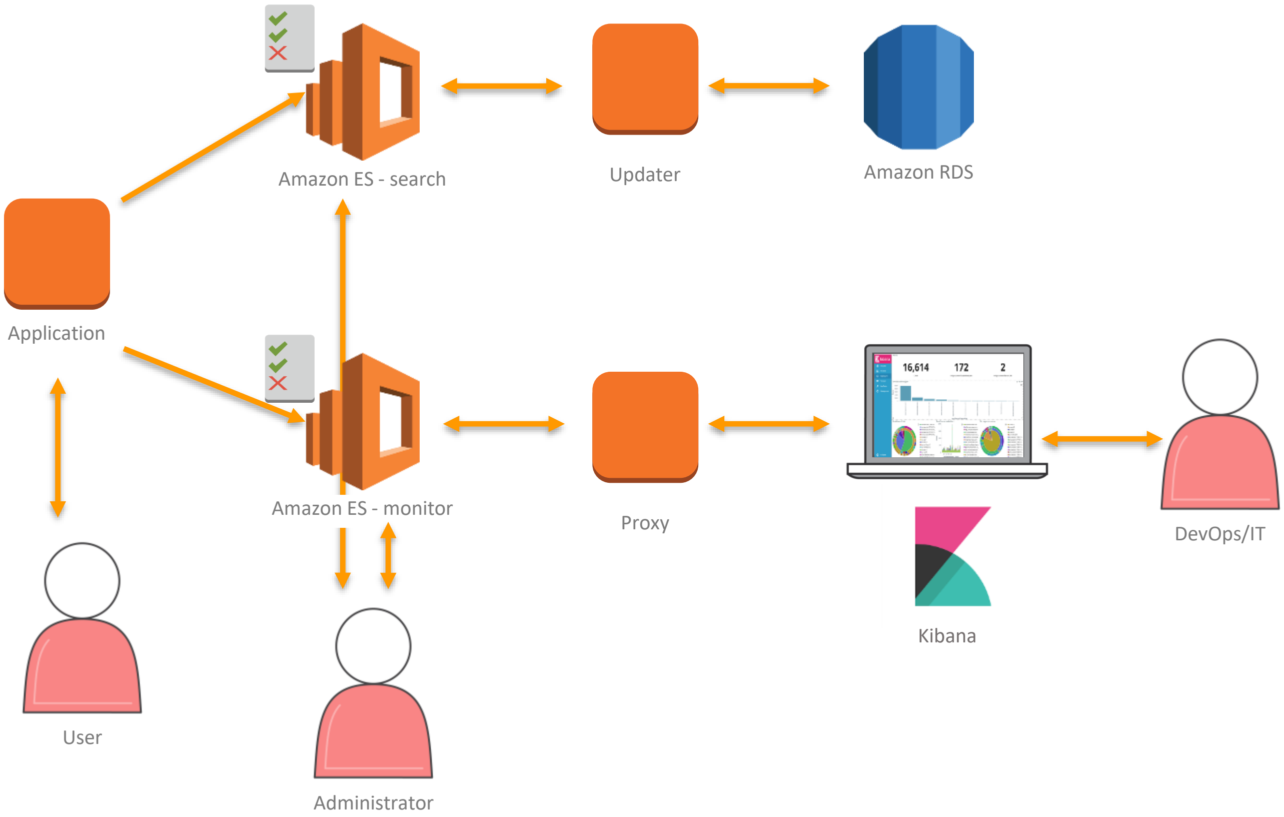
Policy 2: public access IP-based

- IP-based control, resource-based policy. All users, all Actions, all indexes

```
{
    "Sid": "",
    "Effect": "Allow",
    "Principal": { "AWS": "*” },
    "Action": "es:*",
    "Resource": "arn:aws:es:us-east-1:12345678910:domain/test/*”
    "Condition": {
        "IpAddress": {
            "aws:SourceIp": [”1.2.3.4/24"]
    } }
}
```

Differential Access Example

Application access



Different entities and access controls for applications

	Administrator	IT/DevOps	Application	Updater	Proxy
Type	User-based	User-based	Resource-based	Resource-based	Resource-based
Actions	• es:CreateElasticsearchDomain • es:Describe* • es:DeleteElasticsearchDomain • es:ListDomainNames • es:AddTags • es:ListTags • es:RemoveTags • es:Update*	• es:ESHttpGet • es:ESHttpPut • es:ESHttpDelete • es:ESHttpPost • es:Describe* • es:ListDomainNames • es:AddTags • es:ListTags • es:RemoveTags • es:Update*	es:ESHttpGet	es:ESHttpPost	es:ESHttpGet
Resources	Amazon ES search Amazon ES monitor	Amazon ES search Amazon ES monitor	Amazon ES search	Amazon ES search	Amazon ES Monitor
IPs	No	No	EIP for the application instance	EIP for the updater instance	EIP for the proxy

Policy 3: administrative access

- For AWS control actions, specify the principal and actions on a user-based policy

```
{
    ...
    "Principal": {"AWS: "arn:aws:iam::12345678910:user/admin”}
    "Action": [ "es:CreateElasticsearchDomain",
                "es:Describe*",
                "es:DeleteElasticsearchDomain",
                "es:ListDomainNames"
                "es:AddTags",
                "es:ListTags",
                "es:RemoveTags",
                "es:Update*"]
    "Resource": ["arn:aws:es:us-east-1:12345678910:domain/search/*”,
                 "arn:aws:es:us-east-1:12345678910:domain/monitor/*”]
}
```

Policy 4: DevOps

- For Devops, add the ES API calls

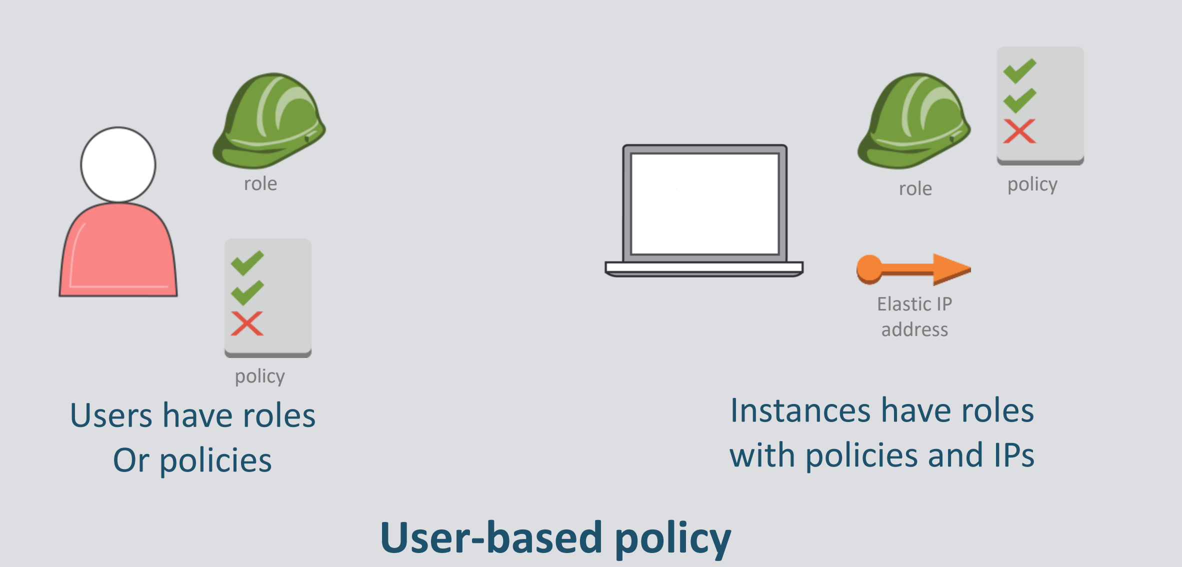
```
{
    ...
    "Principal": {"AWS: "arn:aws:iam::12345678910:user/admin”}
    "Action": [ ...
                "es": "ESHttp*" ]
    "Resource": ["arn:aws:es:us-east-1:12345678910:domain/search/*”,
                 "arn:aws:es:us-east-1:12345678910:domain/monitor/*”]
}
```

Policy 5: application access – read only

- For application and other restricted access, specialize the ESHttp methods and indexes allowed. For read-only

```
{
    "Sid": "",
    "Effect": "Allow",
    "Action": "es:ESHttpGet",
    "Resource": "arn:aws:es:us-east-1:12345678910:domain/search/*”
    "Condition": {
        "IpAddress": {
            "aws:SourceIp": [”1.2.3.4/24"]
    } }
}
```

IAM policy application and resolution



Users have roles
Or policies

Instances have roles
with policies and IPs

User-based policy

Amazon ES domains
have policies

Resource-based policy

- IAM authenticates based on all applicable identification and all policies are in play

Access Policy Application & Resolution

	Allowed in a resource-based policy	**Denied in a resource-based policy**	**Neither allowed nor denied in a resource-based policy**
Allowed in an identity-based policy	Allow	Deny	Allow
Denied in an identity-based policy	Deny	Deny	Deny
Neither allowed nor denied in an identity-based policy	Allow	Deny	Deny

- Deny ALWAYS wins over competing policy types
- If you do not explicitly state a policy, deny is default

VPC Access Control

Private endpoints can take advantage of security groups

Kibana with
proxy/bastion

Application access

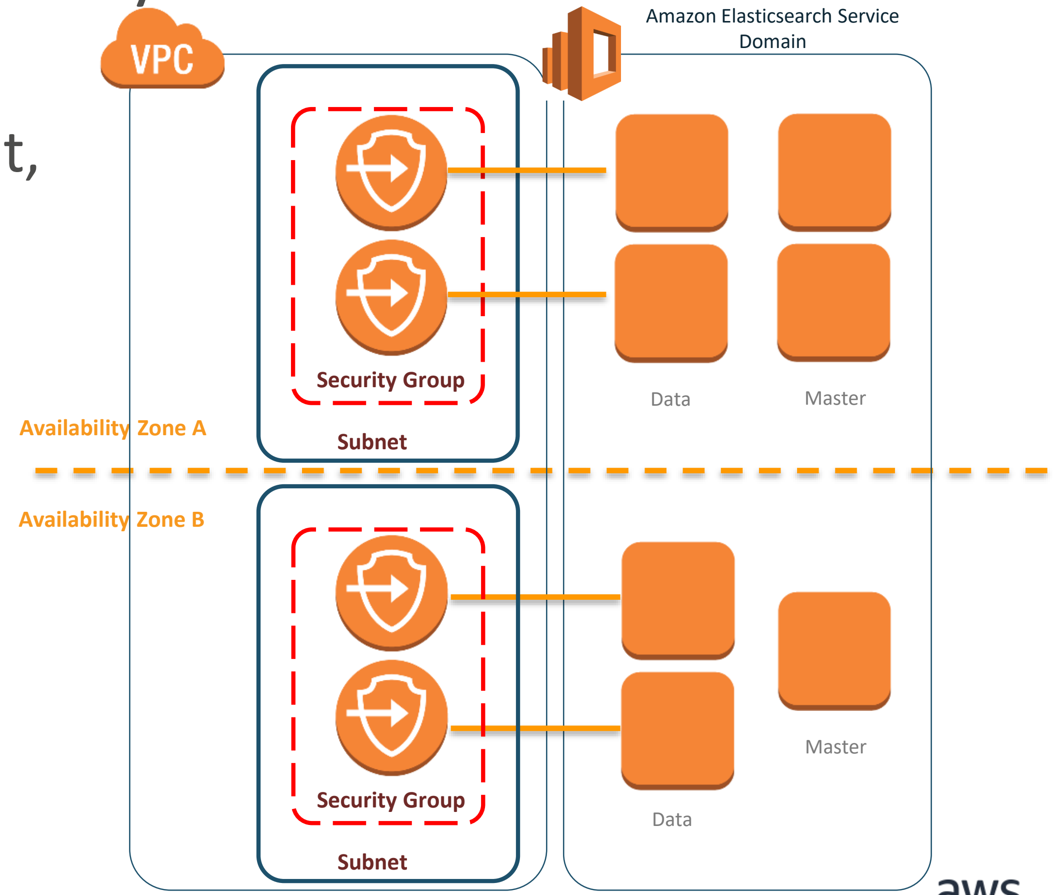
Log delivery

logstash

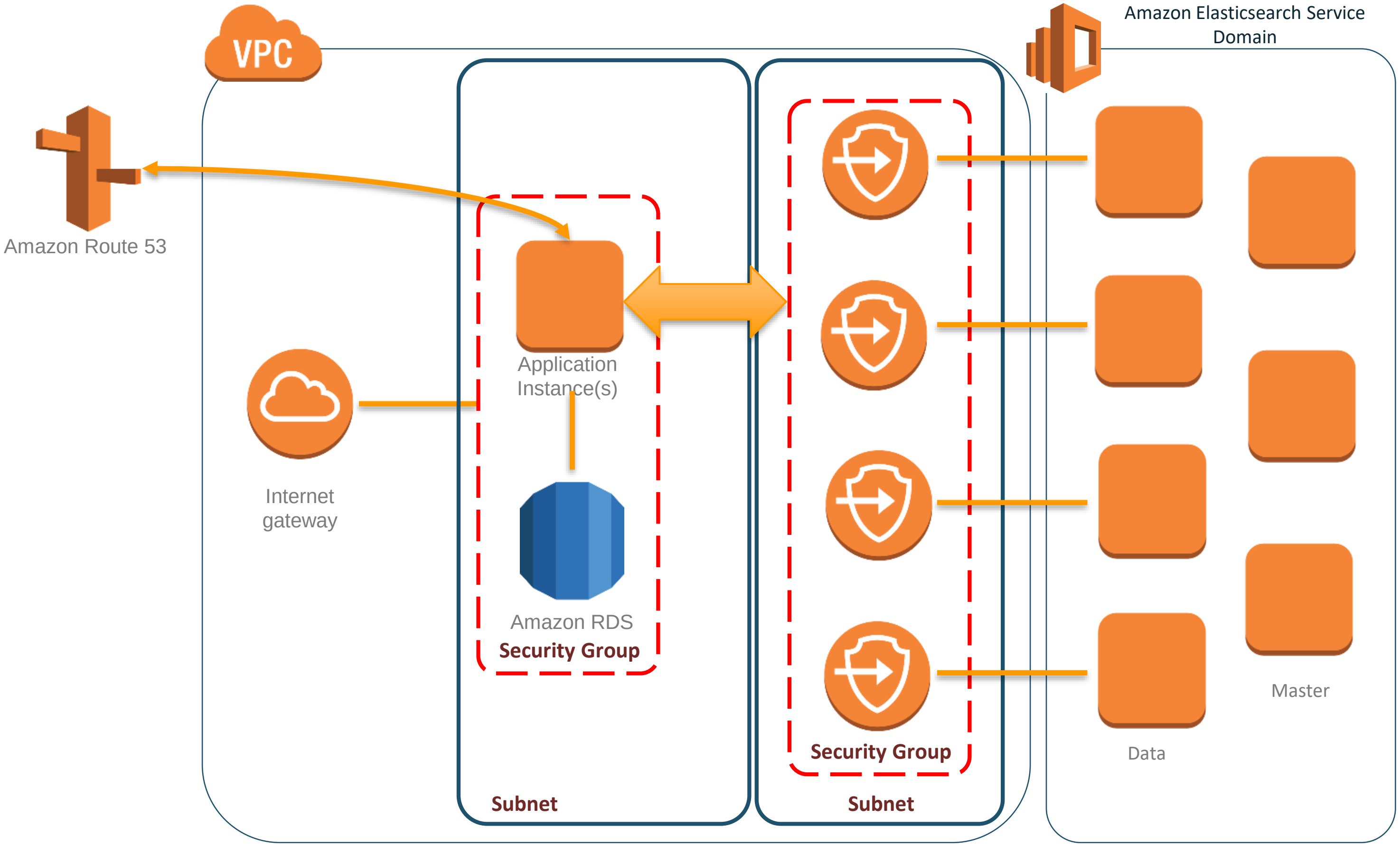
Log delivery

Amazon ES architecture in your VPC

- You still use an endpoint, Route 53 resolves IPs
- Elastic Network Interfaces (ENIs)
- Create a subnet for Amazon ES
- IAM policies applied
- Single- or multi-zone



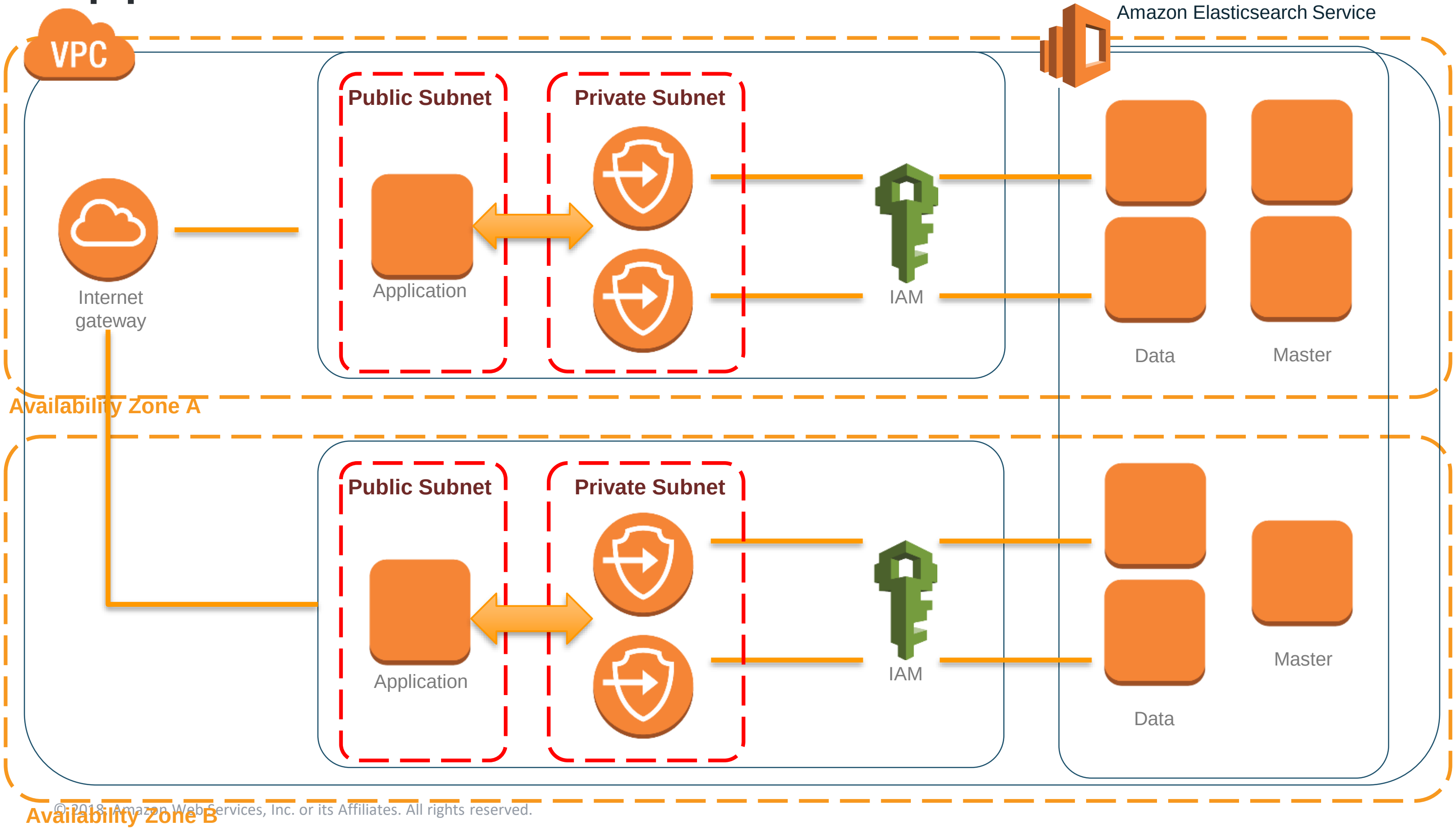
Single zone, single security group



Simple VPC access

- Internet gateway provides access for application users, search, and monitoring traffic within the subnet
- Security group has normal inbound/outbound rules
- Because the IPs are within the security group, SigV4 signing is not required

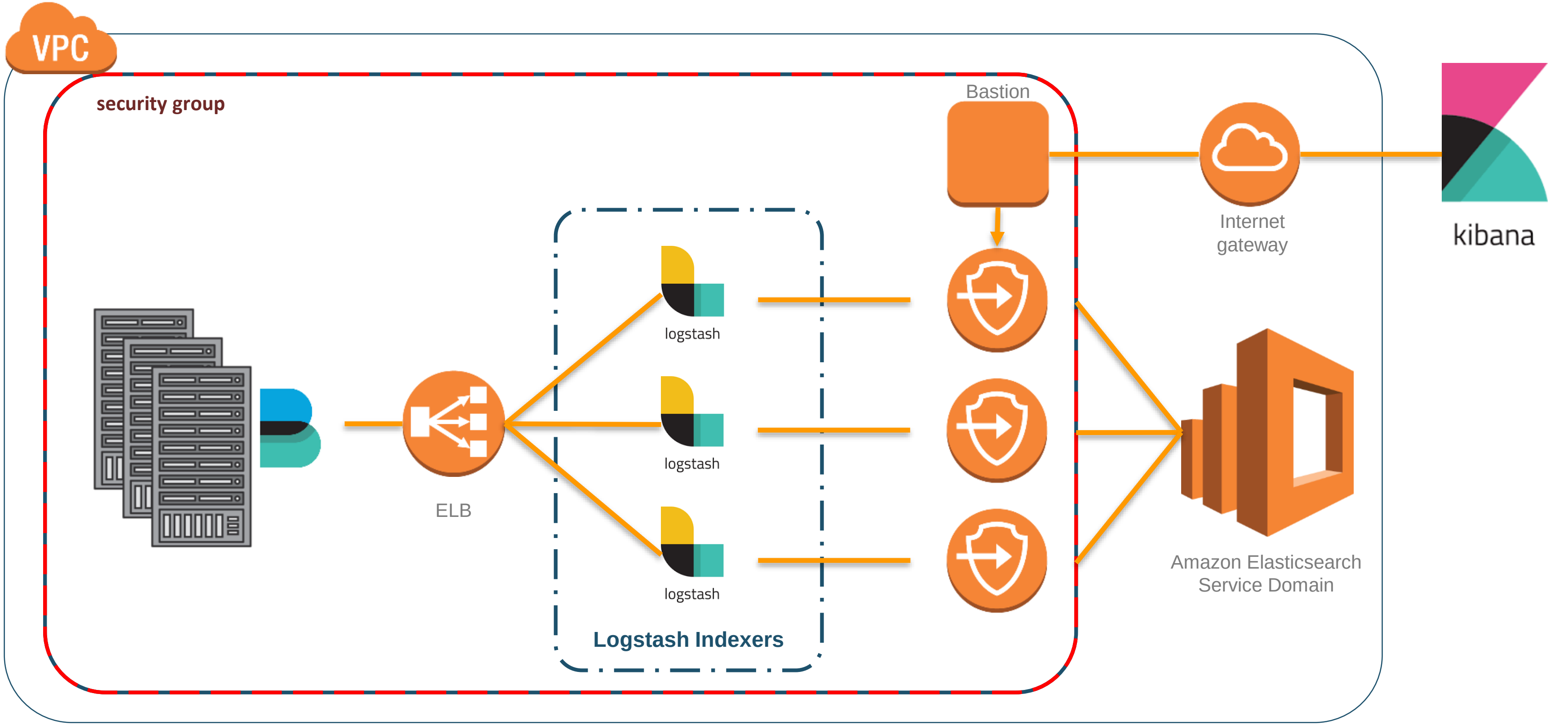
Application search within VPC



Application search within the VPC

- With Zone Awareness enabled, the domain is in 2 subnets
- IAM provides additional security for IP-based or signed requests

Logging infrastructure in your VPC

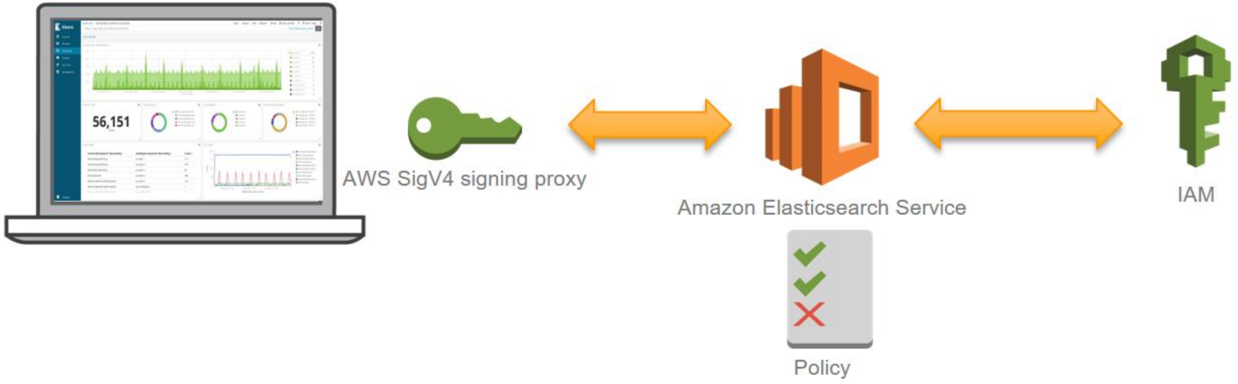


Logging infrastructure in your VPC

- Logstash colocated with the infrastructure you are monitoring
- Use an ELB across an autoscaled group of indexers to batch and forward to Amazon Elasticsearch Service
- Use a reverse proxy in the VPC to forward Kibana traffic to Amazon ES

Kibana Access Control

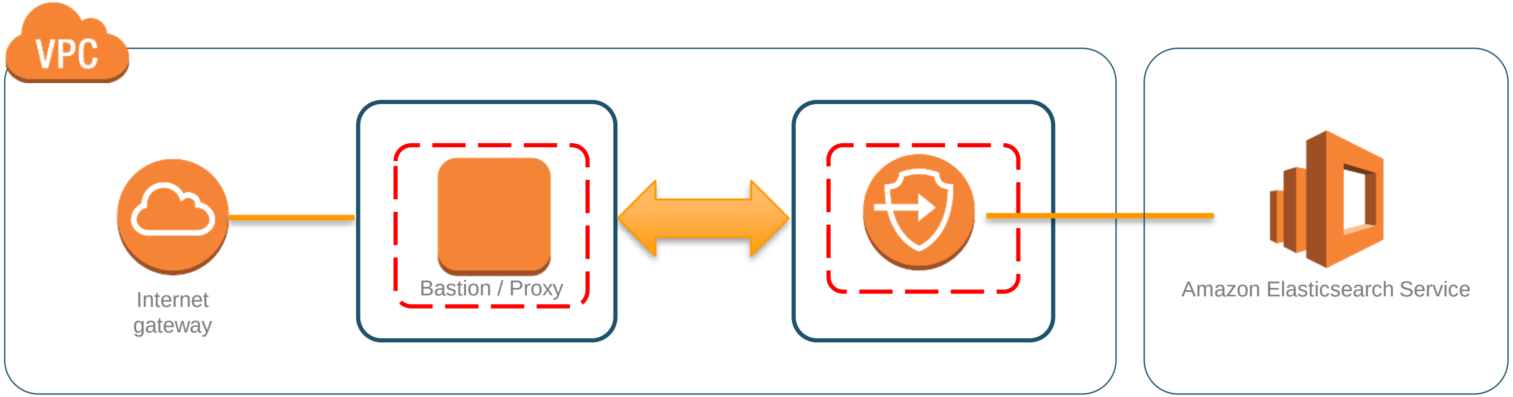
Ad-hoc



- This pattern is for public endpoints
- Run a signing proxy locally to sign all traffic
- Alternate: anonymous access via reverse proxy at a known IP address

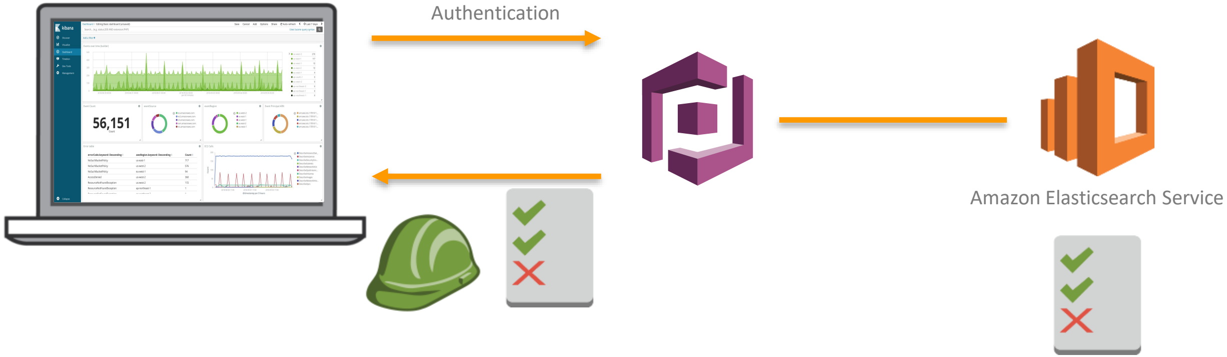
Example signing proxy: **http://tinyurl.com/y88fh3uq**

Kibana access for VPC



- This pattern is for VPC endpoints
- Range of choices for routing traffic to the Elasticsearch service domain
- Use an EIP or implement signing with a reverse proxy

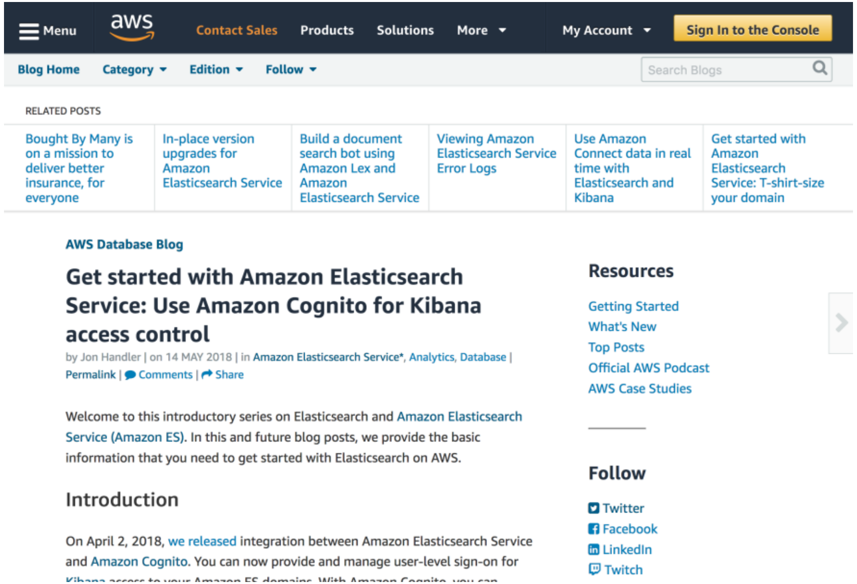
Use Amazon Cognito for Kibana sign-in



- Works for public and private endpoints
- Add the AuthUser to the domain's endpoint
- Create users and roles within Cognito to control access
- Supports federated identities
- Limitation: access control is per-domain

For more information on setting up Cognito

http://tinyurl.com/ydghxh84


Menu
aws
Contact Sales
Products
Solutions
More
My Account
Sign In to the Console
Blog Home
Category
Edition
Follow
Search Blogs
RELATED POSTS
Bought By Many is on a mission to deliver better insurance, for everyone
In-place version upgrades for Amazon Elasticsearch Service
Build a document search bot using Amazon Lex and Amazon Elasticsearch Service
Viewing Amazon Elasticsearch Service Error Logs
Use Amazon Connect data in real time with Elasticsearch and Kibana
Get started with Amazon Elasticsearch Service: T-shirt-size your domain
Resources
Getting Started
What's New
Top Posts
Official AWS Podcast
AWS Case Studies
Follow
Twitter
Facebook
LinkedIn
Twitch

AWS Database Blog

Get started with Amazon Elasticsearch Service: Use Amazon Cognito for Kibana access control

by Jon Handler | on 14 MAY 2018 | in Amazon Elasticsearch Service*, Analytics, Database | Permalink | Comments | Share

Welcome to this introductory series on Elasticsearch and Amazon Elasticsearch Service (Amazon ES). In this and future blog posts, we provide the basic information that you need to get started with Elasticsearch on AWS.

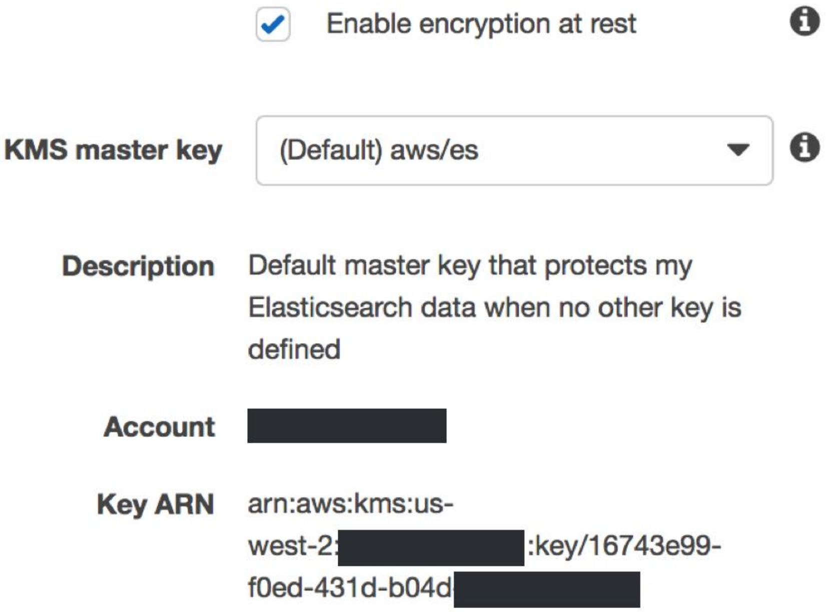
Introduction

On April 2, 2018, we released integration between Amazon Elasticsearch Service and Amazon Cognito. You can now provide and manage user-level sign-on for Kibana access to your Amazon ES domains. With Amazon Cognito, you can

Encryption at rest

Enable encryption at rest

- You can enable encryption at rest at domain creation
- Encrypts data for ephemeral and EBS storage
- Encrypts service (automatic) backups



☑ Enable encryption at rest

KMS master key (Default) aws/es

Description Default master key that protects my Elasticsearch data when no other key is defined

Account [redacted]

Key ARN arn:aws:kms:us-west-2[redacted]:key/16743e99-f0ed-431d-b04d[redacted]

Conclusions

- Amazon Elasticsearch Service provides many touchpoints for controlling access to your domain
- IAM policies are the backbone
- You can have public or private endpoints, control access via IP or signed requests, and use Cognito for Kibana sign-in

Amazon Elasticsearch Service

Find out more:

https://aws.amazon.com/elasticsearch-service/

AWS Centralized Logging:

https://aws.amazon.com/answers/logging/centralized-logging/

Elasticsearch at the AWS Database Blog:

https://aws.amazon.com/blogs/database/category/elasticsearch/

Or ask your Solutions Architect!

Thank you!