AWS Trusted Advisor Explorer

Implementation Guide

AWS Trusted Advisor Explorer: Implementation Guide

Table of Contents

Aggregate cost optimization recommendations and actively track cost optimization health across your organization with AWS Trusted Advisor Explorer

AWS Implementation Guide

AWS Solutions Builder Team

May 2020

This implementation guide discusses architectural considerations and configuration steps for deploying AWS Trusted Advisor Explorer in the Amazon Web Services (AWS) Cloud. It includes links to an AWS CloudFormation template that launches and configures the AWS services required to deploy this solution using AWS best practices for security and availability.

This solution is intended for customers who have multiple accounts in an organization and want to aggregate cost recommendations across their accounts.

The guide is intended for IT infrastructure architects, administrators, and DevOps professionals who have practical experience architecting in the AWS Cloud.

Overview

The AWS Trusted Advisor Explorer solution automatically provisions the infrastructure necessary to help customers actively track their cost recommendations across their organization over time. The solution provides an effective way to drive cost optimization through specific resource tags. The solution also helps deliver cost key performance indicators (KPIs) for your organization.

The solution leverages AWS Trusted Advisor cost optimization recommendations and AWS Resource Groups tag editor data to build a data lake that can be queried using Amazon Athena and visualized using Amazon QuickSight or any other visualization platform.

Accounts must have a Business or Enterprise level AWS support plan to gain access to AWS Trusted Advisor's cost optimization checks.

Cost

You are responsible for the cost of the AWS services used while running the solution. As of the date of publication, the cost for running this solution with default settings in the US East (N. Virginia) Region with 200 accounts is approximately **$5.00 per month**—approximately 2.5 cents per account. For full details, see the pricing information for each AWS service used in the solution.

Architecture Overview

Deploying this solution builds the following environment in the AWS Cloud.

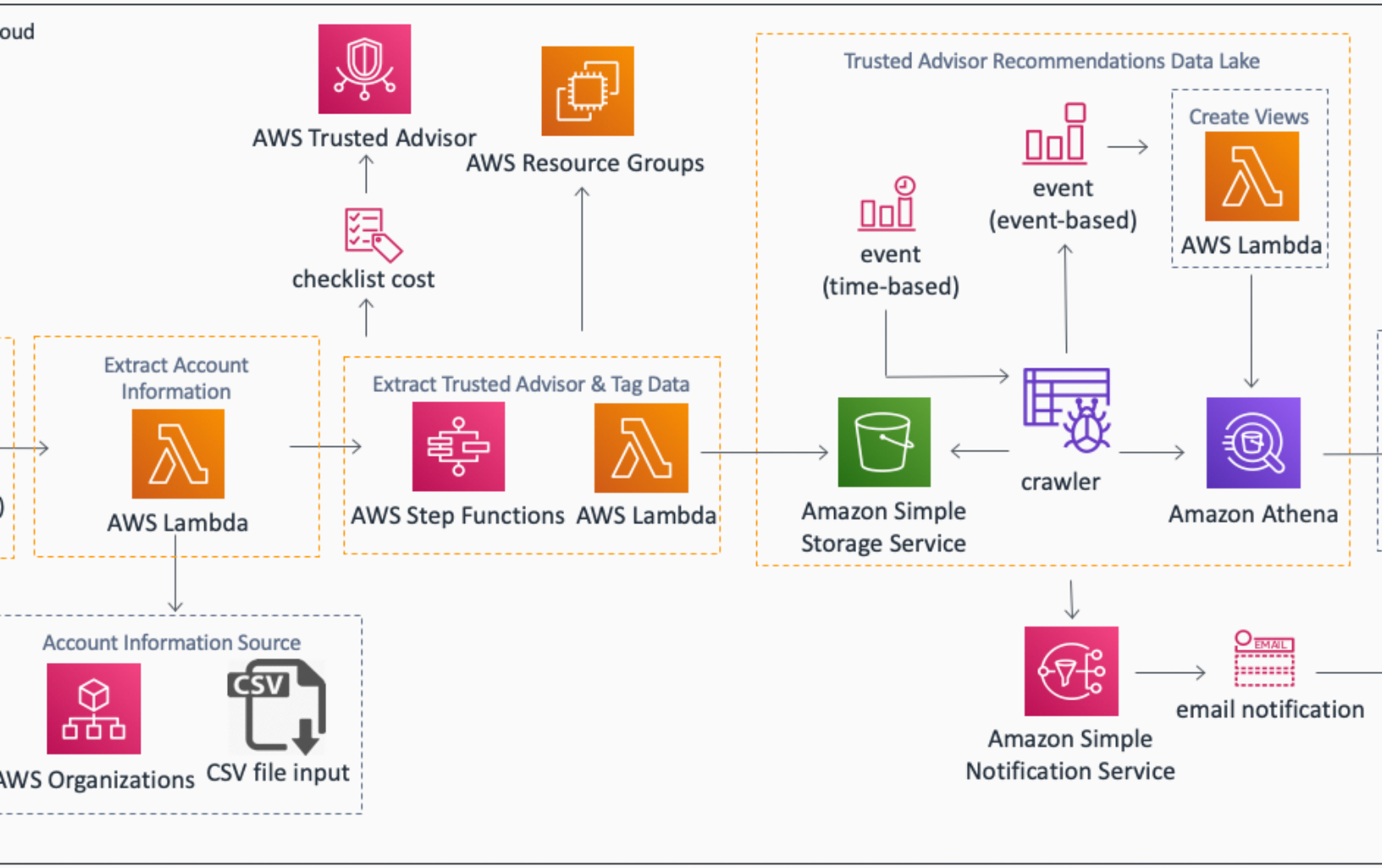

Figure 1: AWS Trusted Advisor Explorer overview

The AWS CloudFormation template must be deployed in your AWS Organization's Primary (Master) account. The Primary account is the AWS account you use to create your organization. For more information, see the AWS Organizations terminology and concepts in the *AWS Organizations User Guide*.

The template creates four essential building blocks for this solution:

- The scheduler block
- The extract account information block
- The extract Trusted Advisor & tag data block
- The Trusted Advisor Recommendations data lake block

The scheduler block is an Amazon CloudWatch Events rule that triggers the solution based on a schedule defined by user.

The extract account information block contains an AWS Lambda function that extracts the list of accounts from the existing organization in the account or from a CSV file input.

The extract Trusted Advisor & tag data block contains four AWS Step Functions. These four Step Functions are composed of five AWS Lambda functions that work in parallel to extract AWS Trusted

Advisor cost recommendations and tag data from all of the member accounts and store them in an Amazon Simple Storage Service (Amazon S3) bucket.

The Trusted Advisor Recommendations data lake block contains Amazon S3, AWS Glue crawlers, Amazon Athena, AWS Lambda, and CloudWatch Events rules. The workflow is triggered by a time-based CloudWatch Events rule on a schedule defined by the user.

The template deploys two Amazon S3 buckets, one for storing the raw Trusted Advisor cost recommendations and tag data, and the other for access logging. It also deploys two Glue crawlers that crawl the raw data from the Amazon S3 bucket to create tables in an Amazon Athena database. When the AWS Glue crawler finishes, another event-based CloudWatch Events rule triggers which invokes an AWS Lambda function to create the required Amazon Athena views.

The solution leverages AWS Trusted Advisor cost optimization recommendations and AWS Resource Groups Tag Editor data to build a data lake that can be queried using Amazon Athena and visualized using Amazon QuickSight or any other visualization platform.

For more information and a detailed solution workflow, see Appendix B (p. 13).

AWS CloudFormation Template

The solution uses AWS CloudFormation to automate the deployment of AWS Trusted Advisor Explorer in the AWS Cloud. It includes the following AWS CloudFormation template, which you can download before deployment:

View Template **aws-trusted-advisor-explorer.template:** Use this template to launch AWS Trusted Advisor Explorer and all associated components. The default configuration deploys AWS Lambda functions, CloudWatch Events rules, AWS Step Functions, Amazon Simple Storage Service (Amazon S3) buckets, an Amazon Athena database, and AWS Glue crawlers. You can customize the template based on your specific needs.

Automated Deployment

Before you launch the automated deployment, review the architecture, configuration, network security, and other considerations discussed in this guide. Follow the step-by-step instructions in this section to configure and deploy the AWS Trusted Advisor Explorer solution into your account.

Time to deploy: Approximately 5 minutes

Prerequisites

Each member account must have a Business or Enterprise level AWS Support plan in order to gain access to the AWS Trusted Advisor cost optimization checks.

Each member account must have a cross-account role that trusts the Primary account. The name of this cross-account role must be identical (case sensitive) in all the member accounts.

> **Note**
> When you create a member account in your organization, AWS Organizations automatically creates an AWS Identity and Access Management (IAM) role in the member account that enables IAM users in the Primary account to exercise full administrative control over the member account. This role is subject to any service control policies (SCPs) that apply to the member account. If you don't specify a name, AWS Organizations gives the role a default name: `OrganizationAccountAccessRole`.

See Appendix E (p. 23) for more information about creating the cross-account member role.

Launch the Stack

The automated AWS CloudFormation template deploys AWS Trusted Advisor Explorer in the AWS Cloud. Ensure that your member accounts have a Business or Enterprise level AWS Support plan, and that you have already deployed the cross-account role into the member accounts.

> **Note**
> You are responsible for the cost of the AWS services used while running this solution. See the Cost (p. 2) section for more details. For full details, see the pricing webpage for each AWS service you will be using in this solution.

1. Sign in to the AWS Management Console and click the button below to launch the `aws-trusted-advisor-explorer` AWS CloudFormation template.

   

   You can also download the template as a starting point for your own implementation.
2. The template launches in the US East (N. Virginia) Region by default. To launch this solution in a different AWS Region, use the Region selector in the console navigation bar.
3. On the **Create stack** page, verify that the correct template URL shows in the **Amazon S3 URL** text box and choose **Next**.

4. On the **Specify stack details** page, assign a name to your solution stack.
5. Under **Parameters**, review the parameters for the template and modify them as necessary. This solution uses the following default values.

Parameter	Default	Description
Cross Account Role Name	*<Requires input>*	Specify the cross-account role name that exists in all of the member accounts
Language	`en`	English is the only supported language.
Report Schedule	`cron(0 9 1 * ? *)`	Enter the frequency at which you would like to trigger the data collection and aggregation. For more information, see Cron Expressions in the *Amazon CloudWatch Events User Guide*.
Interested Tag Keys	`optional input`	Enter the resource tags you would like to extract from the member accounts. For example: `env`, `costcenter`, `asset_id`, etc.
Glue Crawler Schedule	`cron(0 11 1 * ? *)`	Enter the frequency for triggering the AWS Glue crawler to update the data lake. For more information, see Cron Expressions in the *Amazon CloudWatch Events User Guide*. **Note** Set this value for two hours past the report scheduler's cron.
Log Level	`ERROR`	Choose the log level for the Lambda functions. Enter either `ERROR` or `INFO`.
Mask Account Information	`TRUE`	This value ensures that the Account ID, Account Name, and Account Email information is masked in the logs.
SNS Email	*<Requires input>*	Enter an email address to receive a notification every time the solution successfully runs.

6. Choose **Next**.
7. On the **Configure stack options** page, choose **Next**.
8. On the **Review** page, review and confirm the settings. Be sure to check the box acknowledging that the template will create AWS Identity and Access Management (IAM) resources.

9. Choose **Create stack** to deploy the stack.

   You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should see a status of **CREATE_COMPLETE** in approximately five minutes.

CloudFormation Output

This solution created the following resources.

Resource	Description
`AthenaDatabase`	The name of the Athena database.
`RawTADataBucketName`	The name of the bucket in which the raw Trusted Advisor check data & tag information will be stored.
`SNSTopic`	The name of the SNS topic that will be notified after every data refresh.
`UUID`	Random, universally unique identifier for the deployment used for operational metrics.

Security

When you build systems on AWS infrastructure, security responsibilities are shared between you and AWS. This shared model reduces your operational burden as AWS operates, manages, and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the services operate. For more information about security on AWS, visit the AWS Security Center.

Amazon S3

The Amazon Simple Storage Service (Amazon S3) buckets created in the solution is private and has server-side encryption enabled. We recommend that you review the Amazon S3 buckets and further restrict access as needed after the deployment is up and running.

Lambda Logs

By default, Account ID, Account Name, and Account Email are masked while storing AWS Lambda logs in CloudWatch. To unmask this information, set the `MaskAccountInformation` template parameter to `FALSE`.

IAM Roles

AWS Identity and Access Management (IAM) roles enable customers to assign granular access policies and permissions to services and users on the AWS Cloud. The solution creates IAM roles and sets permissions in the respective accounts to allow the solution to assume a defined role in the member account and extract data when necessary.

Additional Security Enhancements (Optional)

Glue Catalog

You can encrypt the metadata stored in the Glue Data Catalog using keys that you manage with AWS Key Management Service (AWS KMS). For more information, see Encrypting Your Data Catalog in the *AWS Glue Developer Guide*.

SNS

You can enable server-side encryption (SSE) for the topic created by the solution to protect its data. To learn more, refer to the Enabling server-side encryption (SSE) for an Amazon SNS topic with an encrypted Amazon SQS queue subscribed topic in the *Amazon Simple Notification Service Developer Guide*.

Amazon SNS uses AWS Key Management Service (AWS KMS) to provide encryption at rest. Messages published to the Amazon SNS encrypted topic must have access permissions to execute the AWS KMS operation GenerateDataKey and Decrypt. For more information, see Encrypting Messages Published to Amazon SNS with AWS KMS blog post.

CloudWatch Logs Logs

You can enable encryption for the log groups created when running the solution. To learn more, refer to the CloudWatch Logs documentation.

Additional Resources

- AWS CloudFormation
- AWS Lambda
- Amazon Simple Notification Service
- Amazon Simple Storage Service
- Amazon QuickSight
- AWS Trusted Advisor
- CloudWatch
- Amazon Step Functions
- Amazon Athena
- AWS Glue

Appendix A: Create a Custom Account List

If you want to run this solution for a limited number of accounts, and if you do not want the solution to obtain a list of active accounts from your AWS Organizations, you can provide a csv input file.

The following is an example CSV file.

```
 AccountId,AccountName,AccountEmail
123456789101,aws-sample-account-dev,sampleemail1@yourdomain.com
123456789102,aws-sample-account-prod,sampleemail2@yourdomain.com
123456789103,aws-sample-account-beta,sampleemail3@yourdomain.com
...
```

Use the following procedure to upload the CSV file to the Amazon S3 bucket created by the solution:

1. Create a folder called `ManualInput` in the solution's Amazon Simple Storage Service (Amazon S3) bucket.
2. Upload the csv file to the `ManualInput` folder.
3. Navigate to the AWS Lambda console and choose the `GetAccountslambda` Lambda function.
4. Update the following environment variables of the `GetAccountsLambda` Lambda function:
   - FILE_OVERRIDE: **`TRUE`**
   - OBJECT_NAME : **`ManualInput/`***<filename>*.csv

Appendix B: Detailed Solution Workflow

The following section describes the workflow for extracting Trusted Advisor cost recommendations and resource tags.

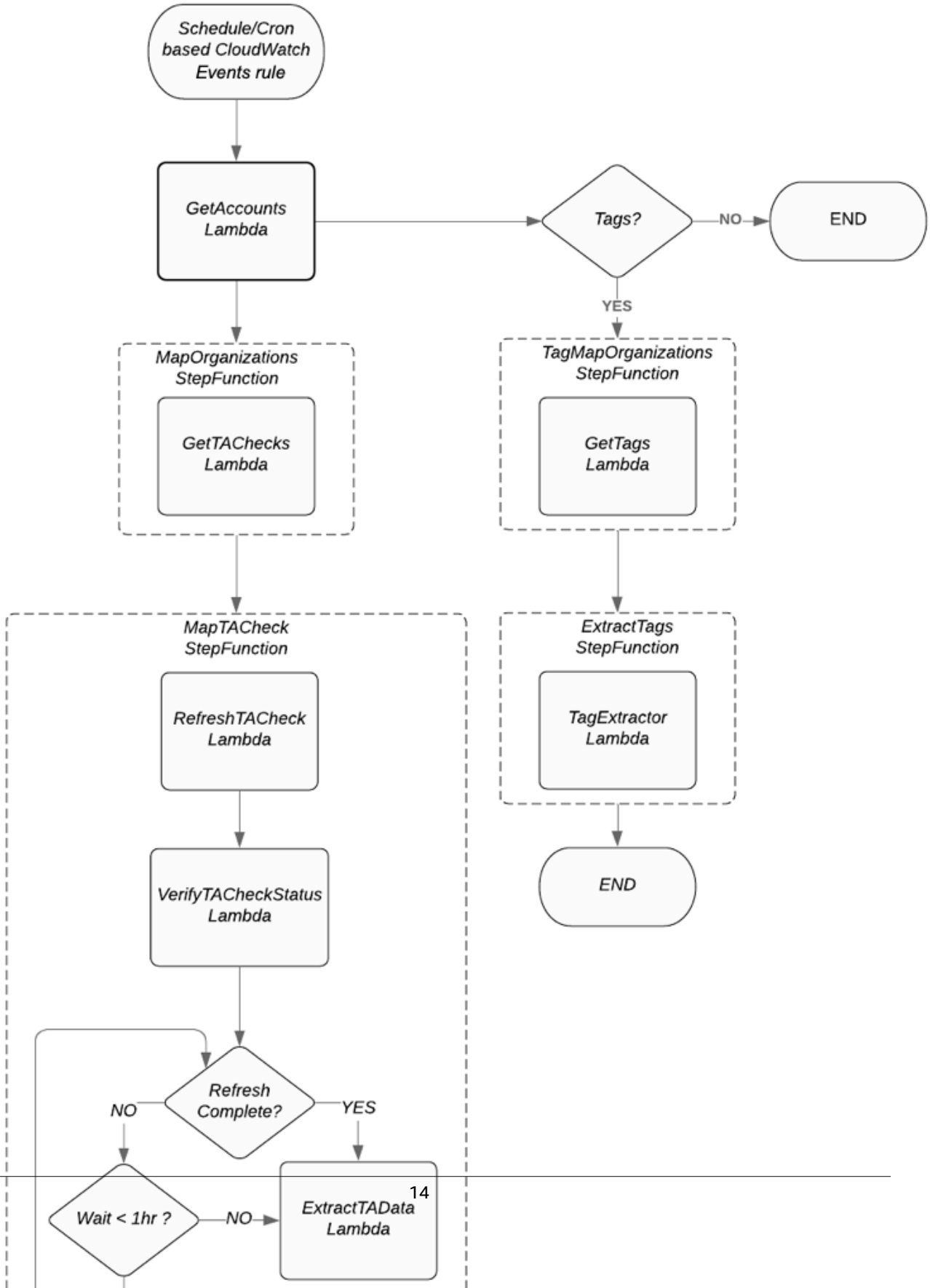
Schedule/Cron based CloudWatch Events rule
GetAccounts Lambda
Tags?
NO
END
YES
MapOrganizations StepFunction
GetTAChecks Lambda
TagMapOrganizations StepFunction
GetTags Lambda
MapTACheck StepFunction
RefreshTACheck Lambda
ExtractTags StepFunction
TagExtractor Lambda
VerifyTACheckStatus Lambda
END
Refresh Complete?
NO
YES
Wait < 1hr ?
NO
ExtractTAData Lambda

Figure 2: Extract Trusted Advisor recommendations and resource tag data workflow

The `GetAccountsLambda` AWS Lambda function is triggered on a schedule using an CloudWatch Events rule. The cron schedule is defined by the user while deploying the solution. This Lambda function retrieves all active accounts from AWS Organizations, batches them into groups of fifty, and then executes the `MapOrganizations` step function once per batch.

If resource tags are specified by the user as part of input parameters of the AWS CloudFormation template, the `GetAccountsLambda` Lambda function also executes the `TagMapOrganizations` Step Functions once per batch of 50 accounts.

> **Note**
> You can provide a file input instead of using AWS Organizations. For more information, see Appendix A (p. 12).

The `MapOrganizations` and `TagMapOrganizations` Step Functions are executed once per batch of 50 accounts. Each account entry in the batch contains of the following parameters:

- Account ID
- Account Name
- Account Email
- Date (for example, 12-01-2019)
- DateTime (for example, 2019-12-01 09:00:13)

MapOrganizations Step Functions

The `MapOrganizations` Step Functions are composed of the `GetTAChecks` Lambda function that runs the `DescribeTrustedAdvisorChecks` API and extracts the Cost Optimization Check Ids, Check Names, and Categories text fields. It appends these extracted fields to each of the entries in the input batch and then invokes the `MapTACheck` Step Functions once per Account.

The following is an example of the input batch passed on to the `MapTACheck` Step Functions:

- Account ID
- Account Name
- Account Email
- Date (for example, 12-01-2019)
- DateTime (for example, 2019-12-01 09:00:13)
- CheckID (for example, Qch7DwouX1)
- CheckName (for example, Low Utilization Amazon EC2 Instances)
- Category (for example, cost_optimizing)
- Language (for example, en)

TagMapOrganizations Step Functions

The `TagMapOrganizations` step function is composed of one Lambda function, (`GetTAChecks`) that runs the `DescribeRegions` API and extracts all the AWS Regions. It appends the regions and resource types to the batch and invokes the `ExtractTags` Step Functions.

The following is an example of the input batch passed on to `ExtractTags` Step Functions:

- Account ID

- Account Name
- Account Email
- Date (for example, 12-01-2019)
- DateTime (for example, 2019-12-01 09:00:13)
- ResourceType (for example, Amazon RDS:db)
- Region (for example, eu-north-1)

MapTACheck Step Functions

The `MapTACheck` Step Functions contains three AWS Lambda functions: `RefreshTACheck`, `VerifyTACheckStatus`, and `ExtractTAData`. The Step Functions starts off with first running `RefreshTACheck` AWS Lambda function that runs a `RefreshTrustedAdvisorCheck` API call to refresh the Trusted Advisor checks in all of the member accounts.

The `VerifyTACheckStatus` AWS Lambda function runs the `DescribeTrustedAdvisorCheckRefreshStatuses` API call and determines the wait duration for the check refresh to complete.

> **Note**
> The Step Functions only waits for 3600 secs. If the check takes more than 3600 seconds to refresh, the solution ignores the wait time and proceeds to extracting the recommendations data.

The `ExtractTAData` AWS Lambda function runs the `DescribeTrustedAdvisorCheckResult` API call for extracting the Trusted Advisor check data, write it into a csv file and pushes the csv file to an Amazon S3 bucket.

ExtractTags Step Function

The `ExtractTags` Step Functions contains one Lambda function. The `TagExtractor` Lambda function runs the `ResourceGroupsTaggingAPI`'s `GetResources` API and is responsible for extracting the associated resource tags for the given resource type in the input batch. The output is stored in a CSV file and is pushed to an Amazon S3 bucket.

The following section describes the workflow for creating the Trusted Advisor recommendations data lake.

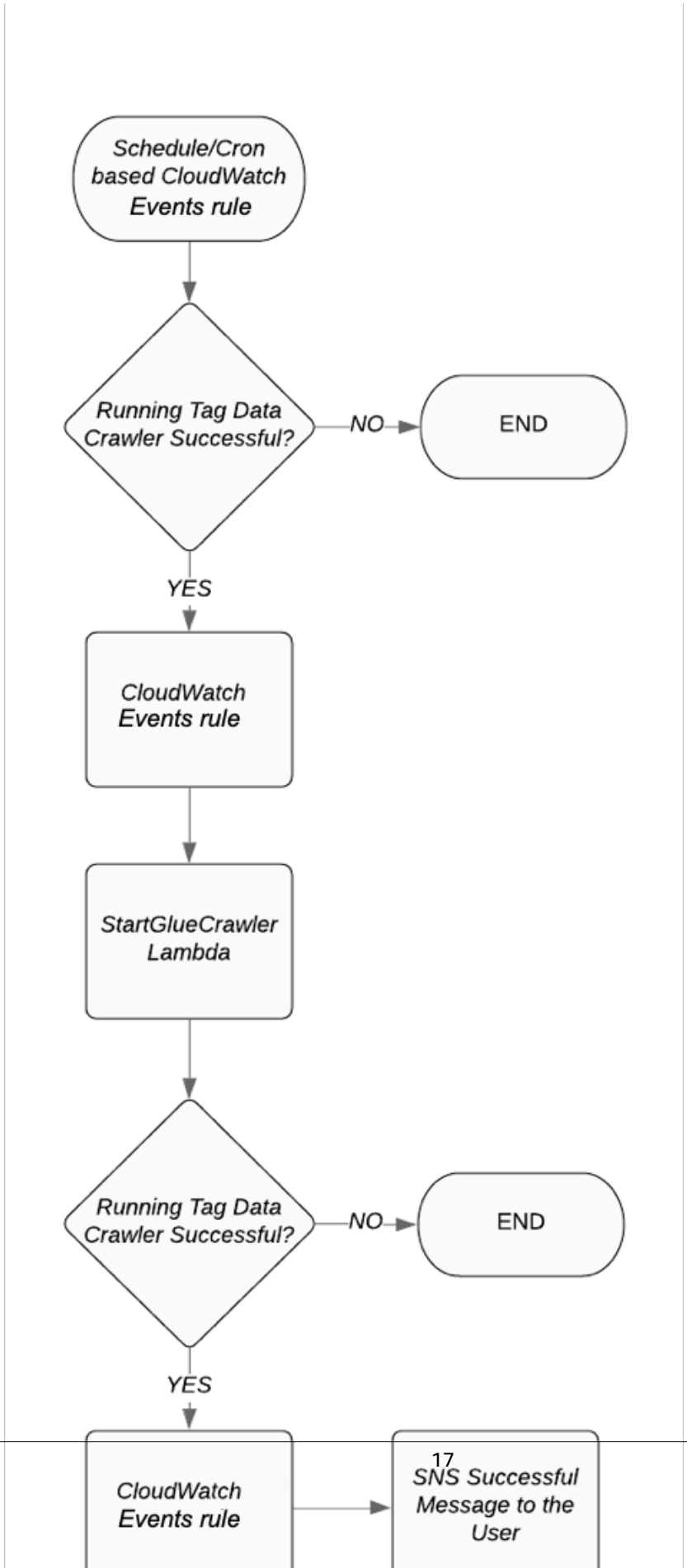
Schedule/Cron based CloudWatch Events rule
Running Tag Data Crawler Successful?
NO
END
YES
CloudWatch Events rule
StartGlueCrawler Lambda
Running Tag Data Crawler Successful?
NO
END
YES
CloudWatch Events rule
SNS Successful Message to the User

Figure 3: Create Trusted Advisor recommendations data lake workflow

Note
The create data lake workflow must be triggered at least two hours after the extract Trusted Advisor recommendations and resource tag data workflow is triggered.

`AWSTrustedAdvisorExplorer_Tags_Crawler` is triggered on a schedule based on the cron defined by the user at the time of deploying the AWS CloudFormationtemplate. This crawler populates the AWS Glue Data Catalog with the Resource Tag table.

An event based CloudWatch Events rule triggers as result of successful completion of `AWSTrustedAdvisorExplorer_Tags_Crawler`. This CloudWatch Event rule invokes the `StartGlueCrawlerLambda` Lambda function, which triggers the `AWSTrustedAdvisorExplorer_Crawler` crawler.

`AWSTrustedAdvisorExplorer_Crawler` populates the AWS Glue Data Catalog with Trusted Advisor check data tables. Another CloudWatch Events rule triggers after `AWSTrustedAdvisorExplorer_Crawler` finishes. This CloudWatch Event rule invokes the `CreateAthenaViewLambda` Lambda function, which creates the required Athena views and posts an Amazon Simple Notification Service (SNS) notification to the `AWSTrustedAdvisorExplorer-DataRefresh` topic.

The user can now access the Athena console and run queries against the populated data. The user can also import the views into Amazon QuickSightto build Amazon QuickSight dashboards for visualization.

Appendix C: Enhance Solution Performance

You can enhance performance by raising the Lambda concurrent executions limit in the account where the solution is deployed. You can request a limit increase in the AWS Management Console.

Appendix D: Visualize Data in Amazon QuickSight

Configure Amazon QuickSight

Use this procedure to visualize the data this solution collects.

Before you begin, your account must be registered for Amazon QuickSight. For more information, refer to Setting Up Amazon QuickSight.

1. Navigate to the Amazon QuickSight console.
2. Choose your username on the top right of the console, then select **Manage QuickSight**.
3. Choose **Security & Permissions**.
4. Under QuickSight access to AWS services, choose **Add or Remove**.
5. Select **Amazon S3**. If this option is already selected, uncheck and recheck the option.
6. Select the specific Amazon Simple Storage Service (Amazon S3) solution bucket listed in the output section of the AWS CloudFormation deployment from the list of Amazon S3 buckets. Select **Write permission for Athena Workgroup**.
7. Choose **Finish**.
8. Choose **Update**.

Create a Data Source and Import the First Data Set

Use the Creating a Data Set Using Amazon Athena Data topic in the *Amazon QuickSight Developer Guide* to create a new data source and import the first data set into QuickSight.

For Step 9 in the *Creating a Data Set Using Amazon Athena* documentation, choose the `aws_trusted_advisor_explorer_db` database.

For Step 10 in the *Creating a Data Set Using Amazon Athena* documentation, select any table ending with `_view` (for example, `amazonrdsidledbinstances_view`).

Creating a Data Set Using an Existing Amazon Athena Data Source

Use the Creating a Data Set Using an Existing Amazon Athena Data Source topic in the *Amazon QuickSight Developer Guide* to create the remaining Athena views as new Data Sets in Quicksight.

For Step 5 in the *Creating a Data Set Using an Existing Amazon Athena Data Source* documentation, select tables ending with `_view` (for example, `amazonrdsidledbinstances_view`) and repeat the process until you create a data set for all the Athena views tables.

Adding All the Data Sets to an Analysis

After you create the data sets for the Athena views tables, you must add them to an analysis.

Use the Add or Edit a Data Set topic in the *Amazon QuickSight Developer Guide*to add a data set to an existing analysis.

You can delete unwanted analysis that were created as part of the data set creation.

You can create a dashboard that visualizes the AWS Trusted Advisor Explorer data. For more information, see Working with Data in Amazon QuickSight and Working with Analyses in the *Amazon QuickSight User Guide*.

Columns Appended & Modified to aid Amazon QuickSight Visualization

In the `summary_view` table, the following additional columns have been added/modified:

- **date_time**: Use this column in Amazon QuickSight for any visuals requiring date-time measure. This column is adjusted to reflect as a Date datatype field in QuickSight.
- **optimizationPercent**: The Optimization percentage per checkID; This field can be used to determine how optimized the specific check is for an account. This can also be used to understand the cost savings opportunity percentage by account.

```
Formula: (1-(resourcesflagged/resourcesprocessed)) *100
```

- **trueoptimizationPercent**: This field has the same function as **optimizationPercent** but it excludes resources marked as ignored and suppressed.

```
Formula:(1-(resourcesflagged-(resourcesignored+resourcessuppressed)/resourcesprocessed))
 *100
```

In the `amazonrdsidledbinstances_view`, `idleloadbalancers_view` tables, the following additional columns have been added/modified:

- **date_time**: Use this column in QuickSight for any visuals requiring date-time measure. This column is adjusted to reflect as a Date datatype field in Quicksight.
- **estimated_monthly_savings**: Use this column in QuickSight for any visuals requiring monthly savings measure. This column is adjusted to reflect as a Decimal datatype field in QuickSight.

In the `unassociatedelasticipaddresses_view`, `route53latencyresourcerecordsets_view` tables, the following additional column has been added/modified:

- **date_time**: Use this column in QuickSight for any visuals requiring date-time measure. This column is adjusted to reflect as a Date datatype field in Quicksight.

In the `ec2reservedinstanceleaseexpiration_view` table, the following additional columns have been added/modified:

- **date_time**: Use this column in QuickSight for any visuals requiring date-time measure. This column is adjusted to reflect as a Date datatype field in QuickSight.
- **current_monthly_cost**: Use this column in QuickSight for any visuals requiring current monthly costs measure. This column is adjusted to reflect as a Decimal datatype field in QuickSight.
- **estimated_monthly_savings**: Use this column in QuickSight for any visuals requiring monthly savings measure. This column is adjusted to reflect as a Decimal datatype field in QuickSight.

- **expiration_date**: The expiration date field from the original data is modified to reflect as a Date datatype field in QuickSight.

In the `ec2reservedinstancesoptimization_view` table, the following additional columns have been added/modified:

- **date_time**: Use this column in QuickSight for any visuals requiring date-time measure. This column is adjusted to reflect as a Date datatype field in QuickSight.
- **estimated_savings_with_recommendation_monthly, upfront_cost_of_ris, estimated_cost_of_ris_monthly, estimated_on-demand_cost_post_recommended_ri_purchase_monthly**: These fields from the original data are modified to reflect as a Decimal datatype field in QuickSight.

In the `lowutilizationamazonec2instances_view` table, the following additional columns have been added/modified:

- **date_time**: Use this column in QuickSight for any visuals requiring date-time measure. This column is adjusted to reflect as a Date datatype field in QuickSight.
- **estimated_monthly_savings**: Use this column in QuickSight for any visuals requiring monthly savings measure. This column is adjusted to reflect as a Decimal datatype field in QuickSight.
- **average_cpu_utilization_14_days, average_network_i/o_utilization_14 days**: These fields from the original data are modified to reflect as a Decimal datatype field in QuickSight.

In the `underutilizedamazonebsvolumes_view` table, the following additional columns have been added/modified:

- **date_time**: Use this column in QuickSight for any visuals requiring date-time measure. This column is adjusted to reflect as a Date datatype field in QuickSight.
- **Monthly_Storage_Cost**: This field from the original data is modified to reflect as a Decimal datatype field in QuickSight.

Refreshing a Data Set on a Schedule

You can configure an automatic refresh for all of your imported data sets. Use the Refreshing a Data Set on a Schedule topic in the *Amazon QuickSight Developer Guide* to set up a refresh schedule for all of your imported data sets.

> **Note**
> Ensure that your QuickSight refresh schedule is aligned with your Glue crawler schedule. The QuickSight data set refresh must run after the Glue crawlers finish running.

Appendix E: Deploy a Cross-Account IAM role in AWS Member Accounts

This appendix is applicable for customer who do not have a cross-account role that trusts the Primary (Master) account in all of the member accounts. In such cases, deploy the following template in all member accounts.

Use this template to launch the cross-account role. The default configuration deploys an AWS Identity and Access Management (IAM) role that trusts the Primary account. A Primary account is the AWS account you use to create your organization and is the account in which the `aws-trusted-advisor-explorer` solution stack will be deployed. You can also customize the template based on your specific needs.

> **Note**
> If you have an IAM role in your member account that trusts the payer account, you can reuse that role. You may need to adjust permissions associated to that role to include AWS managed `AWSSupportAccess` and `ResourceGroupsandTagEditorReadOnlyAccess` permissions policies.

1. Sign in to the AWS Management Console and launch the `cross-account-member-role` AWS CloudFormation template.

   You can also download the template as a starting point for your own implementation.
2. The template launches in the US East (N. Virginia) Region by default.
3. On the **Create stack** page, verify that the correct template URL shows in the **Amazon S3 URL** text box and choose **Next**.
4. On the **Specify stack details** page, assign a name to your solution stack.
5. Under **Parameters**, review the parameters for the template and modify them as necessary. This solution uses the following default values.

Parameter	Default	Description
MasterAccountNumber	*<Requires input>*	12-digit account id of the Primary (Master) account where you will deploy the solution.
CrossAccountRoleName	*<Requires input>*	Name of the IAM member Role. This name must be consistent across all the member accounts.

6. Choose **Next**.
7. On the **Configure stack options** page, choose **Next**.
8. On the **Review** page, review and confirm the settings. Be sure to check the box acknowledging that the template will create AWS Identity and Access Management (IAM) resources.
9. Choose **Create stack** to deploy the stack.

   You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should see a status of **CREATE_COMPLETE** in approximately two minutes.

Appendix F: Additional Logs

Assume Role Failure Logs

Assume Role failures encountered in the member accounts are logged into the solution's `logs` folder in the Amazon S3 bucket. These logs help you identify which member accounts have a missing or misconfigured cross-account role.

Athena Logs and Outputs

Athena logs and outputs will be stored into the solution's Amazon S3 bucket in folder called `AthenaOutputs`.

Appendix G: Collection of Operational Metrics

This solution includes an option to send anonymous operational metrics to AWS. We use this data to better understand how customers use this solution and related services and products. When enabled, the following information is collected and sent to AWS each time the `get-accounts-info-lambda` Lambda function runs:

- **Solution ID:** The AWS solution identifier
- **Version** Solution version
- **Unique ID (UUID):** Randomly generated, unique identifier for each AWS Trusted Advisor Explorer deployment
- **Timestamp:** Data-collection timestamp
- **Region:** Region in which the solution is deployed
- **SolutionRunTime:** Trusted Advisor Data-collection timestamp

AWS will own the data gathered via this survey. Data collection will be subject to the AWS Privacy Policy. To opt out of this feature, complete the following task.

Modify the AWS CloudFormation template mapping section as follows:

```
"Send" : {
"AnonymousUsage" : { "Data" : "yes" }
},
```

to

```
"Send" : {
"AnonymousUsage" : { "Data" : "no" }
},
```

Source Code

You can visit our GitHub repository to download the templates and scripts for this solution, and to share your customizations with others.

Document Revisions

Date	Change
May 2020	Initial release

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.