DataSunrise

aws partner network

Advanced Technology Partner

Security Competency
Amazon RDS
Amazon Redshift
Amazon Linux 2
Microsoft Workloads

Configuring DataSunrise with CloudFormation script on Amazon AWS

Instruction Manual

January, 2022

Table of Contents

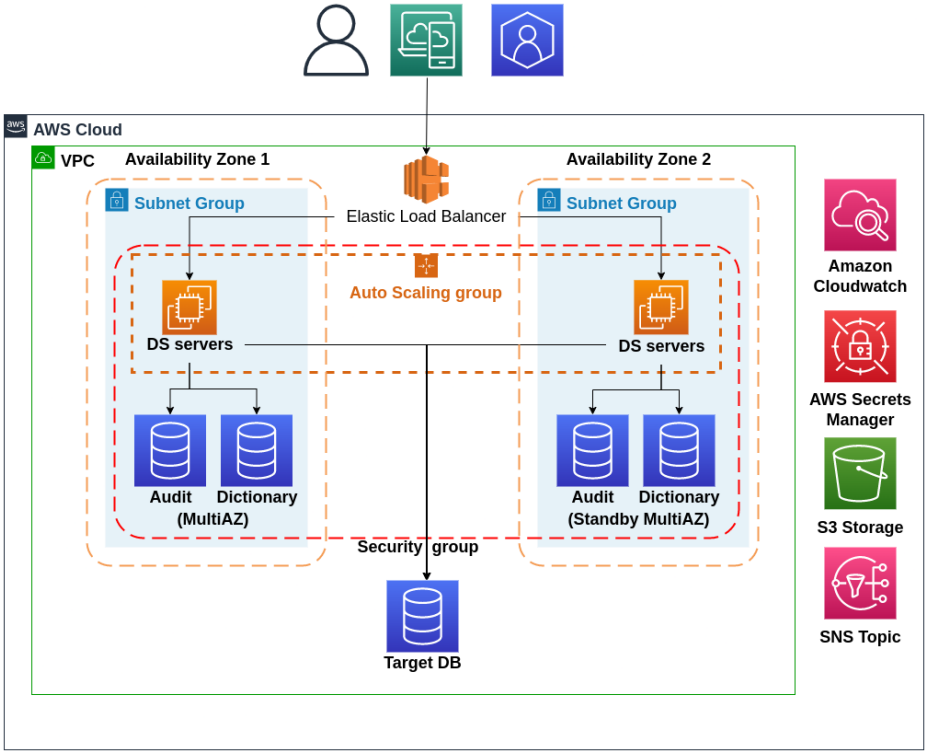
1. Introduction

Manual deployment of a High Availability configuration within any kind of environment presumes to consider a lot of different aspects to be kept in mind. Even though a deployment process is successfully finished, there are additional actions to be done for a flawless and stable configuration to be created.
To avoid all of the possible issues and skip manual adjusting, DataSunrise provides a dedicated script for HA environment (infrastructure) deployment within the Amazon Web Service based on Amazon Cloud Formation Service. The deployment process is automated and does not require manual Amazon Web Service resources creation.

Estimated deployment time: 10-20 minutes.
Required skills: basic knowledge of AWS CloudFormation (how to create/update/delete Stacks), AWS S3 (create bucket, add objects via AWS CLI/Management Console)

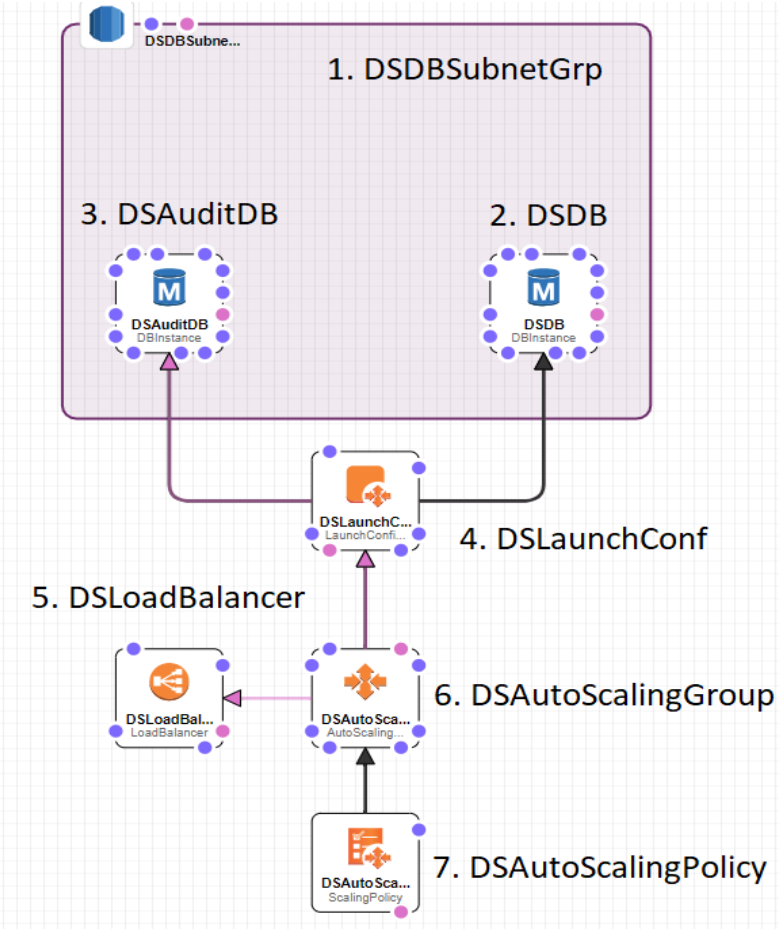
The following diagram demonstrates the architectural overview of the DataSunrise deployment. The setup uses S3 as the build installer and backups storage, CloudWatch as a resilient log store and Secrets Manager for keeping sensitive data used in the deployment.

1.1



Description and structural scheme of a CloudFormation stack

The following picture displays the most important objects created by Cloud Formation as they are lsted in the script, creation order is parallel, considering objects' dependencies (marked with arrows).



1. DSDBSubnetGrp (AWS::RDS::DBSubnetGroup) - RDS security group. Restricts access to the Dictionary and Audit Storage databases;

2. DSDictionaryDB (AWS::RDS::DBInstance) - RDS database instance used to store DataSunrise settings (Dictionary). The EBS of the RDS instance is encrypted by default.

3. DSAuditDB (AWS::RDS::DBInstance) - RDS database instance used to store DataSunrise's audit journal and other journals (Audit Storage). The EBS of the RDS instance is encrypted by default;

4. DSLaunchConf (AWS::AutoScaling::LaunchConfiguration) - Virtual machine configuration, includes operating system's AMI, VM instance type, DataSunrise installation script;

5. DSLoadBalancer (AWS::ElasticLoadBalancing::LoadBalancer) - Load Balancer;

6. DSAutoScalingGroup (AWS::AutoScaling::AutoScalingGroup) - Auto scaling group, includes configuration of the failover cluster;

7. DSAutoScalingPolicy (AWS::AutoScaling::ScalingPolicy) - Auto scaling policy, defines the way of deployment of new VMs of the failover cluster.

Note: There are different dependent resources that will be created as well. Describing them here will not bring any useful information.

DataSunrise CloudFormation uses Secrets Manager secrets to store the following sensitive data:

- Target Database User Login – required to onboard a database to DataSunrise configuration and perform tasks like updating database metadata, running Data Discovery, Static Masking and other features requiring direct database connection
- Audit and Dictionary Database User Login – DB user IDs used by DataSunrise to access Audit and Dictionary (Configuration) Storage Databases
- DataSunrise WEB-UI Admin password
- DataSunrise BYOL (if used) License Key

1.2 Supported Regions

The supported Regions list is determined by the Amazon Machine Images (AMI) leveraged by the CloudFormation Template and covers the major regions:

Region Name	Region
Europe (Frankfurt)	eu-central-1
Europe (Stockholm)	eu-north-1
Asia Pacific (Mumbai)	ap-south-1
Europe (Paris)	eu-west-3
Europe (London)	eu-west-2
Europe (Ireland)	eu-west-1
Asia Pacific (Seoul)	ap-northeast-2
Middle East (Bahrain)	me-south-1
Asia Pacific (Tokyo)	ap-northeast-1
South America (São Paulo)	sa-east-1
Canada (Central)	ca-central-1
Asia Pacific (Hong Kong)	ap-east-1
Asia Pacific (Singapore)	ap-southeast-1
Asia Pacific (Sydney)	ap-southeast-2

US East (N. Virginia)	us-east-1
US East (Ohio)	us-east-2
US West (N.California)	us-west-1
US West (Oregon)	us-west-2

1.3 Licensing and costs

You are responsible for the cost of the AWS services used while running this CloudFormation script. There is no additional cost for using this template. The AWS CloudFormation template for DataSunrise includes configuration parameters that you can customize. Some of these settings, such as instance type, will affect the cost of deployment. For cost estimates, see the pricing pages for each AWS service you will be using. Prices are subject to change.

There are two license types you can leverage with AWS DataSunrise deployments: Hourly Billing (HB) and Bring Your Own License (BYOL). With HB license, DataSunrise sends units (usage data) to Amazon Marketplace hourly for later billing. The cost scales with the number of Database instances added to DataSunrise Configuration. Hourly Billing is perpetual and doesn't require renewal like BYOL does.

The BYOL license is active for the period set by the trial key and applicable for the list of Database hosts and instances (MSSQL and Oracle). For pricing and quote details contact DataSunrise Sales team (sales@datasunrise.com)

1.4 AWS Services Limits considerations

During the deployment of CloudFormation template in a Region actively using EC2 Service, you might encounter a vCPU Quota Limit for On-Demand instances in your AWS account.

1.5 Security and Least Privilege Design

DataSunrise EC2 instances provisioned by CloudFormation use least privilege IAM policies to access used AWS Services (AWS S3, CloudWatch, Secrets Manager). Do not deploy CloudFormation template as AWS Account root user. Use least-privileged IAM User and a Service-linked roles for CloudFormation to avoid giving excessive permissions to the deploying user. The Guide provides the IAM Policy which can be created for CloudFormation service-linked Role to deploy the template. See Appendix for the IAM Policy listing.

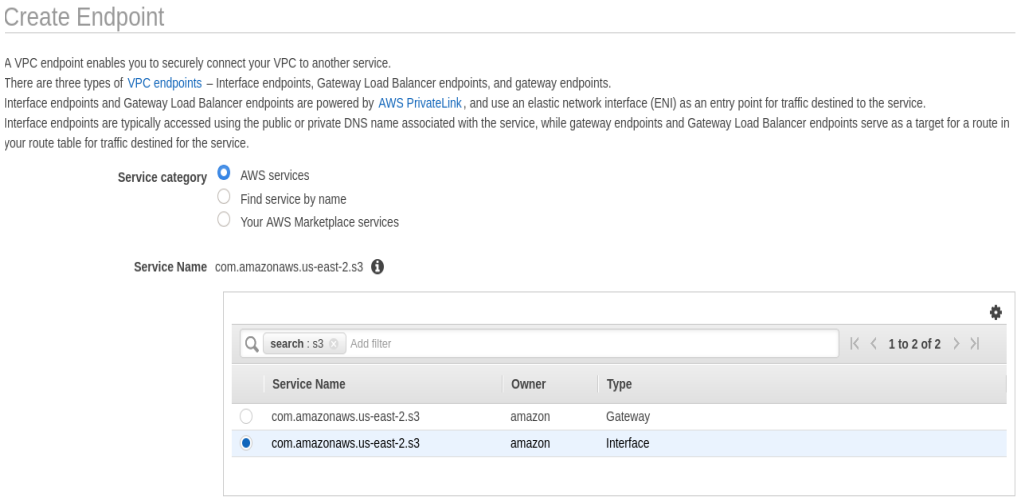
Please refer to EC2 Service Documentation https://aws.amazon.com/ec2/faqs/#EC2_On-Demand_Instance_limits on checking and expanding quota for your AWS account if it is required to fulfill the deployment.

2. Prerequisites

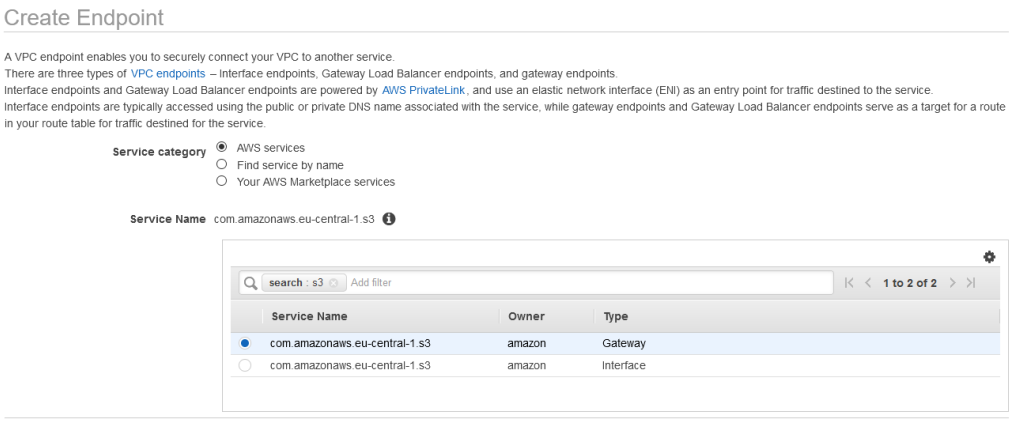
First, visit the following page and click **Continue to Subscribe**:
https://aws.amazon.com/marketplace/pp/B07P5JWLJ4?qid=1618471664103&sr=0-4&ref_=srh_res_product_title

There are also some entities that should exist in your AWS before you start the deployment process. Here is the list of required items:

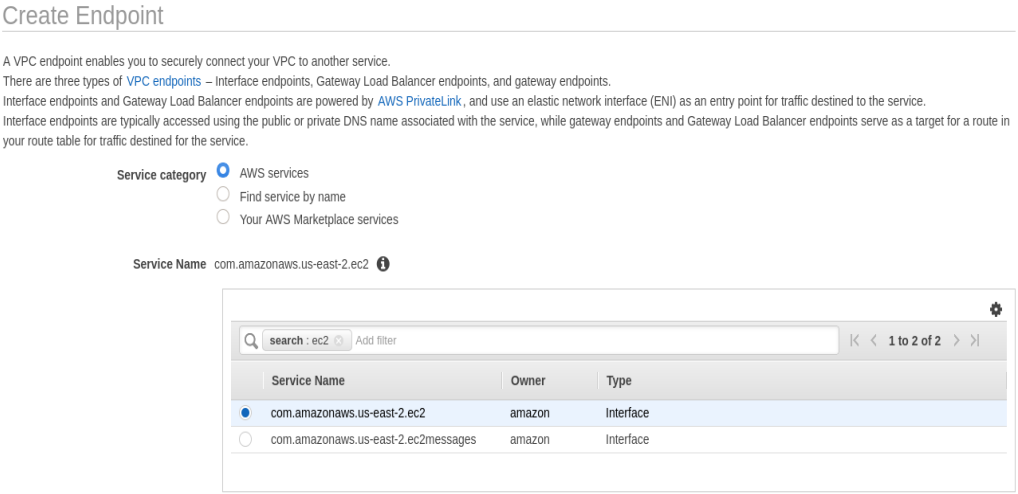
- AWS Account that will be used as the environment where the CloudFormation stack will be deployed;
- Amazon Virtual Private Cloud (VPC) that will be used as the environment where the CloudFormation stack will be deployed;
- Subnets within the VPC that you will designate for your DataSunrise cluster running in the specified VPC;
- Subnets to be used should have the following resource endpoints attached:
    - S3 Endpoint (com.amazonaws.<region>.s3): required in case S3 bucket is used as the DataSunrise Suite distribution source;
      Two endpoint types: interface and gateway are to be attached;



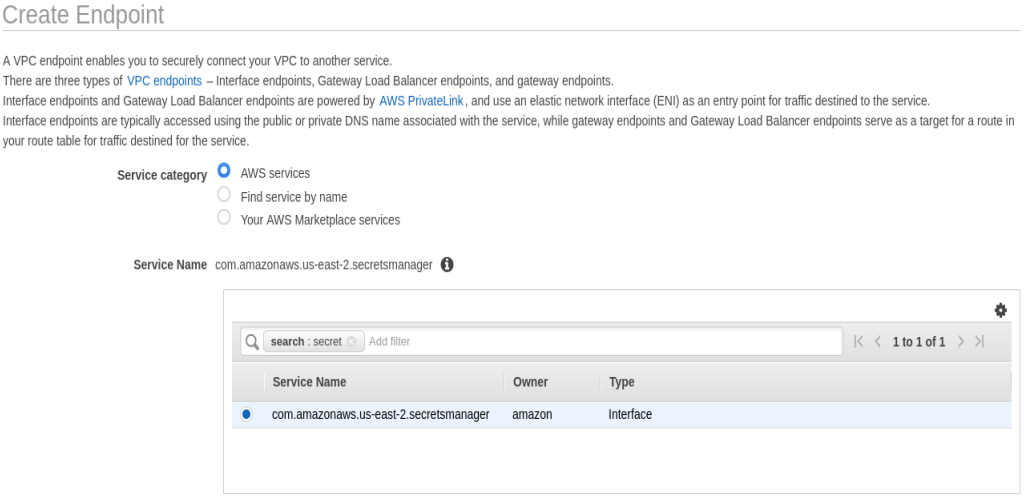
      Gateway type endpoint should be attached to VPC with private subnets;



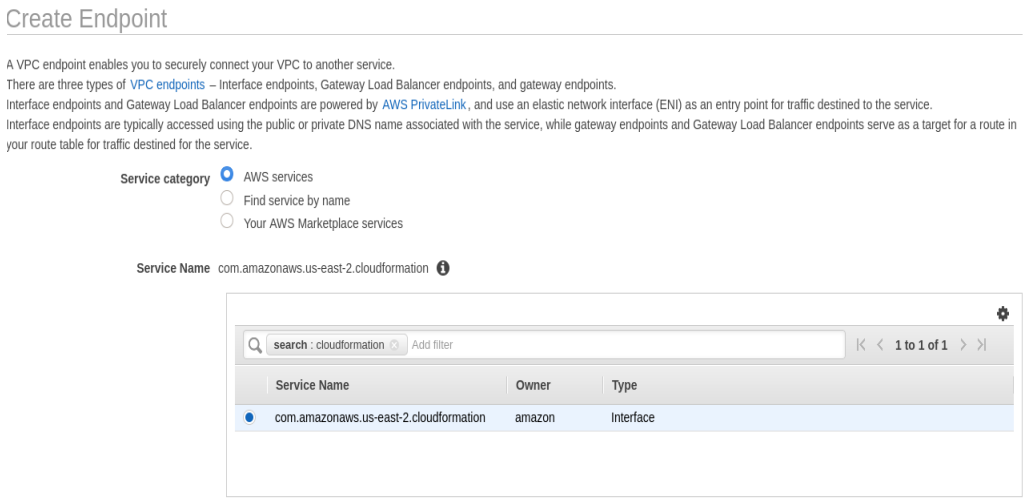
- EC2 Endpoint (com.amazonaws.<region>.ec2): required for Launch Configuration access within the deployment process;



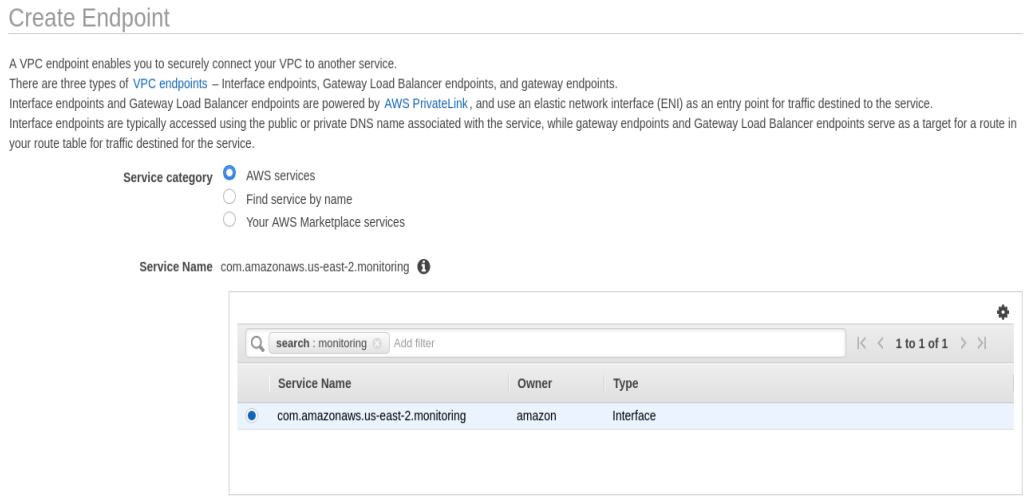
- Secrets Manager Endpoint (com.amazonaws.<region>.secretsmanager): required for obtaining DataSunrise administrator password from the Secrets Manager;



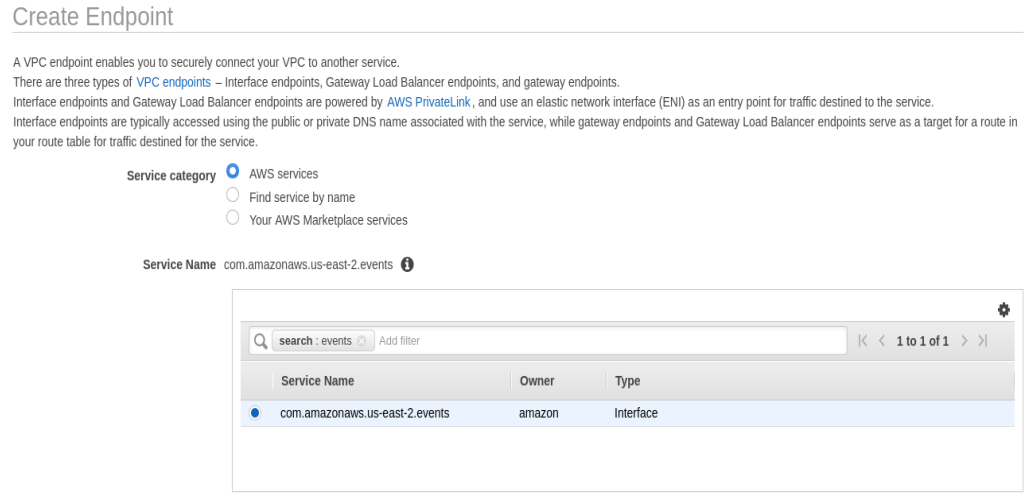
- Cloud Formation Endpoint (com.amazonaws.<region>.cloudformation): required for Cloud Initialization;



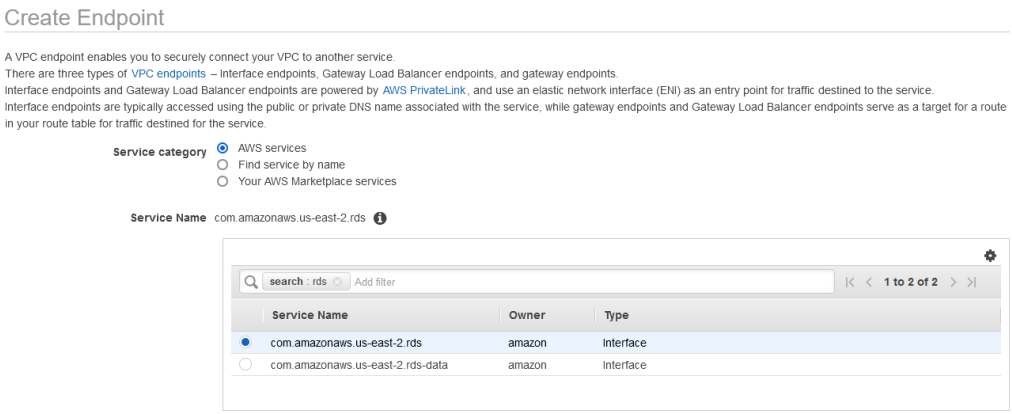
- Cloud Watch Endpoint (com.amazonaws.<region>.monitoring): required for Cloud Watch synchronization and DataSunrise Suite Log Files upload;



- Events Endpoint (com.amazonaws.<region>.events): required for placing DataSunrise events to DataSunrise Cloud Watch namespace;



- RDS Endpoint (com.amazonaws.<region>.rds): required for interaction with RDS in private subnets;



- NAT Gateway attached to a subnet is required in case DataSunrise is deployed in a private subnet and the Hourly Billing license type is used. Basically, in order to use Hourly Billing license type, DataSunrise should be able to access these endpoints (basing on the used region): https://docs.aws.amazon.com/general/latest/gr/aws-marketplace.html

Important:

The endpoints presented in the list should have private DNS names enabled.

- Virtual Machine Key Pair that will be used to connect to your EC2 machines via SSH;
- Target Database credentials and information required for establishing a connection (hostname, port, DB name, login, password);
- (Optional) S3 bucket to store various output information from DataSunrise.

Important:

In case you are going to create a separate account for the Cloud Formation deployment, we suggest keeping your environment based on least privileges concept and use a prepared policy for a newly-created dedicated AWS Account. The policy contains minimal privileges for successful deployment. Please refer to the Appendix section for *MinimalAccountPolicy.json*.

2.1 Public Deployment

To publicly deploy a Cloud Formation stack, you will need to choose two public subnets from the same VPC but from different Availability Zones. An Internet Gateway attached to the VPC is also required.

For each subnet, you must verify that its route table (could be one shared route table or one per subnet) defines a route pointing the destination "0.0.0.0/0" to the Internet Gateway.

When deploying a Cloud Formation stack, you will need to choose Elastic load balancer working mode — select "internet-facing".

2.2 Private Deployment

To privately deploy a Cloud Formation stack, you will need to choose two private subnets from the same VPC but from different Availability Zones. A NAT Gateway attached to the VPC is also required.

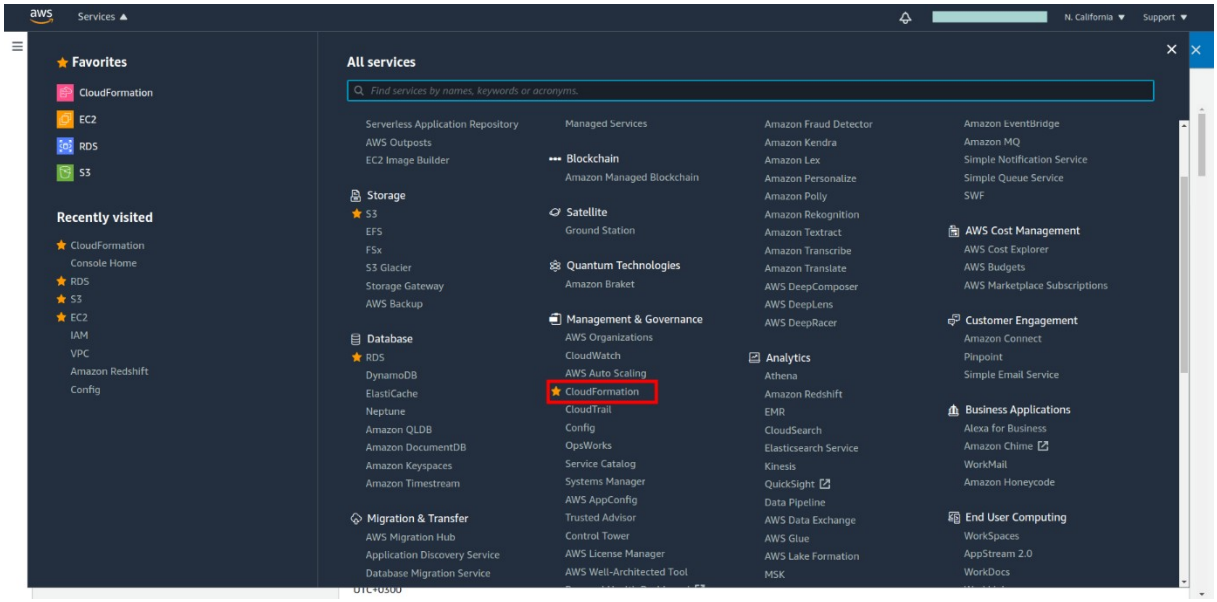
For each subnet, you should verify that its route table (one shared route table or one per subnet) defines a route pointing the destination "0.0.0.0/0" to the NAT Gateway.

When deploying a Cloud Formation stack, you will need to use the "internal" Elastic load balancer mode.

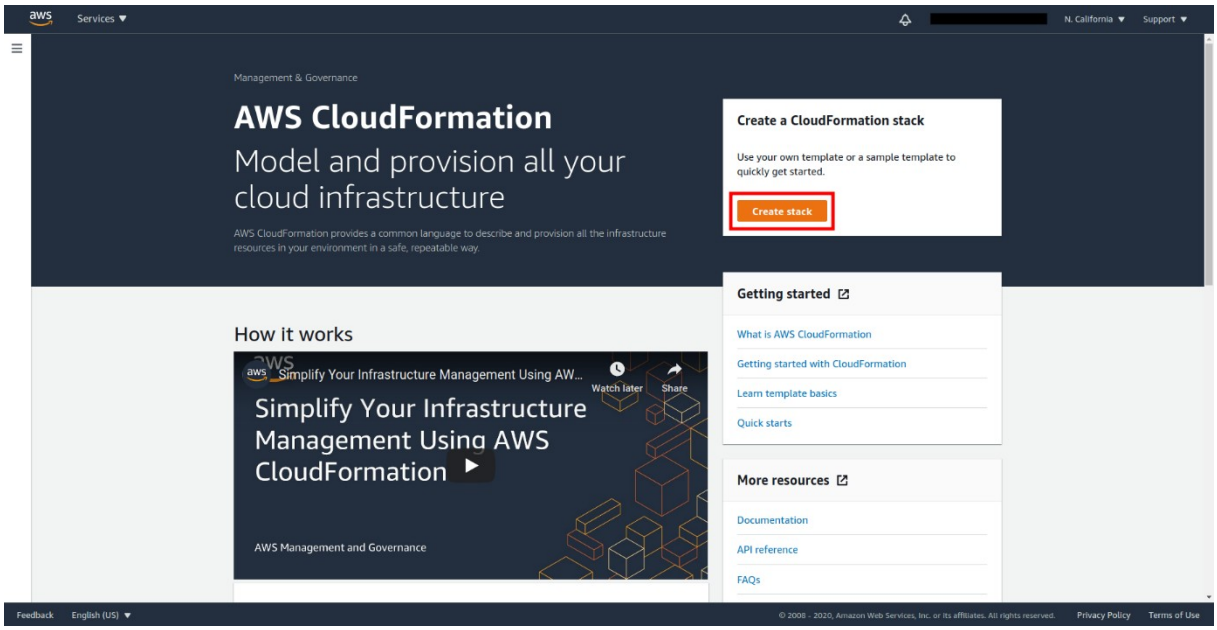
You can also use HTTP Proxy in case NAT Gateway attachment is not possible.

3. Deploying a CloudFormation stack

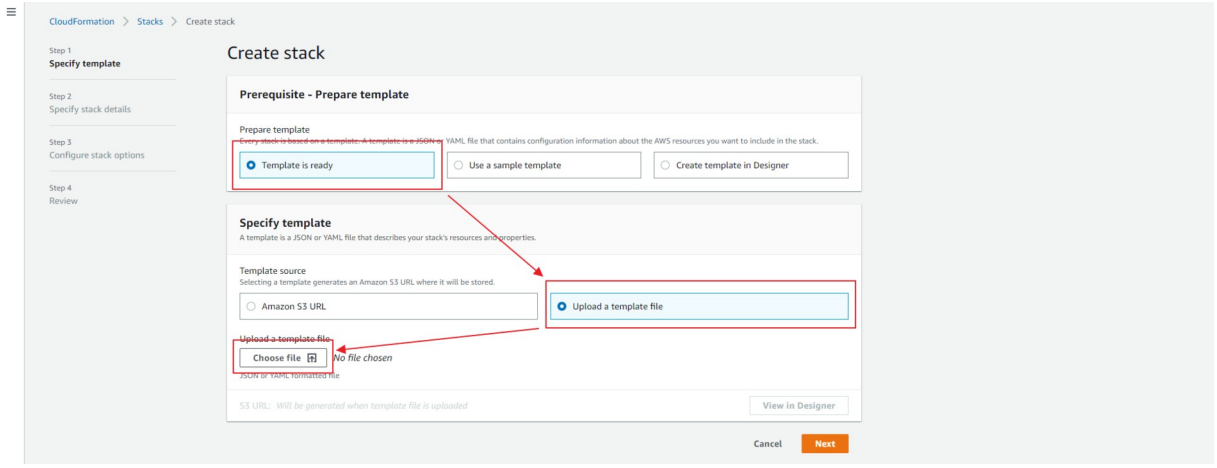
To deploy DataSunrise in HA configuration, navigate to the **Cloud Formation** subsection of AWS Console services.



Click **Create stack**:



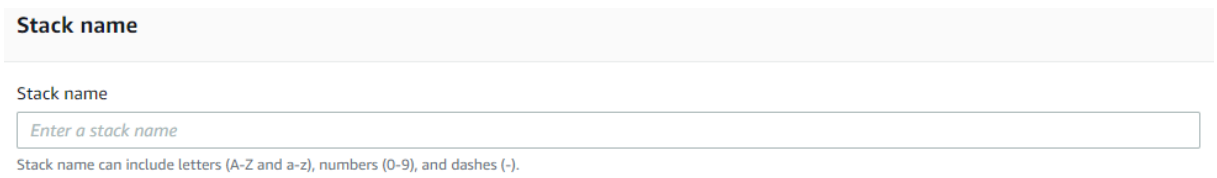
Select your Cloud Formation script and click **Next**:



3.1. Stack Name and Deployment Name

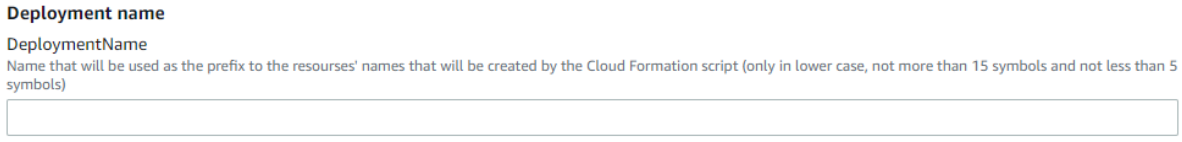
Click **Next** and you will be directed to the CloudFormation script's settings. You will see different fields. Some of them are already filled in with default values and some of them are empty. In this instruction, we will dwell on each field.

The first thing that should be done during the deployment process is naming the stack:



You can use any name here. This name will be used only to identify the stack during the deployment process and for you to differ it from other deployed stacks.

The next name needs to be filled is the Deployment name:



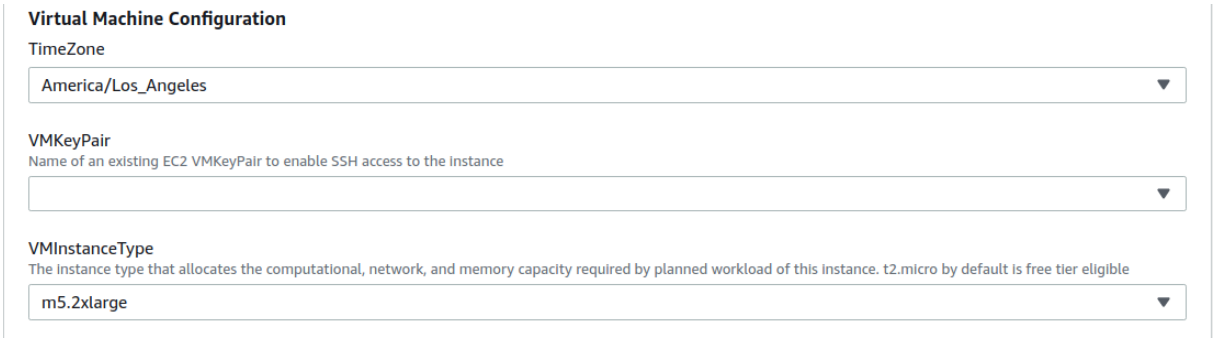
This name features a restriction on symbols that can be used to specify the name. Since this name is going to be used as a name prefix for all the objects that will be created automatically by the Cloud Formation script, it should be restricted to suit every object's name policies. The restrictions are: lower case, not more than 15 symbols and not less than 5 symbols.

Note: we recommend to specify the same name in both fields in the lower case to avoid possible issues.

3.2 Stack Parameters

Basically, CloudFormation script's parameters are divided into 7 sections that should be configured. In this instruction, we will take a look at them one by one.

1. Virtual Machine Configuration Section



This section contains 3 fields:

- **TimeZone** – preferred time zone of your EC2 machine.
- **VMKeyPair** – a key pair that will be used to access your EC2 machine via SSH.
- **VMInstanceType** – a type of machine that will be used to deploy DataSunrise on.

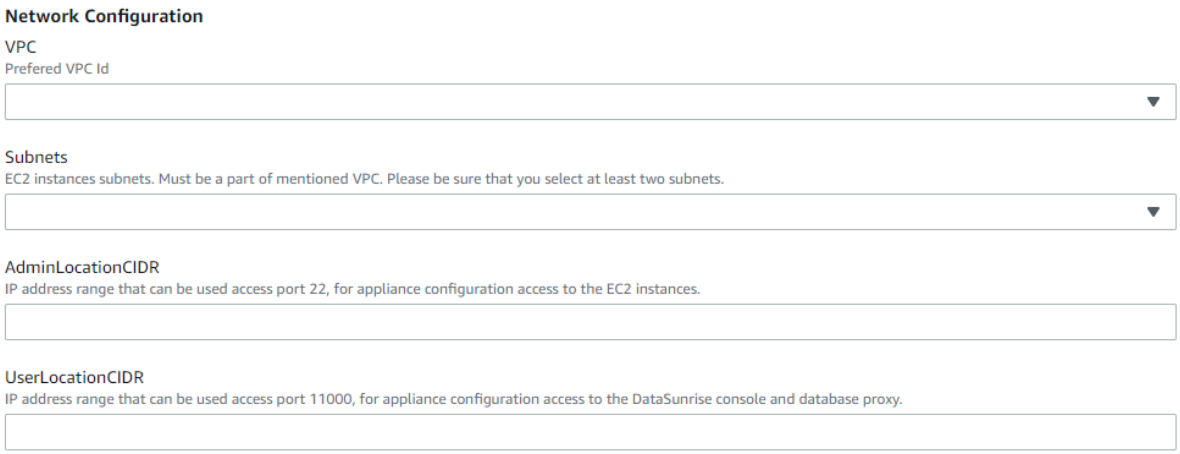
DataSunrise's High Availability and Auto Scale offerings:

- m5.8xlarge: 24500 operations/sec
- m5.4xlarge: 18700 operations/sec
- m5.2xlarge: 15350 operations/sec
- m5.xlarge: 7800 operations/sec
- m5.large: 3900 operations/sec

While these calculations are assessed using median environmental workloads in different configuration, they can be used as a realistic baseline based on important aspects such as database protocol and usual activities in your environment.

Note: you can choose other instance types like M5 [M5 instances feature the Intel Xeon Platinum 8000 series (Skylake-SP) processor with a sustained all core Turbo CPU clock speed of up to 3.1 GHz, and deliver up to 20% improvement in price/performance compared to M4 instances]. M5 instances provide support for the new Intel Advanced Vector Extensions 512 (AVX-512) instruction set, offering up to 2x the FLOPS per core compared to the previous generation M4 instances).

2. Network Configuration Section



This section contains 4 fields:

- **VPC** – a VPC to be used as environmental.
- **Subnets** – Subnets. Choose at least two subnets to get access from different zones.

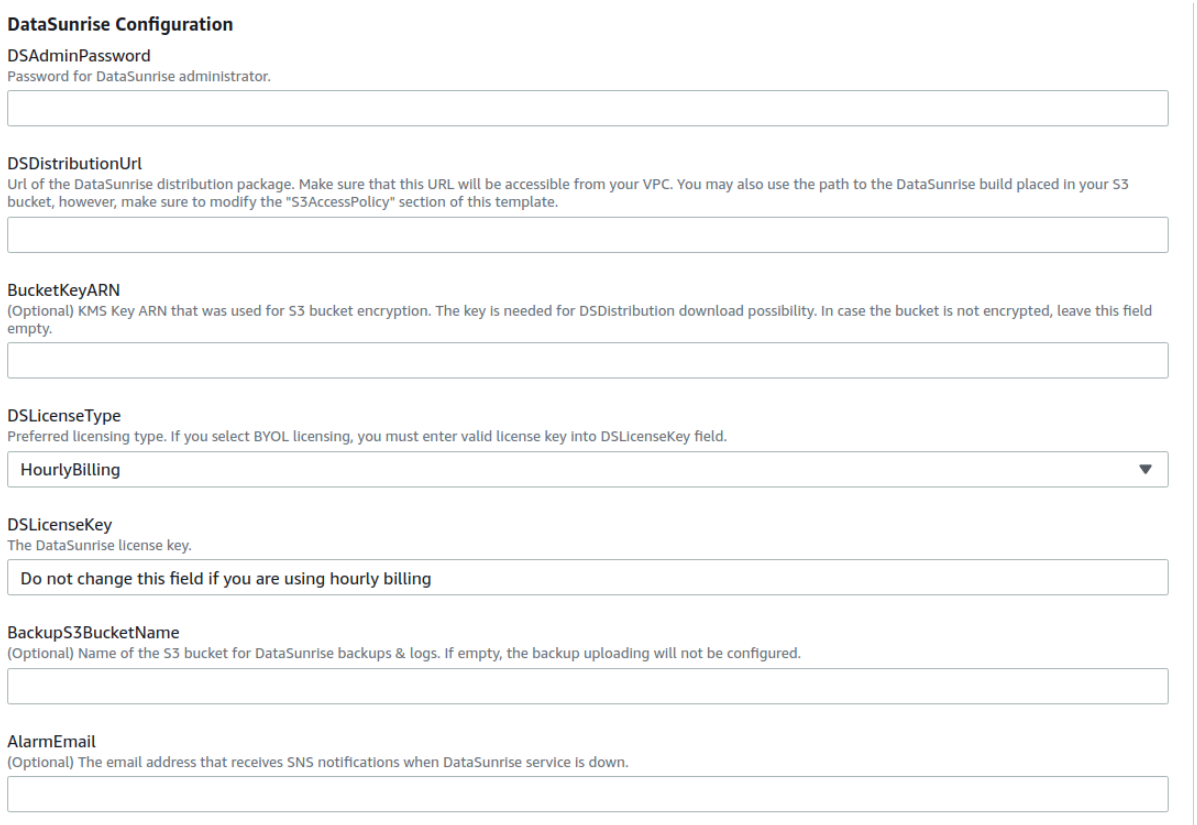
- **AdminLocationCIDR** - Classless Inter-Domain Routing address. By filling in this field, you determine the address from which the Administrator can access port 22 of the EC2 machine DataSunrise is installed on.

  Example: 172.32.0.0/16 to access EC2 machine port 22 from the default VPC IP-range.

- **UserLocationCIDR** – the address from which a user can access the DataSunrise Console and DataSunrise proxy.

  Example: 172.32.0.0/16 to be able to connect to the DataSunrise Proxy or Console from the default VPC IP-range.

3. DataSunrise Configuration Section



This section contains 6 fields:

- **DSAdminPassword** – specify any password here. This password will be used to connect to the DataSunrise Web Console.

  Note: remember this password or store it in some safe place.

- **DSDistributionUrl** – specify the download link for the DataSunrise installation package. It does not matter from what hostname it will be downloaded. This link should be available according to outbound rules of your VPC. You may also use the path to a DataSunrise build placed in your S3 bucket, however make sure to modify the "S3AccessPolicy" section of this template.

  Note: we recommend you to make sure the link provides an access to the installation package successfully and to add the specified link to an accessible hostnames list of your VPC.

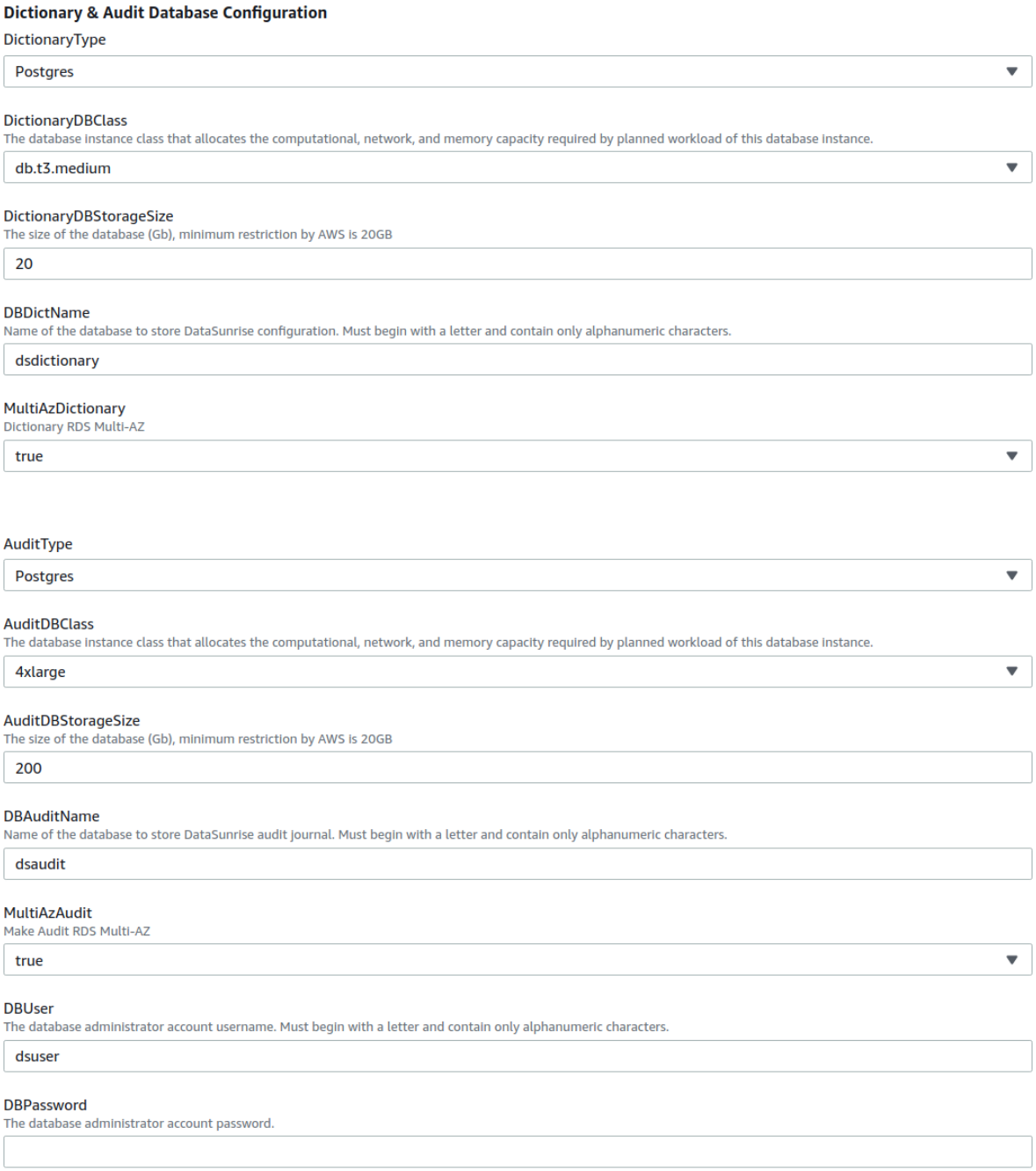
  The best way is to create an S3 bucket and modify the "S3AccessPolicy" section of the script to allow privileges to read from the bucket you created.

- **BucketKeyARN** – in case the DataSunrise Distribution will be placed in the encrypted bucket, provide KMS Key ARN in this field.
- **DSLicenseType** – choose one of two options for license usage. You can select either Hourly Billing type of license or in case you have a license key already, select BYOL.
- **DSLicenseKey** – in case BYOL license type is chosen, place the key in this field.
- **BackupS3BucketName (Optional)** – specify the name of the bucket within the region you are deploying the CloudFormation script in. This bucket will be used as a storage for keeping your DataSunrise's useful data such as log files, audit overdue information archiving, DataSunrise configuration backup, etc.

  Note: we recommend to create an S3 bucket and use it as a storage for such information. This might be helpful in case of facing any issues.

- **AlarmEmail (Optional)** – specify an email address here to receive SNS notifications at when DataSunrise service is down.

4. Dictionary and Audit Database Configuration Section



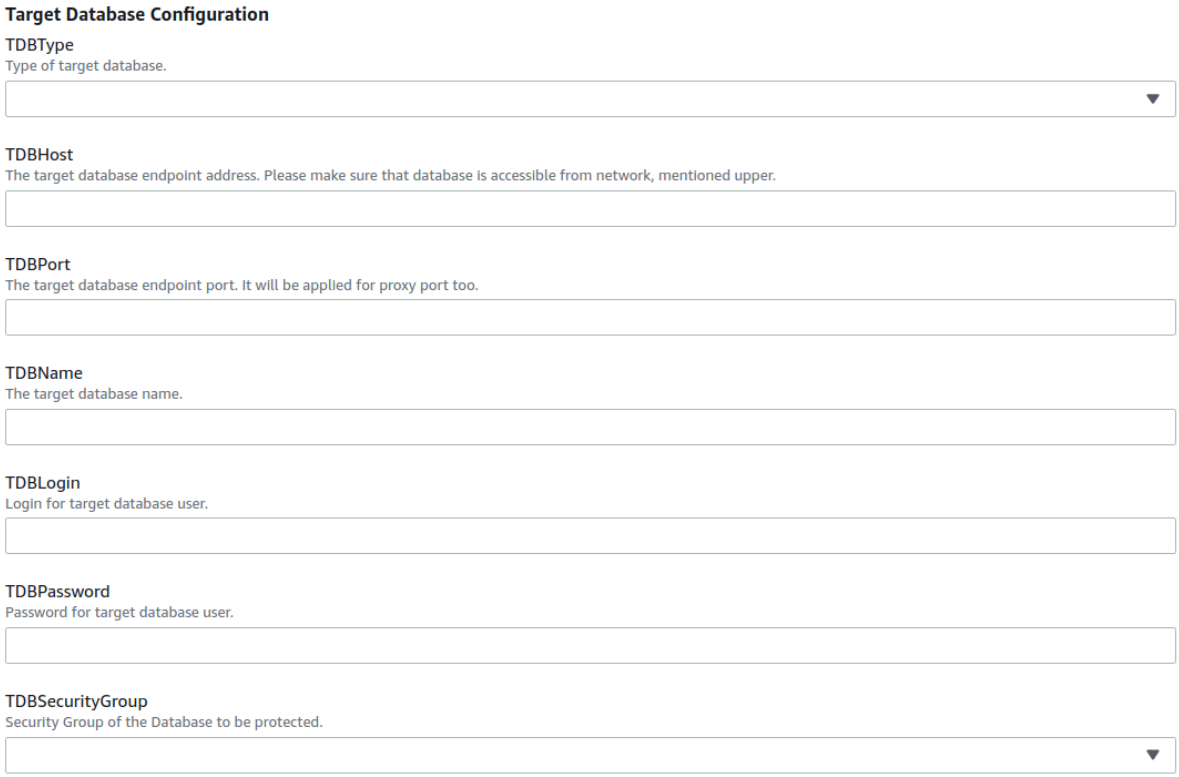
It contains Dictionary and Audit Storage databases' settings. These databases will be created by the Cloud Formation script:

- **DictionaryType** – choose the type of Dictionary Database among 3 available values that suits your requirements.
- **DictionaryDBClass** – specify the type of the machine the Dictionary Storage will be deployed on.
- **DictionaryDBStorageSize** – specify the storage size of the Dictionary database.

- **DBDictName** - specify the name of internal PostgreSQL RDS database. It will be used to establish a connection between your DataSunrise and the Dictionary.
- **MultiAZDictionary** – choose true in case it is required dictionary db to be available from multiple Availability Zones.
- **AuditType** – choose the type of Audit Database among 3 available values that suits your requirements.
- **AuditDBClass** – specify the type of the machine the Audit Storage will be deployed on.
- **AuditDBStorageSize** – specify the storage size of the Audit database.
- **DBAuditName** – specify the name of internal PostgreSQL RDS database. It will be used to establish a connection between your DataSunrise and the Audit Storage.
- **MultiAZAudit** – choose true in case it is required audit db to be available from multiple Availability Zones.
- **DBUser** – specify the name of the user that will be used to access the Dictionary and Audit databases.
- **DBPassword** – specify the password that will be used along with the DBUser as credentials for the Dictionary and Audit databases.

  Note: remember this login and password or store it in some safe place.

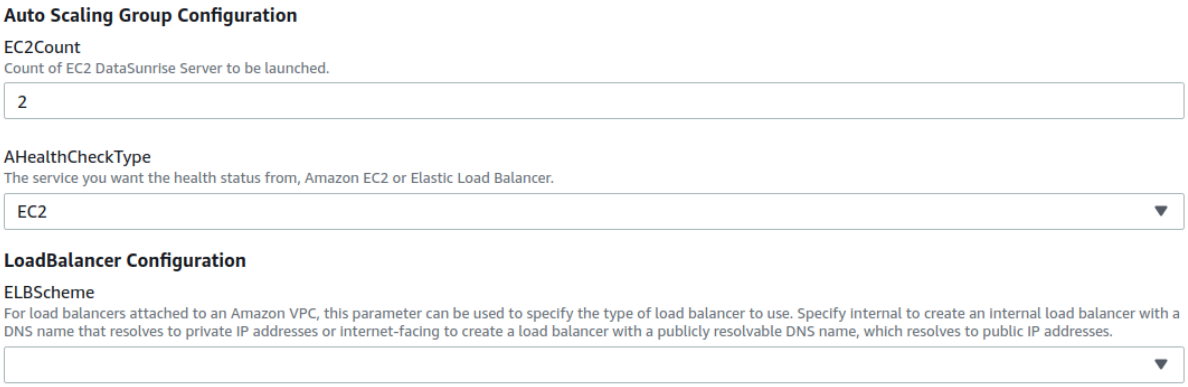
5. Target Database Configuration Section



This section contains 6 fields:

- **TDBType** – specify the type of the database that you are going to protect using DataSunrise (Target database).
- **TDBHost** – specify the hostname or IP address of the Target database.
- **TDBPort** – specify the port number of the Target database.
- **TDBName** – specify the name of any *internal* database of the Target database. This name will be used to establish a connection with the database.
- **TDBLogin** – specify Target database user name that will be used to establish a connection between DataSunrise and the database.
- **TDBPassword** – specify password that will be used along with the TDBLogin.
- **TDBSecuityGroup** – choose security group that is used by protected database from the list.
  Note: in case Target Database is located in different VPC, create a dummy security group and choose it from list.

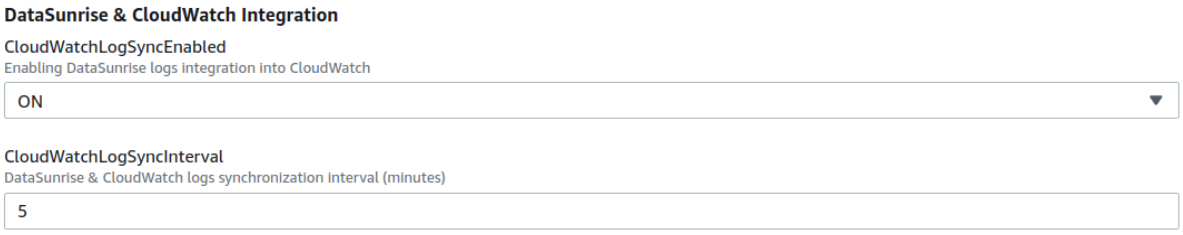
6. Auto Scaling Group and Load Balancer Configuration Sections



These sections contain 3 fields:

- **EC2Count** – count of DataSunrise servers you require to be launched.
- **AHealthCheckType** – type of the healthcheck to be used.
- **ELBScheme** – scheme of the load balancer that will be created. In case you need Load Balancer to be available from public networks choose internet-facing scheme, in case it will be used only by the internal VMs – choose internal.

7. DataSunrise and CloudWatch Integration Section

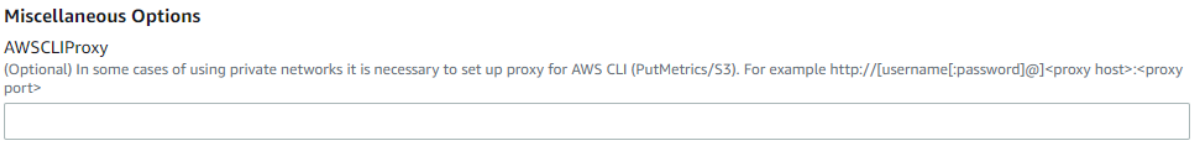


- **CloudWatchLogSyncEnabled** – enables the possibility to upload DataSunrise logs into Cloud Watch for issue troubleshooting.
- **CloudWatchLogSyncInterval** – sets the interval that will be used for sync.

8. AWS DirectoryService Integration (Optional):

- **DirectoryServiceId** – the AWS Active Directory ID that you want DataSunrise EC2 machines to join.
- **ADLogin** – login of the AWS Active Directory Service that will be used to log in.
- **ADPassword** – password that corresponds to login of the AWS Active Directory Service that will be used to log in.
- **ADCIDR** – Active Directory IP range that will be used to access AD ports.

9. Miscellaneous Options Section



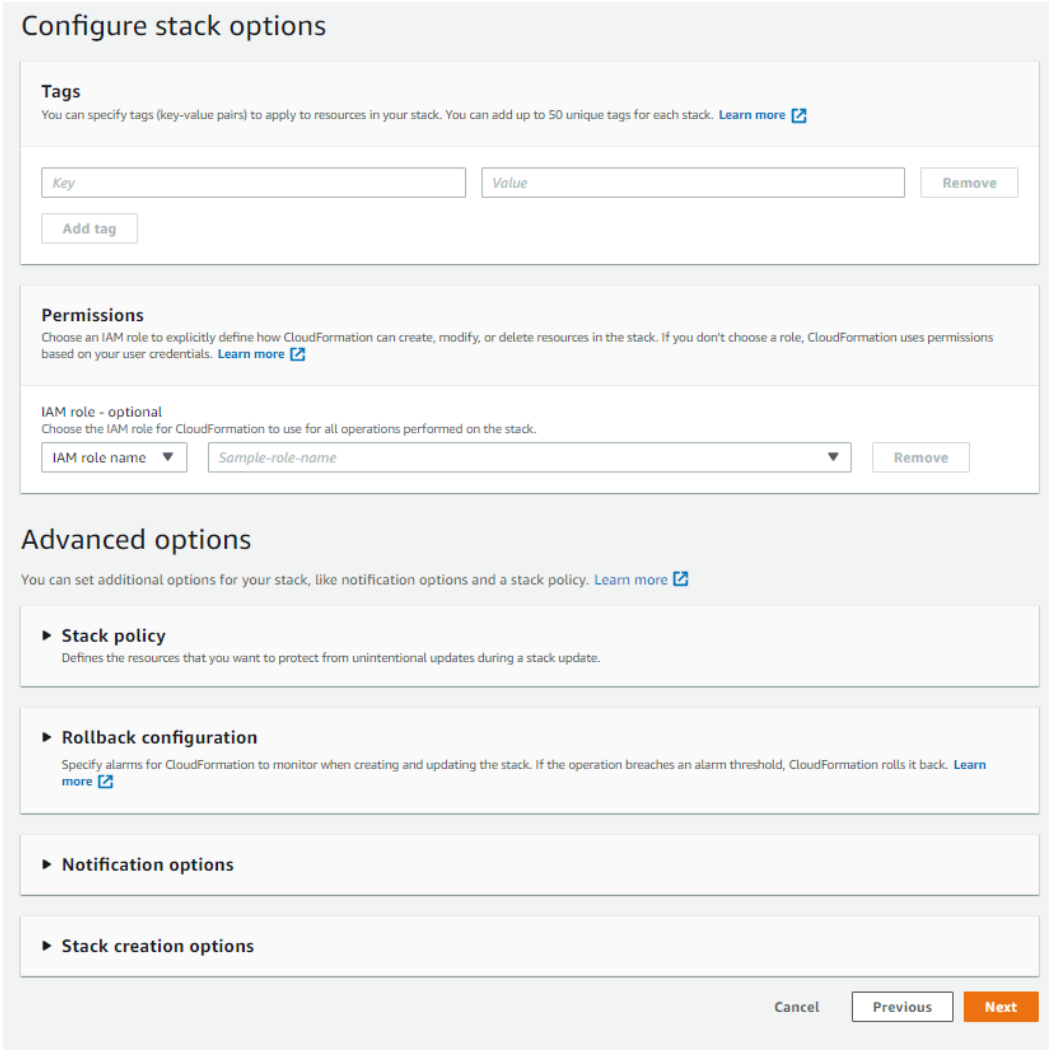
In some cases of using private networks it is necessary to set up a proxy for AWS CLI (PutMetrics/S3).

The template for specifying a proxy is:

http://[username[:password]@]<proxy host>:<proxy port>

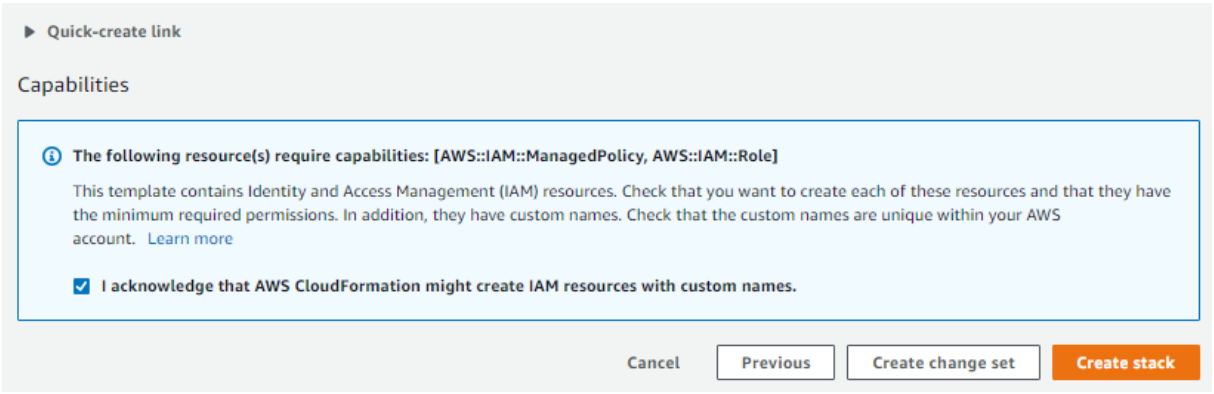
3.3 Finishing the Deployment

Once all the sections are filled out, click **Next**. The following page will appear:

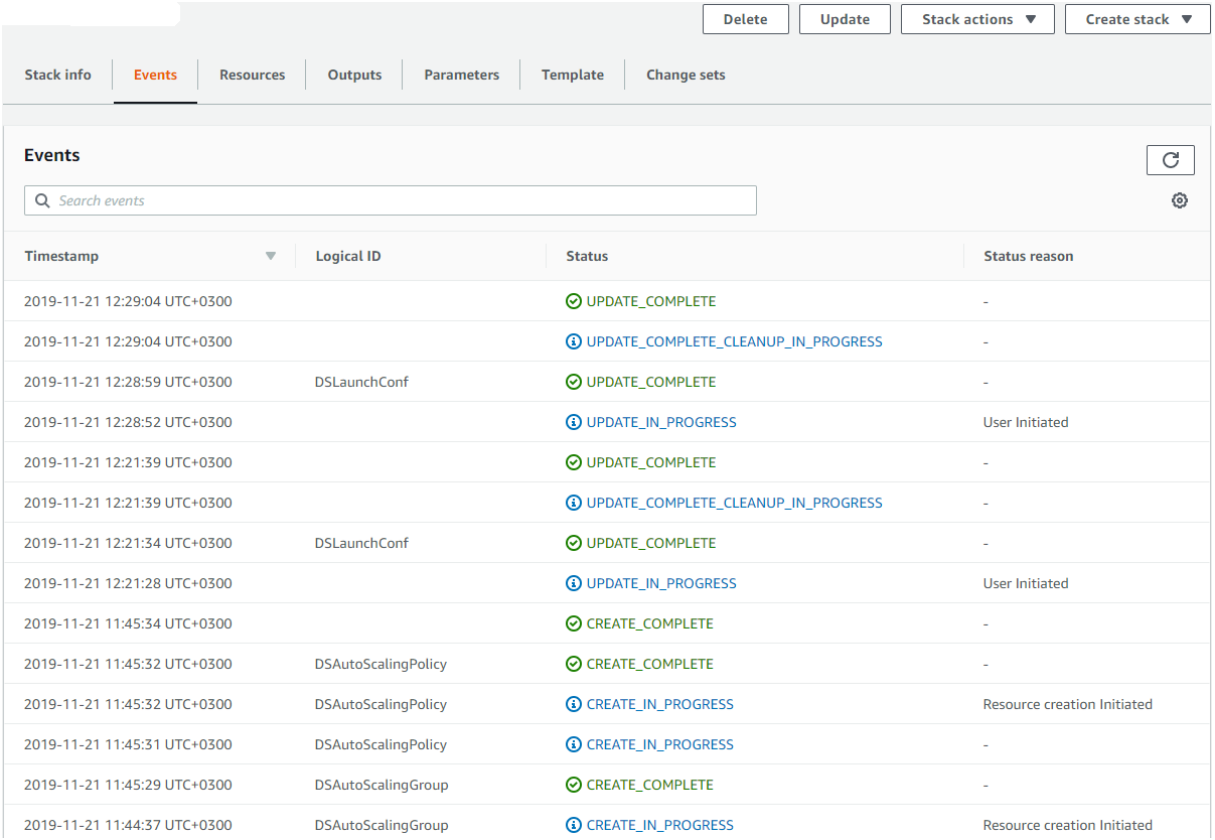


Here you will need to specify a Role that will be used for the Cloud Formation deployment. Choose the role from the drop-down menu in the IAM Role field you created in advance using Appendix IAM Policy Listing.

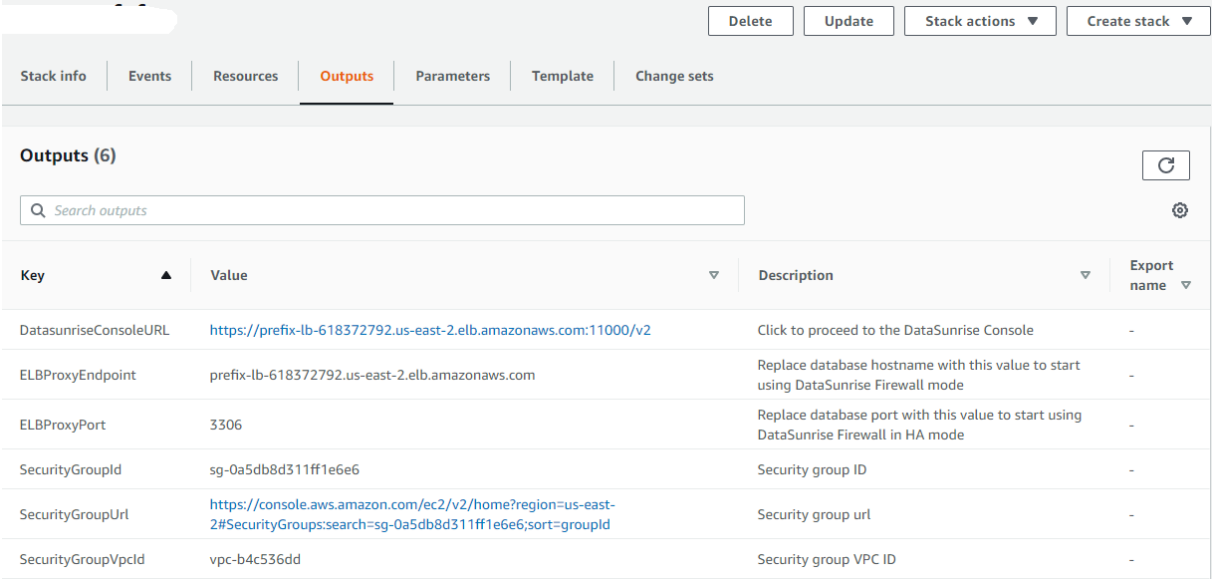
You can check once again the parameters you have specified in the stack's settings. If everything is ok, just check the check box below and click Create Stack:



This is it. If everything is done according to this guide, the stack creation will start and you will be redirected to the Events Page:



Here you can observe resource creation processes. As soon as the Cloud Formation stack receives the “CREATE_COMPLETE” status, you can proceed to the Outputs tab and see the information that can be used to connect to the DataSunrise Web Console or Target database endpoint:



4. Fault-tolerance and Recovery

Detecting an application fault

Application fault detection is one of the top-priority aspects of the infrastructure. The

CloudFormation script has a built-in mechanism for detecting application faults. To activate this mechanism, you must specify their email address in the *AlarmEmail field (optional)* of the CloudFormation script's settings and accept the AWS SNS newsletter as soon as the topic message is received during the deployment of the stack. AWS SNS topic will be triggered if DataSunrise service crashes or any downtime is detected.

In order to improve the fault detection efficiency, we recommend you to configure and use DataSunrise's healthcheck periodic task. This task checks a connection between DataSunrise and the target database, proxies and load balancer. To get notifications, you will need to configure an SMTP server and add a subscriber that will receive notifications about any faults detected. For detailed information, please refer to the DataSunrise User Guide sections "Health Check" and "Subscriber Settings".

Conducting the recovery testing

The CloudFormation template is dedicated to deploying a fault-tolerant and recoverable stack. Fault-tolerance and recovery features are based on AWS Auto Scaling service policies, which contain predefined values for successfully scaling DataSunrise EC2 nodes using the HA configuration.
To test fault tolerance and recovery capabilities of a DataSunrise stack, you can:
1. Simulate CPU usage by running traffic to the proxy server and checking the scaling function.
2. Manually terminate your EC2 instance in order to check if a new EC2 starts back with the DataSunrise service already configured to act as a proxy for a target database.
3. Manually stop the DataSunrise service using the *service datasunrise stop* command to check if the unhealthy EC2 is replaced by a new one with the DataSunrise service already configured to act as a proxy for a target database.

5. IAM Role and Policies

Establishing security settings properly is an obligatory requirement for any software and cloud applications are not exception. Setting up policies and security groups of the cloud infrastructure components can be a challenging and time-consuming task.

DataSunrise CloudFormation script includes all the settings needed for deployment of a production-ready DataSunrise Security Suite in no time and without any manual actions.

The security policies created by the CloudFormation script are based on the least privilege principle and do not provide excessive access to the resources of the environment where DataSunrise is deployed.

In order to use CloudWatch alarms and store backups and logs of DataSunrise, the policies are attached to the **DSVMRole** IAM Role, which is attached to the EC2 instances of

DataSunrise. The creation of the role is also automated via the CloudFormation script and does not require any manual actions to be performed.

The following security policies will be created as an output of CloudFormation script for DataSunrise deployment:

5.1 S3AccessPolicy

This policy is required for using S3 buckets for backup and installation of DataSunrise using its own installer if it was specified:

```
"S3AccessPolicy": {
      "Type": "AWS::IAM::ManagedPolicy",
            "Properties": {
                "PolicyDocument": {
                    "Version": "2012-10-17",
                    "Statement": [
                        {
                            "Effect": "Allow",
                            "Action": [
                                "s3:*"
                            ],
                            "Resource": [
                                { "Fn::Join" : [ "", [ "arn:aws:s3:::", { "Ref" :
"BackupS3BucketName" }, "/*" ] ] },
                                { "Fn::Join" : [ "", [ "arn:aws:s3:::", { "Ref" :
"BackupS3BucketName" } ] ] } ]}]}}}
```

5.2 CWAccessPolicy

This policy allows you to enable and receive CloudWatch notifications about DataSunrise's status:

```
"CWAccessPolicy": {
            "Type": "AWS::IAM::ManagedPolicy",
            "Properties": {
                "PolicyDocument": {
                    "Version": "2012-10-17",
                    "Statement": [
                        {
                            "Effect": "Allow",
                            "Action": [
                                "cloudwatch:DeleteAlarms"
                            ],
                            "Resource": [
                                "*"
                            ] }]}} }
```

5.3 KMSAcessPolicy

This policy allows you to use the specified KMS Key in case the distribution bucket is encrypted:

```
        "KMSAccessPolicy": {
            "Type": "AWS::IAM::ManagedPolicy",
            "Condition" : "HasBucketKeyARN",
            "Properties": {
                "PolicyDocument": {
                    "Version": "2012-10-17",
                    "Statement": [
                        {
                            "Effect": "Allow",
                            "Action": [
                                "kms:Decrypt",
                                "kms:Encrypt",
                                "kms:DescribeKey"
                            ],
                            "Resource": [
                                { "Fn::Join" : [ "", [ { "Ref" :
"BucketKeyARN" } ] ] }
                            ]
                        }
                    ]
                }
            }
        },
```

5.4 DataSunrise VM Policies

Prefixed inline policy – this policy is included in the definition of DSVMRole resource. It provides a limited access to the list of AWS Services:

1. Secrets Manager – storing and handling sensitive DataSunrise instance information (such as admin web console credentials);

2. CloudWatch – define and utilize events on the status of DataSunrise service, put log files into the CloudWatch Log Group;

3. Security Token Service — decode additional information about the authorization status of a request from an encoded message returned in response to an AWS request;

4. EC2 - receive the status of the specified instances or all of your DataSunrise instances;

5. EC2 AutoScaling – manage AutoScaling for DataSunrise EC2 instances;

6. Marketplace Metering – used for Hourly Billing type of licensing of DataSunrise stack.

```
"Policies": [ {
    "PolicyName": { "Fn::Join" : [ "", [ { "Ref" : "DeploymentName" }, "-
DataSunrise-VM-Policy" ] ] },
    "PolicyDocument": {
```

```
        "Version": "2012-10-17",
        "Statement": [
            {
                "Effect" : "Allow",
                "Action" : [
                    "secretsmanager:DescribeSecret",
                    "secretsmanager:GetSecretValue",
                    "secretsmanager:PutSecretValue",
                    "secretsmanager:UpdateSecretVersionStage"
                ],
                "Resource" : [
                    { "Ref" : "dsSecretAdminPassword" },
                    { "Ref" : "dsSecretTargetDBPassword" },
                    { "Ref" : "dsSecretConfigDBPassword" },
                    { "Ref" : "dsSecretLicenseKey" }
                ]
            },
            {
                "Effect" : "Allow",
                "Action" : [
                    "secretsmanager:GetSecretValue"
                ],
                "Resource" : [
                    { "Fn::Join" : [ "", [ "arn:aws:secretsmanager:", { "Ref" :
"AWS::Region" } ,":", { "Ref" : "AWS::AccountId" }, ":secret:db-password-*" ] ] }
                ]
            },
            {
                "Effect": "Allow",
                "Action": [
                    "cloudwatch:PutMetric*",
                    "events:PutEvents",
                    "events:PutRule",
                    "sts:DecodeAuthorizationMessage",
                    "ec2:DescribeInstanceStatus",
                    "ec2:DescribeInstances",
                    "logs:CreateLogGroup",
                    "logs:CreateLogStream",
                    "logs:Put*",
                    "logs:DescribeLogStreams",
                    "ssm:UpdateInstanceInformation"
                ],
                "Resource": [
                    "*"
                ]
            },
            {
                "Effect": "Allow",
                "Action": [
                    "ec2:*SecurityGroup*"
                ],
                "Resource": [
                    { "Fn::Join" : [ "", [ "arn:aws:ec2:", { "Ref" : "AWS::Region"
} ,":", { "Ref" : "AWS::AccountId" }, ":security-group/", { "Fn::GetAtt" :
[ "DSSG", "GroupId" ] } ] ] }
                ]
            },
            {
                "Effect": "Allow",
                "Action": [
                    "autoscaling:SetDesiredCapacity",
                    "autoscaling:UpdateAutoScalingGroup"
```

```
                ],
                "Resource": [
                    { "Fn::Join" : [ "", [ "arn:aws:autoscaling:", { "Ref" :
"AWS::Region" } ,":", { "Ref" : "AWS::AccountId" },
":autoScalingGroup:*:autoScalingGroupName/", { "Fn::Join" : [ "", [ { "Ref" :
"DeploymentName" }, "-asg" ] ] } ] ] }
                ]
            },
            {
                "Effect": "Allow",
                "Action": [
                    "aws-marketplace:MeterUsage"
                ],
                "Resource": "*"
            }]}} ]
```

6. Customer Support

In case you are unable to solve the problem on your own, please send email to support_center@datasunrise.com. In your email please provide the following information:

- CloudFormation Template used
- The issue observed (cannot connect to DataSunrise WEB-UI, proxy, instances constantly terminating etc.)
- The version of DataSunrise used in the deployment
- If possible, provide a copy of /var/log/cloud-init-output.log file which contains the outputs of the User Data-loaded installation script

Alternatively, sign up at support.datasunrise.com and raise a ticket via the Support Task Tracker portal. The customer support queries are processed by the Support Team based on the Severity and the context of a query and timely handled. The Support Team of DataSunrise spans multiple time zones to guarantee the best reaction time.

FAQ

Q: I see the “CREATE_COMPLETE” stack status but cannot connect to the Console using the URL from the Outputs tab. Why?
A: Sometimes it requires time for Amazon EC2 machines to become up and ready. Try to wait for 10 minutes maximum.

Q: I waited for 10 minutes and still cannot access the DataSunrise Web Console using Load Balancer’s URL.
A: Make sure that CIDR you input to the script’s settings corresponds to the IP address of the machine you are trying to log in from.

Q: I deployed the CloudFormation script and when I am trying to connect to the Web

Console using the Load Balancer endpoint and port 11000, I am having a loop. What's the problem?

A: The situation you've encountered is connected with the Load Balancer round-robin system and self-signed certificates that are used for DataSunrise Web Console. As soon as you accept both of the Certificates the issue should be solved. Sometimes loops happen and it's normal behavior.

However, we suggest using Safari in case you're running a macOS, or Google Chrome browser in case you are running Linux or Windows OS.

This issue is also solvable by signing certificates at your local CA or by replacing certificates on each machine with your own certificate that is already trusted and signed by your CA. The certificates should be placed into the appfirewall.pem file by replacing existing content with your trusted certificate.

Q: How to perform Key-Rotation to keep my connection secure?
A: DataSunrise's Web Console features a section to keep your private keys rotated and up-to-date. In order to rotate your keys, simply change them in the WebUI -> Configuration -> SSL Key Groups. For more information please refer to the DataSunrise User Guide section "SSL Key Groups".

Q: I got the following errors in System Events: "DS_31001E: AWS Metering Service failure!" I am also unable to create and use any Rule and get the following error: "New rules are no longer covered by license". These two errors are displayed simultaneously.
A: When using Hourly Billing license type, it is necessary to have public access your AWS Marketplace Metering endpoint. This can be done by attaching Internet Gateway, NAT Gateway to your private subnet route table or using HTTP Proxy with Internet Acess. In case none of these options are permitted, please use the BYOL License type. To get a BYOL license, please contact DataSunrise support team.

Q: When I try to update an AWS CloudFormation stack, I get an error message similar to the following: "CloudFormation cannot update a stack when a custom-named resource requires replacing. Rename 'MYResourceXXX' and update the stack again." How can I resolve this error?
A: In previous Cloud Formation template Audit and Dictionary database publicity was set by default and was not specified. We have added the next parameter to the Dictionary and Audit resources: *PubliclyAccessible: false*
In order to resolve the problem with the AWS Cloud Formation Stack Update simply find and delete this line from the template wherever it's specified and perform the Update procedure again.

Q: How can I manage my DataSunrise license used in the deployment?
A: To update the license type or the license key (BYOL case), use the two-step procedure: first, update the license key in the DataSunrise WEB-UI (System Settings – About – License Manager – Add License – Copy the updated key in the opened modal page) and persist changes. Secondly, update the DataSunrise CloudFormation Stack DSLicenseKey and

DSLicenseType (if you switch from HB to BYOL). This would change the license key stored in the Secrets Manager secret, which is used by DataSunrise installation script. If a key is updated only in the License Manager menu of DataSunrise WEB-UI, then during EC2 instance recreation (due to failed health checks or planned upgrade) the old key would be installed to the DataSunrise config. Although adding a key in the WEB-UI of DataSunrise is enough and both old and new keys can exist in configuration database without any disruption, this might create a confusion.

Q: How do I safely perform the update of sensitive data stored in Secrets Manager secrets?

A: To efficiently update the secrets, update the corresponding DataSunrise CloudFormation Stack Input Parameter. Once the Stack is updated, terminate the existing instances and let the new ones to be provisioned. The fresh instances will be using the updated Stack Inputs.

Appendix

MinimalAccountPolicy.json

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:CreateBucket",
                "s3:DeleteBucket"
            ],
            "Resource": "arn:aws:s3:::cf-templates*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject"
            ],
            "Resource": "arn:aws:s3:::cf-templates*/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudformation:*",
                "iam:PassRole",
                "iam:ListRoles",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:DescribeKeyPairs",
                "sns:ListTopics"
            ],
            "Resource": "*"
        }
    ]
}
```