
Intern Project – Service Limits Check

Teng Liu

Aug 10, 2017

Intern Project – Service Limits Check

Build a feature that shows the potential impact on a customer's service limits based on a proposed stack creation or update. This feature should provide a warning to customers without necessarily invalidating their change. This feature should ideally support services supported for service limits check by Trusted Advisor. The approach should be to build in support for displaying limit information about one type of resource; from there we can build upon that infrastructure to support others.

This could be presented as part of change sets.

Description:

Design and build a Service Limits Check feature which shows the potential impact on a proposed stack creation or update.

Milestones:

1. Design for presenting information about a service limit violation
2. Implementation of service limit checks for one resource
3. Automated testing of service limit checks
4. Implementation of further resource types for service limits

5. Automated testing of service limit checks for all supported resources

Nomenclature

Limit:

An AWS-imposed maximum usage for a certain resource type in AWS. Limits are generally either account-wide or per-region. They have AWS global default values, but can be increased by AWS Support.

Usage:

“Usage” refers to the current customer’s usage of a specific resource that has a limit. Usage values/amounts (such as the number of VPCs) are represented by instances of the Usage class.

Threshold:

The point at which AWS Trusted Advisor will consider the current usage for a limit to be the key issue here. AWS Trusted Advisor will set the usage $\geq 80\%$ of the limit “Warning” message.

Trusted Advisor:

This feature supports retrieving the current service limits and current usage via the AWS Trusted Advisor, for limits which Trusted Advisor tracks (currently a subset of what AWS service knows). The results of this

check may not be available via the API for all accounts, since from the December 2016, the Trusted Advisor documentation illustrates that while this check is available for all accounts, API access is only available to accounts with Business- or Enterprise-level support plans. If the customer does not have Trusted Advisor access, the API call will result in a “SubscriptionRequiredException”

The following table shows the limits that Trusted Advisor checks.

The Project Motivation

The customers find that stack creation/updates fail at the final stage of AWS CloudFormation due to the service limit exhaustion. Once the AWS CloudFormation proceeds to this level, lots of time has been wasted which makes the whole process not effective.

Current state of art

1. AWS CloudFormation does not compute the projected usage and limit check for the current status of resources

Possible solution

Customers should be able to get the query of the current usage for the desired resources from the created AWS CloudFormation stacks with the already used resources apart from AWS CloudFormation and also be able

to receive the upcoming warning from the anticipated “dangerous” action while creating new stacks or updating the stacks (based on AWS Trusted Advisor). The feature should be able to support to display the information about one type of resource; from there it can be built upon that infrastructure to support others.

One more notion is that this feature now will only support the subset of the AWS CloudFormation Resources union AWS Trusted Advisor Limit Check Resources as indicated above.

Technical Design

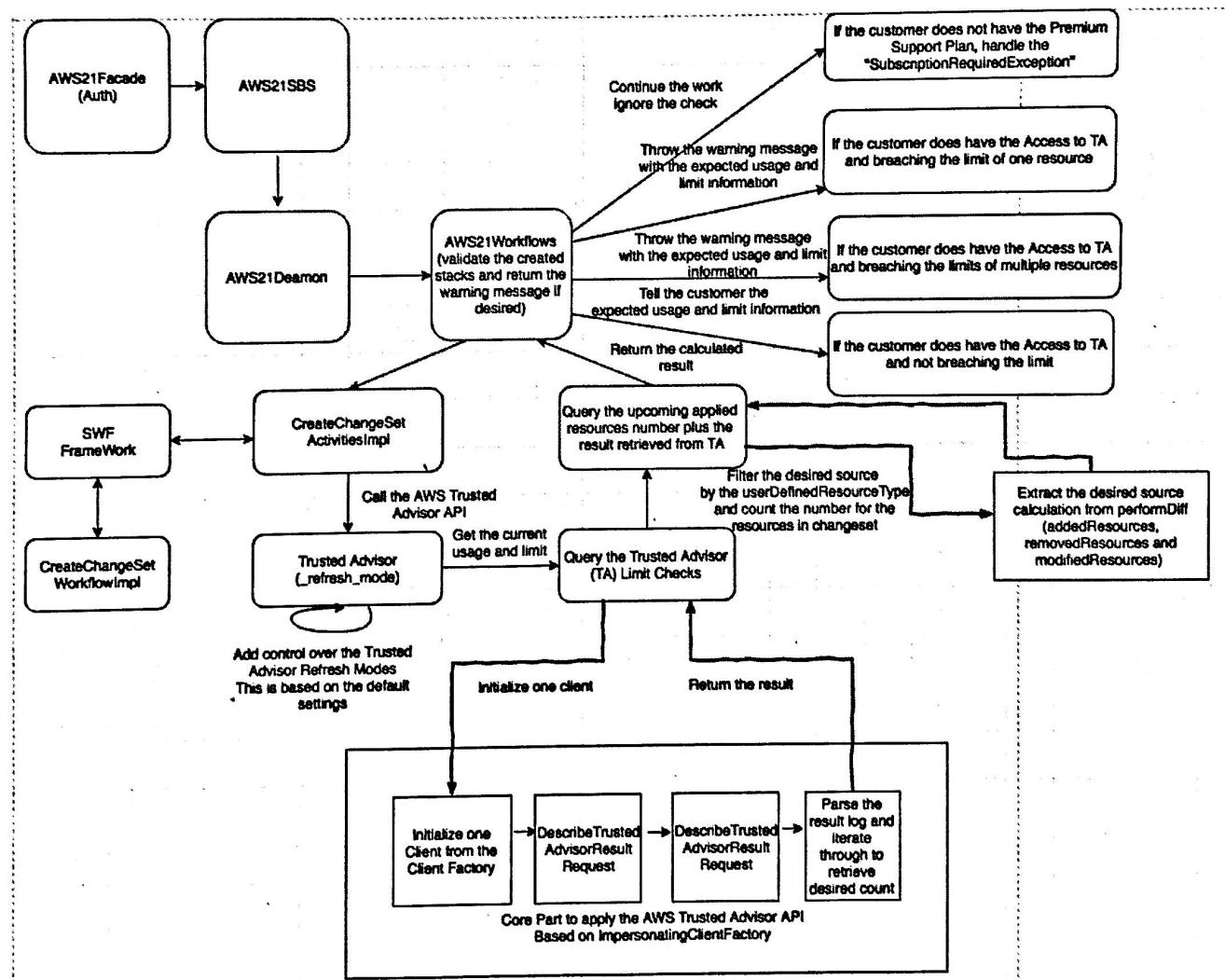
High-Level Design

The core of technical design is to use AWS Trusted Advisor model and merge it into the current AWS CloudFormation workflows structure. By using the segments of AWS Trusted Advisor API (AWS support) to integrate into AWS CloudFormation’s backend calls and setup, the customer will be able to view the expected usage of the desired resources and the anticipated warning message according to the query from AWS Trusted Advisor.

The basic design is to utilize the AWS Trusted Advisor to synchronously detect the service limits based on one resource usage snapshot.

Implementation Details

1. Fully rely on AWS Trusted Advisor, the new feature is developed upon the CreateChangesetActivity in AWS21Workflows the logic flow chart is as described followed:



Technical logic flow and details of AWS Trusted Advisor:

When the program proceeds to the CreateChangeSet, one instance (client) with the AWS Trusted Advisor activity is initialized. The service limit check will also support programmatically refreshing the Trusted Advisor (TA) Check, in order to get updated limit and usage values. The refresh logic operates in the following modes:

Ta_Refresh_Period Integer: This operation will make the check being refreshed with the limit and current usage information polled if its current usage and limit information is at least Integer seconds or minutes old.

Ta_Refresh_Trigger: This operation will be refreshed and the program will continue on immediately, without waiting for the refresh to complete; this will certainly result in still check results in the current run.

Additionally, this feature should support one TA_Refresh_Timeout model. If this has been set as one integer, the refresh will time out after the number of seconds. If a timeout occurs, a message will be logged as error level, but this will not affect the program continuing running. This mode is for the account with huge resources utilization.

Important notion: For customers with the Premium edition, the Trusted Advisor data is automatically refresh weekly.

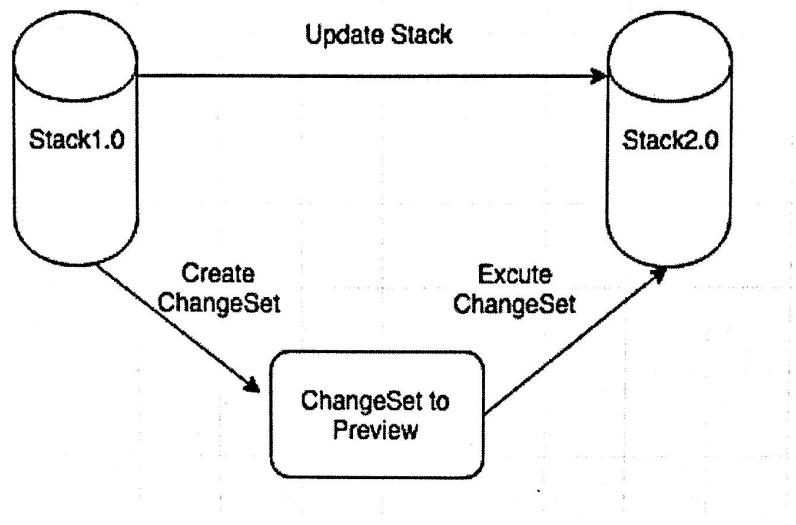
For now, Ta_refresh_period will determine whether the Trusted Advisor Data will be refreshed based on the time when the last TA snapshot was taken; this will retrieve the limits for all “flaggedResources” items in the Service Limits Trusted Advisor check result for the current AWS account. It will then determine which mode to refresh the status, to get the latest updated usage and limit information.

The technical implementation details for impersonating one client in the CreateChangeSetWorkflow is to add desired activities in CreateChangeSet package with the detailed implementation information, also the workers of the whole workflow will be activating the function to get the query and return the usage with limit information from the trusted advisor, once the information queried from the Trusted Advisor is done, it will start to make the calculation for the upcoming applied resource. All the workflow logic in the workers will be implemented under asynchronous way.

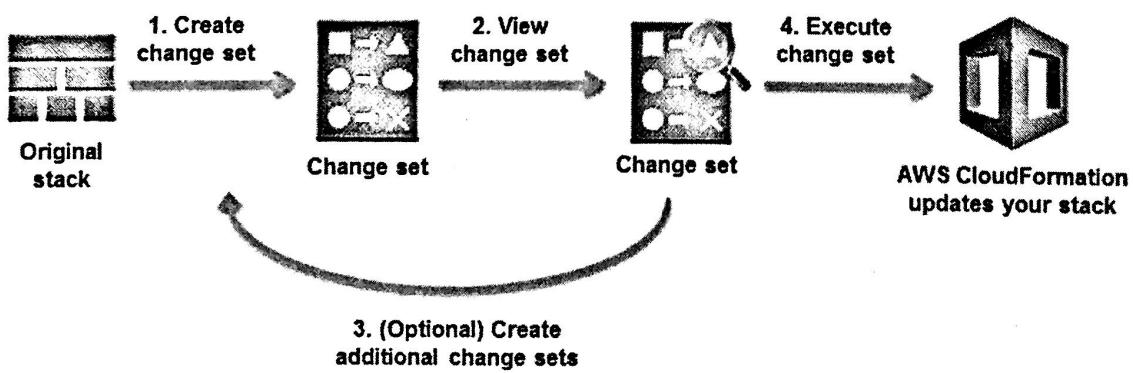
Technical logic flow of the query for upcoming applied resources

The core part should be tracking the dynamic performDiff in the workers to get the accurate upcoming applied desired resources number. The performDiff in the workers determines the logic flow for the createChangeSetActivities. The performDiff will have two parts of the resources, which are sourceResources and destinationResources. Within those resources, three parts of resourcesModification will be obtained, the first one is addedResources, the second one is removedResources and the third one is modifiedResources. For the reason that the TemplateResource from sourceResources and destinationResources includes the type of different resources, the target resources will be retrieved from these three parts of the resourcesModification with the digesting the TemplateResources to get the accurate upcoming applied resources number.

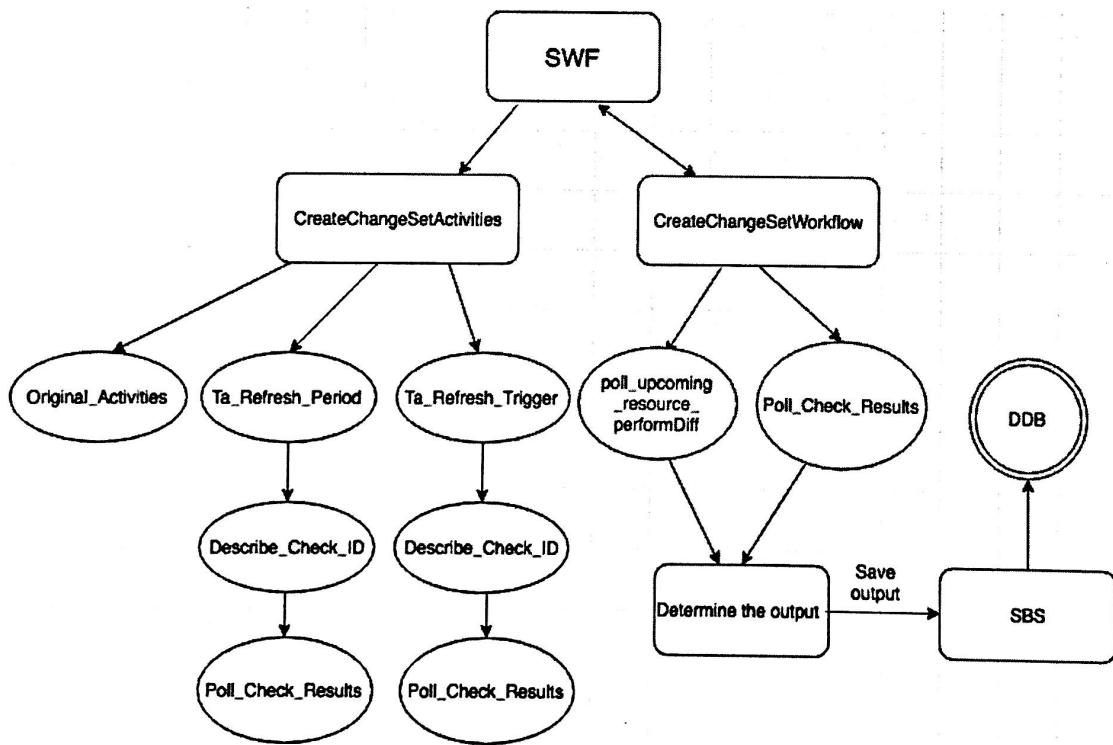
The reason for why CreateChangeSet in AWS21Workflow is one proper place to be equipped with the AWS Trusted Advisor Service to check the limit described as followed:



ChangeSet is one more natural place to be equipped with the TrustedAdvisor and limit check feature, since it will not duplicate any work rendering a more effective way to handle the check. Another reason is that from the CreateChangeSet work flow, it will be more natural and easier to retrieve the upcoming applied resources. Another reason is that CreateChangeSet is one proper place for catching the delta value for the upcoming applied target resources number.



Data flow for the limit check feature



The service limit feature is based on the AWS flow framework. The implementation of polling the usage and limit information from the

Trusted Advisor will be based on four new activities added into CreateChangeSetActivities (two refresh modes activities, one describes check activity, one poll results activities), the dependency is as described above. Once the CreateChangeSetActivity client finishes the calculation of the number for upcoming resources and polling the results from Trusted Advisor, the two numbers will be added together to determine whether there will be the limit breaching warning.

Technical logic flow for visualization of the output

For the output of usage and limit information with warning message, they will be wrapped into the details of each change and outputted in the AWS console or terminal (with cli input). For different user cases, there will be different output based on different conditions, the output will have several categories as followed:

Case 1: If the user does not have the access towards the Trusted Advisor, there will not be any information showing the limit check and current usage;

```
[  
 {  
   "resourceChange": {  
     "logicalResourceId": "ASASG2N2A7",  
     "action": "Add",  
     "physicalResourceId": null,  
     "resourceType": "AWS::AutoScaling::AutoScalingGroup",  
     "replacement": null,  
     "details": [],  
     "scope": []  
   },  
   "type": "Resource"  
 },  
 {  
   "resourceChange": {  
     "logicalResourceId": "RDSDBCS5C0",  
     "action": "Add",  
     "physicalResourceId": null,  
     "resourceType": "AWS::RDS::DBCluster",  
     "replacement": null,  
     "details": [],  
     "scope": []  
   },  
   "type": "Resource"  
 }]
```

Case 2: If the potential CreateChangeSet activity will breach one resource limit, one message will be created within the CreateChangeSet and DescribeChangeSet to tell the user this is one potential “dangerous action” with the expected usage and limit information;

```
[
  {
    "resourceChange": {
      "logicalResourceId": "ASASG2N2A7",
      "action": "Add",
      "physicalResourceId": null,
      "resourceType": "AWS::AutoScaling::AutoScalingGroup",
      "replacement": null,
      "details": [],
      "scope": []
    },
    "type": "Resource"
  },
  {
    "resourceChange": {
      "logicalResourceId": "RDSDBCS5C0",
      "action": "Add",
      "physicalResourceId": null,
      "resourceType": "AWS::RDS::DBCluster",
      "replacement": null,
      "details": [],
      "scope": []
    },
    "type": "Resource"
  },
  {
    "limitCheck": {
      "resourceType": "AWS::RDS::DBCluster",
      "expectedUsage": 41,
      "limit": 40,
      "warning": "This changeSet will breach the service limit for AWS::RDS::DBCluster",
    },
    "type": "Resource"
  }
]
```

Case 3: If the potential CreateChangeSet activity will breach multiple resources' limits, some messages will be created within the CreateChangeSet and DescribeChangeSet to tell the user this is one potential "dangerous action" with the expected usage and limit information;

```
[
  {
    "resourceChange": {
      "logicalResourceId": "ASASG2N2A7",
      "action": "Add",
      "physicalResourceId": null,
      "resourceType": "AWS::AutoScaling::AutoScalingGroup",
      "replacement": null,
      "details": [],
      "scope": []
    },
    "type": "Resource"
  },
  {
    "resourceChange": {
      "logicalResourceId": "RDSDBCS5C0",
      "action": "Add",
      "physicalResourceId": null,
      "resourceType": "AWS::RDS::DBCluster",
      "replacement": null,
      "details": [],
      "scope": []
    },
    "type": "Resource"
  },
  {
    "limitCheck": {
      "resourceType": "AWS::RDS::DBCluster",
      "expectedUsage": 41,
      "limit": 40,
      "warning": "This changeSet will breach the service limit for AWS::RDS::DBCluster"
    },
    "type": "Resource"
  },
  {
    "limitCheck": {
      "resourceType": "AWS::AutoScaling::AutoScalingGroup",
      "expectedUsage": 22,
      "limit": 20,
      "warning": "This changeSet will breach the service limit for AWS::AutoScaling::AutoScalingGroup"
    },
    "type": "Resource"
  }
]
```

Case 4: If the potential CreateChangeSet activity will not breach any resource limit, the current usage with the limit information which can be retrieved from the trusted advisor will be used as the output.

```
[
  {
    "resourceChange": {
      "logicalResourceId": "ASASG2N2A7",
      "action": "Add",
      "physicalResourceId": null,
      "resourceType": "AWS::AutoScaling::AutoScalingGroup",
      "replacement": null,
      "details": [],
      "scope": []
    },
    "type": "Resource"
  },
  {
    "resourceChange": {
      "logicalResourceId": "RDSDBCS5C0",
      "action": "Add",
      "physicalResourceId": null,
      "resourceType": "AWS::RDS::DBCluster",
      "replacement": null,
      "details": [],
      "scope": []
    },
    "type": "Resource"
  },
  {
    "limitCheck": {
      "resourceType": "AWS::RDS::DBCluster",
      "expectedUsage": 38,
      "limit": 40,
      "warning": null,
    },
    "type": "Resource"
  }
]
```

Pros:

- All the operations for the TA will be performed in the synchronous API fashion. This means the parent workflow (CreateChangeSet, DescribeChangeSet) will be waiting for the Trusted Advisor calls to be finished.
- Customers will have the intuition to determine whether this limit will be beyond their threshold.

Cons:

- Only works for the Business and Enterprise support users.
- This feature only monitors a small subset of the AWS limits

Exception Handling

- When the user does not have the Premium edition, the query for the AWS Trusted Advisor will fail, however, this will not have the impact upon the upcoming customer's action, the internal call will throw the exception “SubscriptionRequiredException” and be logged.

How to handle this exception:

There will be one condition upfront to determine whether the user does have the access towards the AWS Trusted Advisor. If the user does not have, “SubscriptionRequiredException” will be logged without breaking the workflow of the CreateChangeSet; for the user does have the access, it will continue the checking work.

- If the refresh of Trusted Advisor takes long time.

How to handle this exception:

There will be refresh_time_out mode equipped into the activities, once the threshold value is breached, it will poll the Trusted Advisor limit check results immediately to continue the workflow.

- If the query from Trusted Advisor hangs or some internal errors occur while retrieving the usage and limit information.

How to handle this exception:

Approach 1: Continue the work for createChangeSet;

Approach 2: Create the messages for customers with the notification of the current issues for Trusted Advisor.

Create Unit Test and Integration Test

Test Plan:

- User with no access towards the trusted advisor:

The expected outcome: Trusted Advisor Client will not be initialized and the program will continue to run.

- The CreateChangeSet action is going to breach the limit of one resource:

The expected outcome: there will be warning message generated with the limit and expected usage information

- The CreateChangeSet action is going to breach the limits of multiple resources:

The expected outcome: there will be warning messages generated with the limit and expected usage information for all the resources

- The CreateChangeSet action is not going to breach the limit:
The expected outcome: there will be message generated with the limit and expected usage information without warning message

Implementation Details

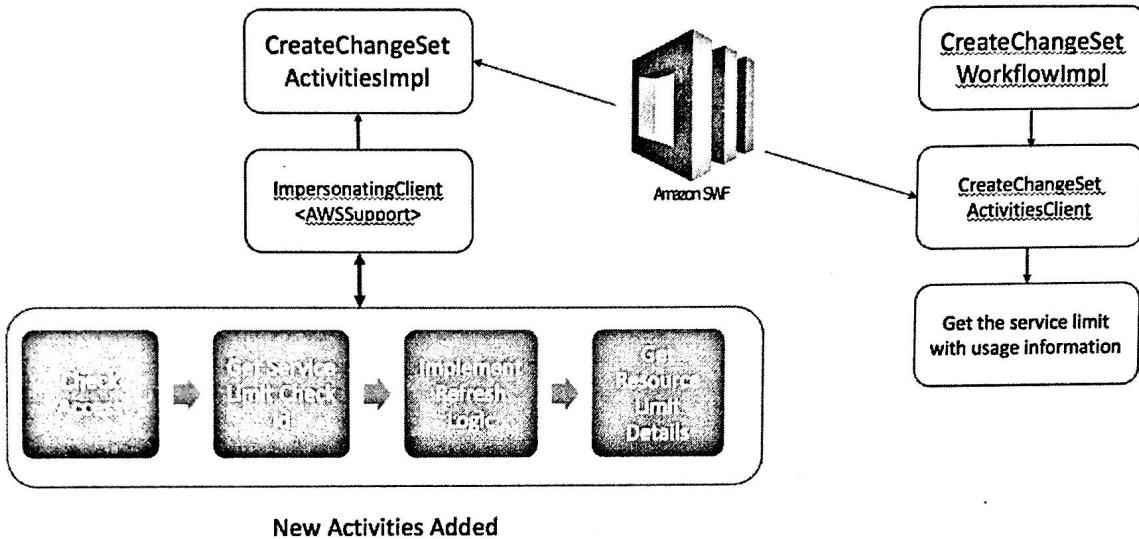
Connecting to Trusted Advisor

This feature supports retrieving the customers' current service limits via the Trusted Advisor "Service Limits" performance check, for limits which Trusted Advisor tracks (currently a subset of what TA knows about intersection with the resources provided by the AWS CloudFormation).

Trusted Advisor information is important to the whole feature, however, as it provides the current service limit values for a number of limits that cannot be obtained any other way.

As of this edition, the service limit check feature supports programmatically refreshing the "Service Limits" Trusted Advisor check, in order to get updated limit values. If this is not done, the data provided by Trusted Advisor may not be updated unless a human does so via the AWS Console. The refresh logic operates in two major modes, controlled by the time interval set by the feature.

Implementation of Retrieving TA Results



The graph mentioned above describes the implementation of retrieving trusted advisor results, the only way to finish this task is to equip one ImpersonatingClient of AWSSupport with the CreateChangeSet activities. At this point, we extended the functionalities of original createchangesetactivities with several more activities, those newly added activities are focused to check whether the user has the access towards the trusted advisor, then get the service limit check id, implement the refresh logic and finally return the resource limit retails also with the current usage information. On the other side, CreateChangesetWorkflowImpl used as the decider will be in the charge of data flow, client is in charge of managing all those data calculation and transfer. All those process has been finished in a asynchronous way which is based on Amazon SWF framework.

Non exhaustive list of identified code changes

File Name and Package Location	Code Change	Performance impact on existing code flows
pkg/AWS21Workflows:/Config	Added AWSSupportAPIJavaClient	no
pkg/AWS21Workflows:/src/com/amazon/aws21/workflows/changest/CreateChangeSetActivities.java	Added activities interfaces for Trusted Advisor	Yes. More time will be needed for finishing the refresh logic
pkg/AWS21Workflows:/src/com/amazon/aws21/workflows/changest/CreateChangeSetActivitiesImpl.java	Added activities details for Trusted Advisor	Yes. More time will be needed for finishing the refresh logic
pkg/AWS21Workflows:/src/com/amazon/aws21/workflows/parent/CreateChangeSetWorkflowImpl.java	Added implementation for the works in the data flow	No
pkg/AWS21Workflows:/src/com/amazon/aws21/workflows/springconfig/WorkflowConfig.java	Added Bean configuration of Spring Framework for the AWS Support	No

Debugging Process from Test Domain in Amazon SWF

Log into the AWS Console first, then access to aws21-dev+test-beta@amazon account as described below:

AWS Console Access



Choose SWF Dashboard, select the proper domain with the alias_desktop

Workflow Execution: 9c435d39-e733-4384-a57a-a1603726b530

Domain: tenliu_desktop

[Summary](#) | [Events](#) | [Activities](#)

Workflow Type:	_CreateChangeSet (Desktop)
Name (Version):	
Execution:	
Run ID:	22lg/c4X3nG6lVbTKQdls/nkm+fWWKFeZCTHT2xnsUvc=
Workflow ID:	9c435d39-e733-4384-a57a-a1603726b530
Execution Status:	CLOSED
Time Started:	Thursday, August 10, 2017 11:22:34 AM UTC-7
Time Closed:	Thursday, August 10, 2017 11:22:54 AM UTC-7
Closed Status:	Completed
Context:	Not Specified
Latest Activity Task Timestamp:	Thursday, August 10, 2017 11:22:53 AM UTC-7
Tags:	am:aws:cloudformation:us-east-1:319866777625:stack/my/teststacklocal1/bf26ea30-5af7-11e7-b178-06ca05b56900 bf351b00-5af7-11e7-b178-06ca05b56900 am:aws:cloudformation:us-east-1:319866777625:changeSet/l081017local03/9c435d39-e733-4384-a57a-a1603726b530
Task Counts:	
Open Timers:	0
Open Decision Tasks:	0
Open Activity Tasks:	0
Other Info:	
Cancel Requested:	NO
Execution Configuration:	
Child Policy:	Terminate
Task List:	ChangeSet/Desktop
Execution Start To Close Timeout:	4 days
Task Start To Close Timeout:	2 minutes
Task Priority:	0
Lambda Role:	Not Specified
Parent:	
Run ID:	No Parent
Workflow ID:	No Parent

Use the summary page to view the general overview of one workflow.

Domain: tenliu_desktop

[Summary](#) | [Events](#) | [Activities](#)

ID	Event Type
113	WorkflowExecutionCompleted
112	DecisionTaskCompleted
111	DecisionTaskStarted
110	DecisionTaskScheduled
109	ActivityTaskCompleted
108	ActivityTaskStarted
107	ActivityTaskScheduled
106	DecisionTaskCompleted
105	DecisionTaskStarted
104	DecisionTaskScheduled
103	ActivityTaskCompleted
102	ActivityTaskStarted
101	ActivityTaskScheduled
100	DecisionTaskCompleted
99	DecisionTaskStarted
98	DecisionTaskCompleted
97	ActivityTaskCompleted
96	DecisionTaskScheduled
95	ActivityTaskCompleted
94	DecisionTaskStarted
93	DecisionTaskScheduled
92	ActivityTaskCompleted
91	ActivityTaskStarted
90	ActivityTaskCompleted
89	ActivityTaskStarted
88	ActivityTaskCompleted
87	ActivityTaskStarted
86	ActivityTaskCompleted
85	DecisionTaskStarted
84	DecisionTaskCompleted
83	DecisionTaskStarted
82	DecisionTaskScheduled
81	ActivityTaskCompleted
80	ActivityTaskStarted
79	ActivityTaskCompleted
78	ActivityTaskStarted
77	ActivityTaskCompleted
76	ActivityTaskStarted
75	ActivityTaskCompleted
74	ActivityTaskStarted
73	ActivityTaskCompleted
72	ActivityTaskStarted
71	ActivityTaskCompleted
70	ActivityTaskStarted
69	ActivityTaskCompleted
68	ActivityTaskStarted
67	ActivityTaskCompleted
66	ActivityTaskStarted
65	ActivityTaskCompleted
64	ActivityTaskStarted
63	ActivityTaskCompleted
62	ActivityTaskStarted
61	ActivityTaskCompleted
60	ActivityTaskStarted
59	ActivityTaskCompleted
58	ActivityTaskStarted
57	ActivityTaskCompleted
56	ActivityTaskStarted
55	ActivityTaskCompleted
54	ActivityTaskStarted
53	ActivityTaskCompleted
52	ActivityTaskStarted
51	ActivityTaskCompleted
50	ActivityTaskStarted
49	ActivityTaskCompleted
48	ActivityTaskStarted
47	ActivityTaskCompleted
46	ActivityTaskStarted
45	ActivityTaskCompleted
44	ActivityTaskStarted
43	ActivityTaskCompleted
42	ActivityTaskStarted
41	ActivityTaskCompleted
40	ActivityTaskStarted
39	ActivityTaskCompleted
38	ActivityTaskStarted
37	ActivityTaskCompleted
36	ActivityTaskStarted
35	ActivityTaskCompleted
34	ActivityTaskStarted
33	ActivityTaskCompleted
32	ActivityTaskStarted
31	ActivityTaskCompleted
30	ActivityTaskStarted
29	ActivityTaskCompleted
28	ActivityTaskStarted
27	ActivityTaskCompleted
26	ActivityTaskStarted
25	ActivityTaskCompleted
24	ActivityTaskStarted
23	ActivityTaskCompleted
22	ActivityTaskStarted
21	ActivityTaskCompleted
20	ActivityTaskStarted
19	ActivityTaskCompleted
18	ActivityTaskStarted
17	ActivityTaskCompleted
16	ActivityTaskStarted
15	ActivityTaskCompleted
14	ActivityTaskStarted
13	ActivityTaskCompleted
12	ActivityTaskStarted
11	ActivityTaskCompleted
10	ActivityTaskStarted
9	ActivityTaskCompleted
8	ActivityTaskStarted
7	ActivityTaskCompleted
6	ActivityTaskStarted
5	ActivityTaskCompleted
4	ActivityTaskStarted
3	ActivityTaskCompleted
2	ActivityTaskStarted
1	ActivityTaskCompleted
0	ActivityTaskStarted

Apply the Events page to view the details of each events

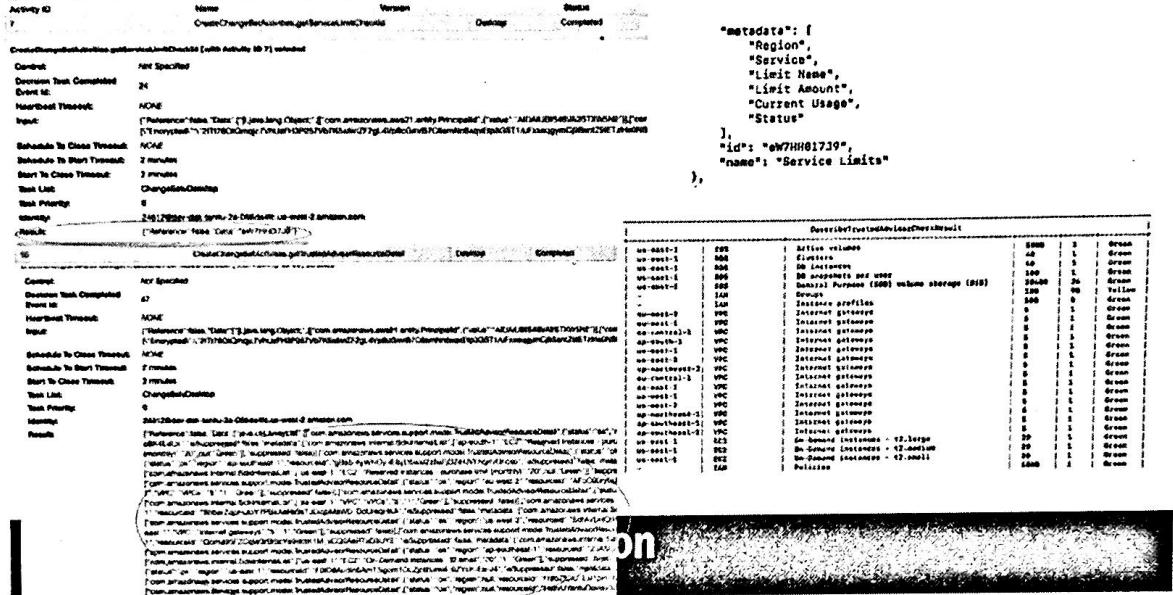
Domain: testlu_desktop					
	Summary	Events	Activities		
Activity ID	Name	Version	Status	Schedule Time	
21	SbsActivities recordWorkflowProgressForChangeSet	Desktop	Completed	Thursday, August 10, 2017 11:22:52 AM UTC-7	
20	SbsActivities putChanges	Desktop	Completed	Thursday, August 10, 2017 11:22:51 AM UTC-7	
17	CreateChangeSetActivities getPartiallyCollapsedTemplateResources	Desktop	Completed	Thursday, August 10, 2017 11:22:49 AM UTC-7	
16	CreateChangeSetActivities getPartiallyCollapsedTemplateResources	Desktop	Completed	Thursday, August 10, 2017 11:22:49 AM UTC-7	
19	CreateChangeSetActivities getResourceDependencies	Desktop	Completed	Thursday, August 10, 2017 11:22:49 AM UTC-7	
18	CreateChangeSetActivities getPartiallyCollapsedTemplateResources	Desktop	Completed	Thursday, August 10, 2017 11:22:40 AM UTC-7	
15	SbsActivities.putTransformedTemplateData	Desktop	Completed	Thursday, August 10, 2017 11:22:38 AM UTC-7	
14	CreateChangeSetActivities.getTrusteeAdviserResourceDetail	Desktop	Completed	Thursday, August 10, 2017 11:22:38 AM UTC-7	
13	CreateChangeSetActivities.getPartiallyCollapsedTemplateResources	Desktop	Completed	Thursday, August 10, 2017 11:22:37 AM UTC-7	
11	KeyValueStoreActivities putTransformedTemplate	Desktop	Completed	Thursday, August 10, 2017 11:22:37 AM UTC-7	
12	CreateChangeSetActivities.getParametersWithValues	Desktop	Completed	Thursday, August 10, 2017 11:22:37 AM UTC-7	
10	CreateChangeSetActivities.getParametersWithValues	Desktop	Completed	Thursday, August 10, 2017 11:22:36 AM UTC-7	
7	CreateChangeSetActivities.getServiceLimitChecklist	Desktop	Completed	Thursday, August 10, 2017 11:22:36 AM UTC-7	
8	TransformActivities.getParametersForTransform	Desktop	Completed	Thursday, August 10, 2017 11:22:36 AM UTC-7	
9	TransformActivities.indirectUnprocessedTransforms	Desktop	Completed	Thursday, August 10, 2017 11:22:36 AM UTC-7	
1	SbsActivities recordWorkflowProgressForChangeSet	Desktop	Completed	Thursday, August 10, 2017 11:22:35 AM UTC-7	
2	SbsActivities.getChangeSetImpersonationData	Desktop	Completed	Thursday, August 10, 2017 11:22:35 AM UTC-7	
3	SbsActivities.getTemplate	Desktop	Completed	Thursday, August 10, 2017 11:22:35 AM UTC-7	
4	KeyValueStoreActivities tryGet	Desktop	Completed	Thursday, August 10, 2017 11:22:35 AM UTC-7	
5	KeyValueStoreActivities tryGet	Desktop	Completed	Thursday, August 10, 2017 11:22:35 AM UTC-7	
6	SbsActivities.setStaticResources	Desktop	Completed	Thursday, August 10, 2017 11:22:35 AM UTC-7	

Apply the Activities page to look at the details for each activity

Domain: testlu_desktop					
	Summary	Events	Activities		
Activity ID	Name	Version	Status	Schedule Time	Start Time
21	SbsActivities recordWorkflowProgressForChangeSet	Desktop	Completed	Thursday, August 10, 2017 11:22:52 AM UTC-7	Thursday, August 10, 2017 11:22
20	SbsActivities putChanges	Desktop	Completed	Thursday, August 10, 2017 11:22:51 AM UTC-7	Thursday, August 10, 2017 11:22
17	CreateChangeSetActivities getPartiallyCollapsedTemplateResources	Desktop	Completed	Thursday, August 10, 2017 11:22:49 AM UTC-7	Thursday, August 10, 2017 11:22
16	CreateChangeSetActivities getPartiallyCollapsedTemplateResources	Desktop	Completed	Thursday, August 10, 2017 11:22:49 AM UTC-7	Thursday, August 10, 2017 11:22
19	CreateChangeSetActivities getResourceDependencies	Desktop	Completed	Thursday, August 10, 2017 11:22:49 AM UTC-7	Thursday, August 10, 2017 11:22
18	CreateChangeSetActivities getPartiallyCollapsedTemplateResources	Desktop	Completed	Thursday, August 10, 2017 11:22:40 AM UTC-7	Thursday, August 10, 2017 11:22
15	SbsActivities.putTransformedTemplateData	Desktop	Completed	Thursday, August 10, 2017 11:22:38 AM UTC-7	Thursday, August 10, 2017 11:22
14	CreateChangeSetActivities.getTrusteeAdviserResourceDetail	Desktop	Completed	Thursday, August 10, 2017 11:22:38 AM UTC-7	Thursday, August 10, 2017 11:22
CreateChangeSetActivities.getPartiallyCollapsedTemplateResources (with Activity 20-14) selected					
Current	Any Specified				
Decision Task Completed					
Event ID	85				
Heartbeat Thread	ACME				
Input:	{Reference: 'Issue' Data: '{'IssueLang': Object, 'J': com.amazonaws.iaw2.IssuePrincipals, 'Issue': 'ADAL02546A2STXW5A2'}}, {com.amazonaws.iaw2.IdentityImpersonationData: {'role': 'IAU', 'null': 'encryptedFile'}, 'Encrypted': 'YVwzOQzGzC8fCfIjZocdDmDlQzA2M72PqQzV8MfOZEL4fDmCkIj34qNfTlqgv-W4zD8fRkfOlyuzLHClBMs+0xen+v51.pCpBfUz78QaLP73fKfMELLwhh3DnYhMkzCPKAJd3mLgPhOThnPlkDfB73';				
Schedule To Close Thread:	ACM				
Schedule To Start Thread:	2 minutes				
Start To Close Thread:	2 minutes				
Task List:	ChangedOnDesktop				
Task Priority:	0				
Workflow:	1104000-testlu-desktop-us-west-2.amazonaws				
Result:	{Reference: 'Issue' Data: '{'IssueLang': Object, 'J': com.amazonaws.iaw2.IssuePrincipals, 'Issue': 'ADAL02546A2STXW5A2'}}, {com.amazonaws.iaw2.IdentityImpersonationData: {'status': 'on', 'region': 'us-east-1', 'role': 'IAU', 'null': 'encryptedFile'}, 'Encrypted': 'YVwzOQzGzC8fCfIjZocdDmDlQzA2M72PqQzV8MfOZEL4fDmCkIj34qNfTlqgv-W4zD8fRkfOlyuzLHClBMs+0xen+v51.pCpBfUz78QaLP73fKfMELLwhh3DnYhMkzCPKAJd3mLgPhOThnPlkDfB73';}				

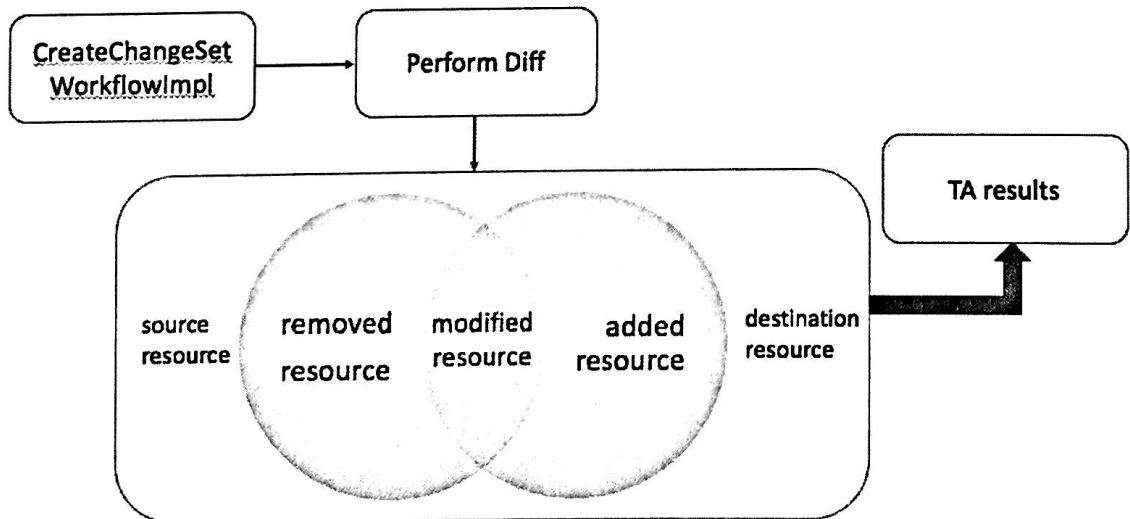
Click the proper activity to view the details of the activity.

Data from SWF Test Domain



This graph shows the data retrieved from the trusted advisor has been successfully returned and utilized in the workflow.

Calculation of Resources Usage



The second part of the service limit check feature is to make the calculation based on upcoming delta value, since at this point we have the

source resource and destination resource, it will be breaking down into three parts: the removed source, modified resource and added resource. For the process of calculation, once it finishes, it will put the information with the expected usage after the execution of this changeset, the service limit information and the warning message to the changes. At this level, one process of retrieving the ta data plus the calculation process has been finished, all of those work is implemented in AWS21Workflows.

Supported Limits From Trusted Advisor Data

The following service limits have been confirmed as being updated from Trusted Advisor:

- CloudFormation
 - Stacks
- EBS
 - Active snapshots
 - Active volumes
 - General Purpose (SSD) volume storage (GiB)
 - Magnetic volume storage (GiB)
 - Provisioned IOPS
 - Provisioned IOPS (SSD) storage (GiB)
- EC2
 - Elastic IP addresses (EIPs)
 - Running On-Demand c3.2xlarge instances

- Running On-Demand c3.4xlarge instances
- Running On-Demand c3.large instances
- Running On-Demand c3.xlarge instances
- Running On-Demand c4.2xlarge instances
- Running On-Demand c4.4xlarge instances
- Running On-Demand c4.large instances
- Running On-Demand c4.xlarge instances
- Running On-Demand m1.small instances
- Running On-Demand m3.2xlarge instances
- Running On-Demand m3.large instances
- Running On-Demand m3.medium instances
- Running On-Demand m3.xlarge instances
- Running On-Demand m4.2xlarge instances
- Running On-Demand m4.large instances
- Running On-Demand m4.xlarge instances
- Running On-Demand r3.2xlarge instances
- Running On-Demand r3.4xlarge instances
- Running On-Demand r3.large instances
- Running On-Demand r3.xlarge instances
- Running On-Demand r4.large instances
- Running On-Demand t1.micro instances
- Running On-Demand t2.large instances
- Running On-Demand t2.medium instances

- Running On-Demand t2.micro instances
 - Running On-Demand t2.nano instances
 - Running On-Demand t2.small instances
 - Running On-Demand t2.xlarge instances
 - VPC Elastic IP addresses (EIPs)
- ELB – Active load balancers
 - IAM
 - Groups
 - Instance profiles
 - Policies
 - Roles
 - Server certificates
 - Users
 - RDS
 - DB Cluster Parameter Groups
 - DB Clusters
 - DB instances
 - DB parameter groups
 - DB security groups
 - DB snapshots per user
 - Event Subscriptions
 - Max auths per security group
 - Read replicas per master

- Storage quota (GB)
- Subnet Groups
- Subnets per Subnet Group

- VPC
 - Internet gateways – VPCs

Note on On-Demand vs Reserved Instances:

The EC2 limits for “Running On-Demand” EC2 Instances apply only to On-Demand instances, not Reserved Instances. If you list all EC2 instances that are running in the Trusted Advisor API, you’ll get back instances of all types (On-Demand, Reserved, etc.). For the AWS CloudFormation with Trusted Advisor, the On-Demand instances are the targets for doing the limit check.

Complicated Cases:

Complicated case 1: The modify process, replacement required



Complicated case 2: Some resources hold different types’ limit information, such as EC2 instance, ebs volume storage, etc

Complicated case 3: Creating of some certain resources will have effect for creating some other resources, ec2 instance with ebs volume storage

There are several complicated cases which still need to be investigated. The biggest issue is from the modified resource, since the modified resource will have a lot of ways to do the modification, if some modified parameter will need to do the replacement, at this point, a new resource type will be first added and the old one will be deleted. For the complicated case 2: some resources hold different types limit information, such as ec2 instance and ebs volume storage which has a bunch of the different types, use the sourceResource and destinationResource to iterate all the properties to find theinstancetype is one way to do a perfect mapping, for the complicated case 3: creating some certain resources will have effect for creating some other resources

Non exhaustive list of identified code changes

File Name and Package Location	Code Change	Performance impact on existing code flows
pkg/AWS21Workflows:/Config	Added AWSSupportAPIJavaClient	no
pkg/AWS21Workflows:/src/com/amazon/aws21/workflows/changese t/CreateChangeSetActivities.java	Added activities interfaces for Trusted Advisor	Yes. More time will be needed for finishing the refresh logic
pkg/AWS21Workflows:/src/com/amazon/aws21/workflows/changese t/CreateChangeSetActivitiesImpl.j ava	Added activities details for Trusted Advisor	Yes. More time will be needed for finishing the refresh logic
pkg/AWS21Workflows:/src/com/amazon/aws21/workflows/parent/Cr eateChangeSetWorkflowImpl.java	Added implementation for the works in the data flow	No

pkg/AWS21Workflows:/src/com/amazon/aws21/workflows/springconfig/WorkflowConfig.java	Added Bean configuration of Spring Framework for the AWS Support	No
pkg/AWS21Workflows:/src/com/amazon/aws21/workflows/parent/CreateChangeSetWorkflowImpl.java	Added the calculation process and computation model for the limit check feature for two resources	No

Visualization Process

In order to visualize the warning message, one more field was added into ResourceChange of the describe change set output, once the calculation process is finished, the AWS21Workflow will allocate StackBuilderService to put the changes with the warning message into describe-changeset-output, then describe change set will present the latest changes with warning message.

The graph showed below is visualization process of the newly added property of ‘resourceLimitInfo’.

Non exhaustive list of identified code changes

File Name and Package Location	Code Change	Performance impact on existing code flows
pkg/AWS21CommonCoralTypes:/model/DataTypes.xml	Added resourceLimitInfo data type	no
pkg/AWS21DataAccess:/src/amzn/aws21/sbs/persistence/model/changeset/ResourceChangeData.java	Added resourceLimitInfo data type	no
pkg/AWS21StackBuilderService:/src/amzn/aws21/sbs/activities/external/DescribeChangeSetActivity.java	Added resourceLimitInfo data type for the output	no

pkg/AWS21StackBuilderService:/ src/amzn/aws21/sbs/activities/internal/PutChangesActivity.java	Put the resourceLimitInfo into the change	no
--	---	----

CR History:

<https://cr.amazon.com/r/7465108/> : [CFN-5111] Add getTrustedAdvisorResourceDetails and getServiceLimitCheckId activities with CloudFormation Stack and IAM Group resource limit check

<https://cr.amazon.com/r/7424561/> : [CFN-5111] Add getTrustedAdvisorDetails and getServiceLimitCheckId activities with CloudFormation Stack and IAM Group resource limit check. Add getTrustedAdvisorDetails and getServiceLimitCheckId activities with CloudFormation Stack and IAM Group resource limit check.

<https://cr.amazon.com/r/7425115/> : Add endpoint information for AWSSupport

<https://cr.amazon.com/r/7509434/> : [CFN-5111] Add a ResourceLimitInfo to ResourceChange of describeChangeSet, and update it from CreateChangeSetWorkflowImpl to reflect the service limit information of ChangeSets by customers