stonebranch

Universal Task Documentation

Universal Automation Center support for scheduling AMAZON S3 file Transfers

ut-aws-s3-monitor-key-in-bucket-linux

Associated Activities:

Date: 10 April 2018

Author: Nils Buer

Revision: 01

CONFIDENTIALITY INFORMATION

Distribution list: Stonebranch Marketplace

Revision	Date	Author	Changes
00	20180405	Nils Buer	Initial Document (WIP)
01	20180410	Nils Buer	Test Cases Added for key download

Abstract:

The here described Universal Tasks allow to Transfer and retrieve files from Amazon AWS S3. As a result, you can integrate any AWS S3 file transfers into you existing or new scheduling workflows, providing a true hybrid cloud (on-premise and cloud computer) file transfer solution.

Contents

1 Disclaimer

No support and no warranty are provided by Stonebranch GmbH for this document and the related Universal Task. The use of this document and the related Universal Task is on your own risk.

Before using this task in a production system, please perform extensive testing.

Stonebranch GmbH assumes no liability for damage caused by the performance of the Universal Tasks

2 Scope

This document provides a documentation how to install and use the Universal Tasks for AWS S3 File Transfers. If more Task will be created in the future this document will be updated accordingly.

3 Introduction

Storing data in the cloud becomes an integral part of most modern IT landscapes. With Universal Automation Center you can securely automate your AWS, Azure or any other Cloud File Transfer and integrate them into your existing scheduling flows.

As security is one of the key concerns, when moving to the cloud, the provided solution supports multi-level of security:

- All Credential for AWS S3 (Access Key, Secret Access key and Region) are stored in an encrypted form in the database
- Connections towards the AWS VPC via a Proxy Server are supported
- Secure access to AWS S3 buckets using *AWS bucket policies* can be configured
- Restrict sending files only to specific buckets using *AWS End Points*

The here described Series of Universal Tasks focuses on the AMAZON AWS S3 file transfer. A similar solution as for AWS S3 is also available for Microsoft Azure.

Some details about the universal tasks for AWS S3:

- The Universal Tasks are calling the python module Boto3 - the Amazon Web Services (AWS) SDK for Python. Both Boto3 API Types are used the "low-level" Client API and Resource APIs.
- The python boto3 module is called by a Universal Agent running on a Linux Server or Windows Server – Note: This document focuses on the Linux Version
- The Server Running the Universal Agent needs to have Python 2.7.x or 3.6.x installed
- All Universal Task support encrypted connections via a Proxy Server
- All Credential for AWS S3 (Access Key, Secret Access key and Region) are stored in an encrypted form in the database
- You can configure all connection Parameters for the Proxy and AWS via the Universal Task
- You can select different log-levels e.g. Info and debug

The following Universal Task for AWS S3 have been implemented:

Command	UT Name	Description
Create_Bucket	ut_AWS_S3_Create_Bucket_linux	Creates a bucket in AWS S3
Monitor_key_in_bucket	ut_AWS_S3_Monitor_key_in_bucket_linux	Monitors at a given interval for a key in a bucket
Copy_file_to_bucket	ut_AWS_S3_Copy_file_to_bucket_linux	Copies a file to a bucket
List_keys_in_bucket	ut_AWS_S3_List_keys_in_bucket_linux	Lists all keys in a bucket
Download_file_from_bucket	ut_AWS_S3_Download_file_from_bucket_linux	Downloads an AWS S3 key to a local file
Delete_key_from_bucket	ut_AWS_S3_Delete_key_from_bucket_linux	Deletes a key from a bucket
Delete_empty_bucket	ut_AWS_S3_Delete_empty_bucket_linux	Deletes an empty bucket
List_buckets	ut_AWS_S3_List_buckets_linux	List all buckets of an AWS account
Check_for_key_in_buckets	ut_AWS_S3_Check_for_key_in_buckets_linux	Checks for the existence of a key in a bucket

4 Installation

4.1 Software Requirements

Universal Task name: *ut_AWS_S3_<xxx>_bucket_linux*

Related UAC XML Files for template and task: *Github repository*

Software used:

For the set-up you need:

1. Python 2.7.x (or 3.6.x) for Linux installed on a server where a Universal Agent is installed.
2. For Python the following modules are required:

- ***Re,*** to support regular expression matching operations
- ***glob,*** *to* find Unix pathnames matching a specified pattern
- ***os,*** to support operating system dependent commands
- ***sys***, for output re-direct processing
- ***datetime, date*** and time stamps for messages
- ***logging***, to provide logging capabilities for debug, info etc.
- ***boto3,*** provide the Amazon Web Services (AWS) SDK for Python
- ***botocore,*** Botocore is a low-level interface to a growing number of Amazon Web Services. Botocore serves as the foundation for the AWS-CLI command line utilities.
- ***argparse***, to allow testing of the Universal TPL. script on the command line

*Note: Only the module **boto3** and **botocore** need to be added to python 3.6.x. e.g. using pip.*

- *pip install boto3*
- *pip install botocore*

3. Universal Controller 6.4.5.x or higher
4. Universal Agent 6.4.2.2 or higher installed on a Linux Server
5. An Amazon AWS S3 account to try it out

4.2 Installation Steps

The following describes the installation steps:

1. **Install Python 2.7.x or 3.6.x for Linux on the Universal Controller server or any Linux Server running a Universal Agent.**

   Official Download link: https://www.python.org/downloads/

   Note:

   In most cases python is already available on Linux. Check availability with: *python -V*

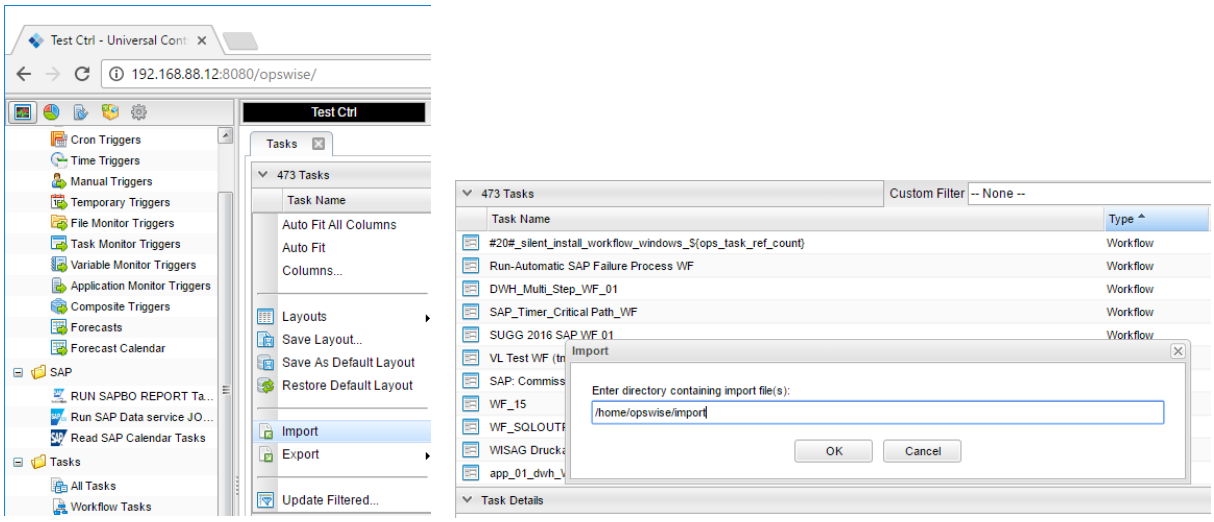
2. **Add the boto3 and botocore modules to your python installation**

   In a command shell run as root or sudo:

   - *pip install boto3*
   - *pip install botocore*

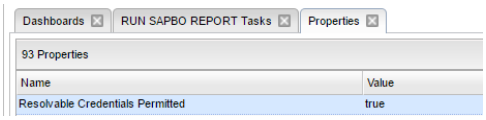
3. **Import each AWS S3 Universal Task including the Universal Template to your Controller**

Go to “All Tasks” and load via the Import functionality the Universal Task configuration into the Controller.



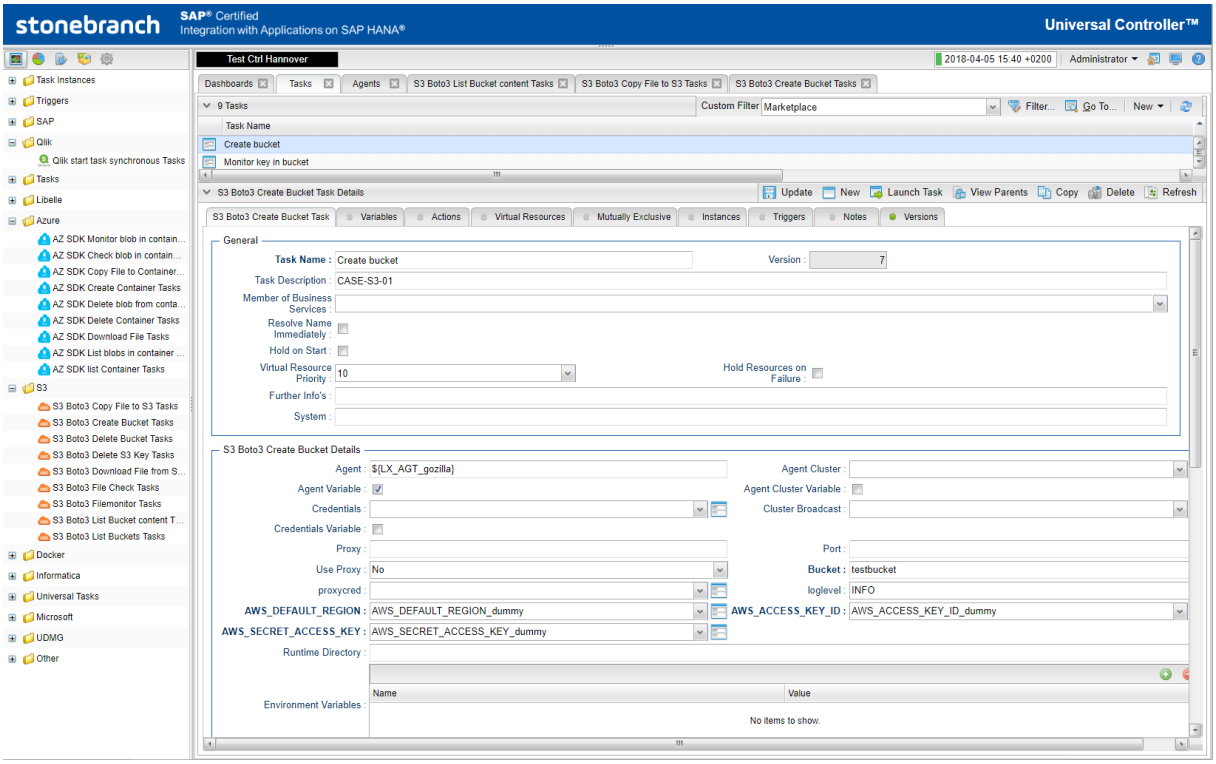
5 Universal Task Configuration

1. Activate: Resolvable Credentials in Universal Automation Center:



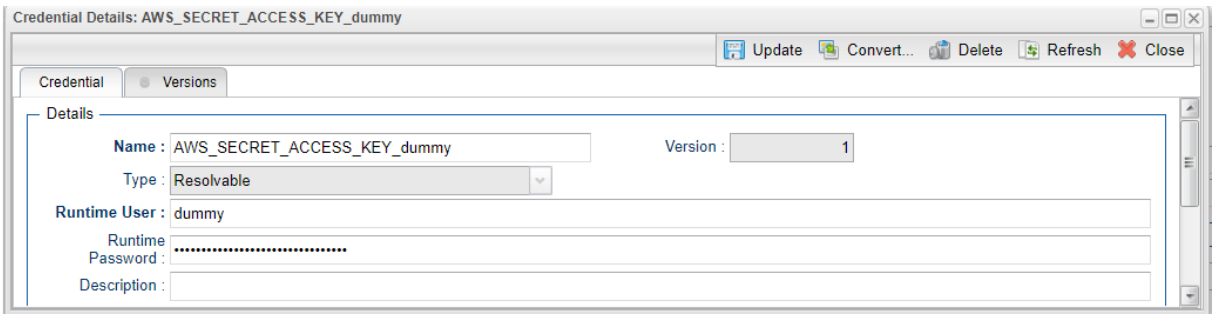
2. Fill Out the Universal Task for each AWS command, which you want to execute:

In the example below the *S3 Boto3 Create Bucket Task* was selected



Fill out or select the required Credentials for AWS and optionally a Proxy Server

In the example below the *AWS_SECRET_ACCESS_KEY* credentials are shown:



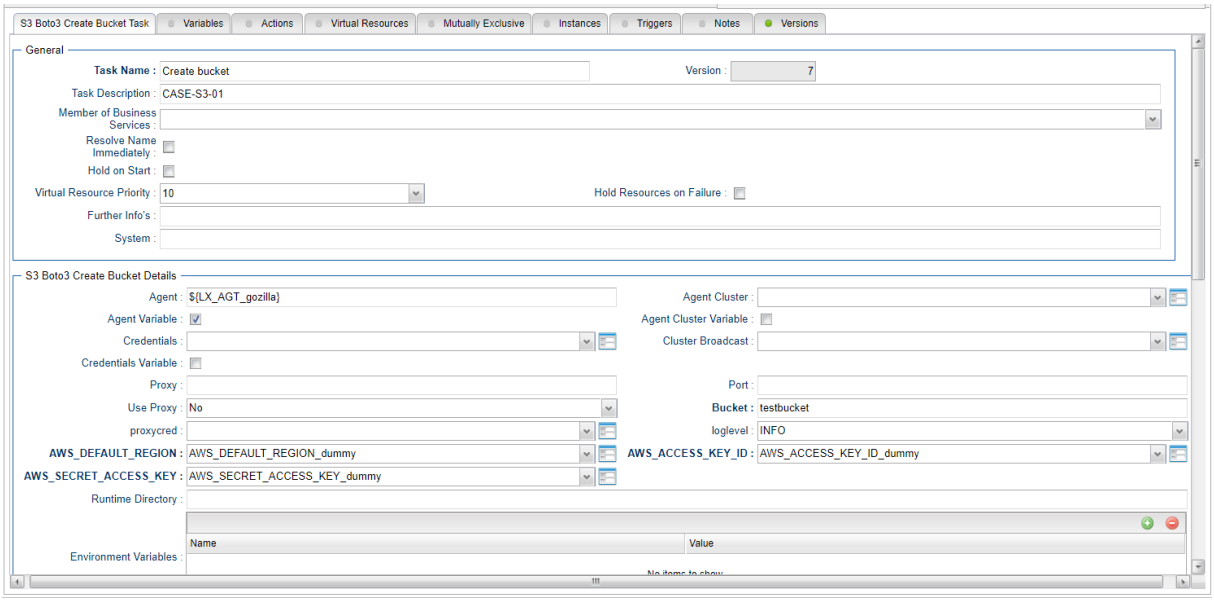
6 Universal Tasks for AWS S3

The following chapter describes the provided AWS S3 Universal Tasks.

6.1 Create_Bucket

Command	UT Name	Description
Create_Bucket	ut_AWS_S3_Create_Bucket_linux	Creates a bucket in AWS S3

Task Screenshot:



Field Description:

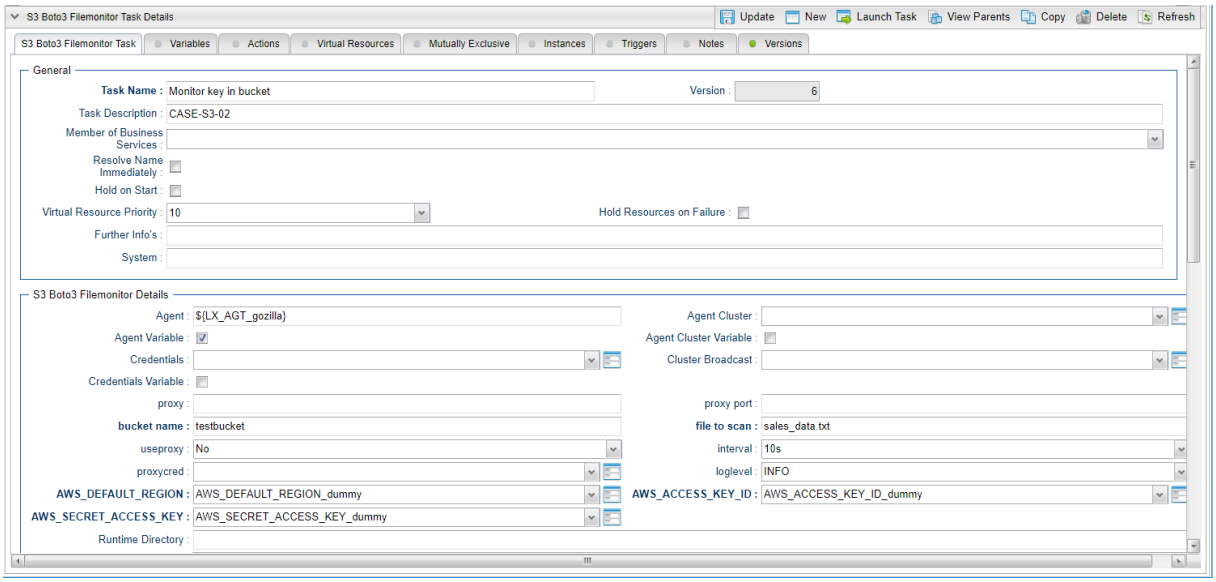
Field	Required	Description
Agent	Mandatory	The Linux Universal Agent, which runs the Python boto3 module to call the AWS S3 commands
Credentials	Optional	The Credentials used on the Linux Server
Use Proxy (yes/no)	Mandatory	Decide If a Proxy Server should be used yes or no (If “no” is selected the fields Proxy,Port and proxycred are ignored)
Proxy	Optional	Proxy Server IP or hostname (only used in case Use Proxy = yes)
proxycred	Optional	Proxy Server Credentials (only used in case Use Proxy = “yes” is selected)
Port	Optional	Proxy Server Port (only used in case Use Proxy = yes)
loglevel	Mandatory	logging settings DEBUG, INFO, WARNING, ERROR, CRITICAL

AWS_DEFAULT_REGION	Mandatory	AWS Region kept as credential
AWS_SECRET_ACCESS_KEY	Mandatory	AWS Secret Key
AWS_ACCESS_KEY_ID	Mandatory	AWS Access Key
Bucket	Mandatory	Bucket to created

6.2 Monitor_key_in_bucket

Command	UT Name	Description
Monitor_key_in_bucket	ut_AWS_S3_Monitor_key_in_bucket_linux	Monitors at a given interval for a key in a bucket

Task Screenshot:



Field Description:

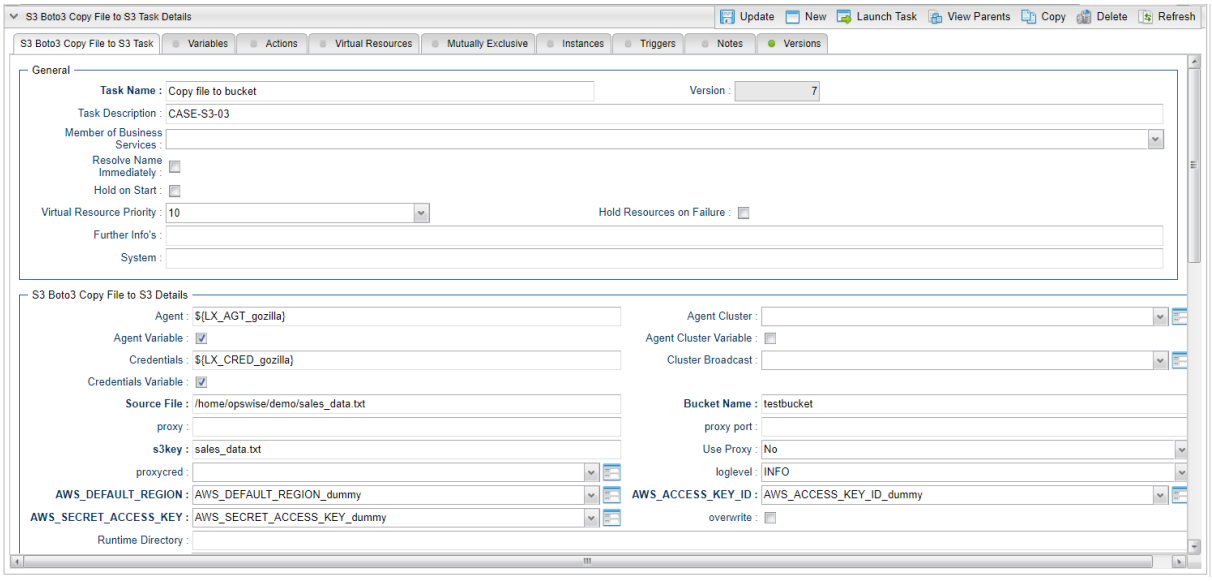
Field	Required	Description
Agent	Mandatory	The Linux Universal Agent, which runs the Python boto3 module to call the AWS S3 commands
Credentials	Optional	The Credentials used on the Linux Server
Use Proxy (yes/no)	Mandatory	Decide If a Proxy Server should be used yes or no (If “no” is selected the fields Proxy,Port and proxycred are ignored)
Proxy	Optional	Proxy Server IP or hostname (only used in case Use Proxy = yes)
proxycred	Optional	Proxy Server Credentials (only used in case Use Proxy = “yes” is selected)

Port	Optional	Proxy Server Port (only used in case Use Proxy = yes)
loglevel	Mandatory	logging settings DEBUG, INFO, WARNING, ERROR, CRITICAL
AWS_DEFAULT_REGION	Mandatory	AWS Region kept as credential
AWS_SECRET_ACCESS_KEY	Mandatory	AWS Secret Key
AWS_ACCESS_KEY_ID	Mandatory	AWS Access Key
File to scan	Mandatory	AWS key (=file) to scan in the given bucket
Bucket name	Mandatory	AWS Bucket to scan for the given AWS key (=file)
interval	Mandatory	Scan Interval

6.3 Copy_file_to_bucket

Command	UT Name	Description
Copy_file_to_bucket	ut_AWS_S3_Copy_file_to_bucket_linux	Copies a file to a bucket

Task Screenshot:



Field Description:

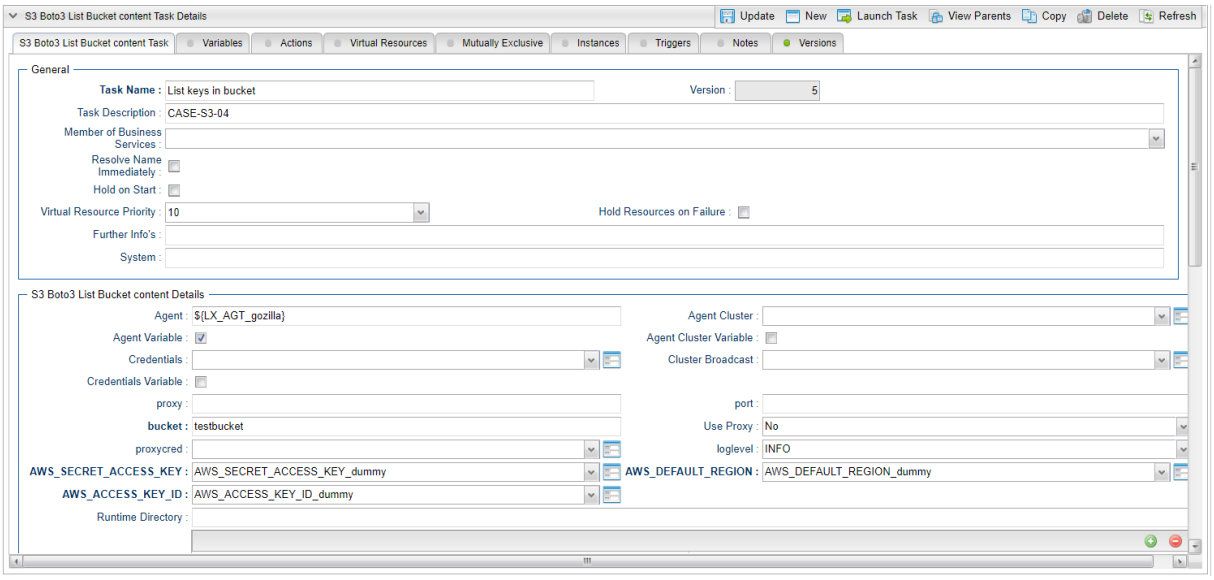
Field	Required	Description
Agent	Mandatory	The Linux Universal Agent, which runs the Python boto3 module to call the AWS S3 commands
Credentials	Optional	The Credentials used on the Linux Server

Use Proxy (yes/no)	Mandatory	Decide If a Proxy Server should be used yes or no (If “no” is selected the fields Proxy,Port and proxycred are ignored)
Proxy	Optional	Proxy Server IP or hostname (only used in case Use Proxy = yes)
proxycred	Optional	Proxy Server Credentials (only used in case Use Proxy = “yes” is selected)
Port	Optional	Proxy Server Port (only used in case Use Proxy = yes)
loglevel	Mandatory	logging settings DEBUG, INFO, WARNING, ERROR, CRITICAL
AWS_DEFAULT_REGION	Mandatory	AWS Region kept as credential
AWS_SECRET_ACCESS_KEY	Mandatory	AWS Secret Key
AWS_ACCESS_KEY_ID	Mandatory	AWS Access Key
Bucket Name	Optional	Name of the target bucket
overwrite	Mandatory	If checked, allow to overwrite an existing key in the bucket
S3key	Mandatory	S3 key to copy to the given bucket

6.4 List_keys_in_bucket

Command	UT Name	Description
List_keys_in_bucket	ut_AWS_S3_List_keys_in_bucket_linux	Lists all keys in a bucket

Task Screenshot:



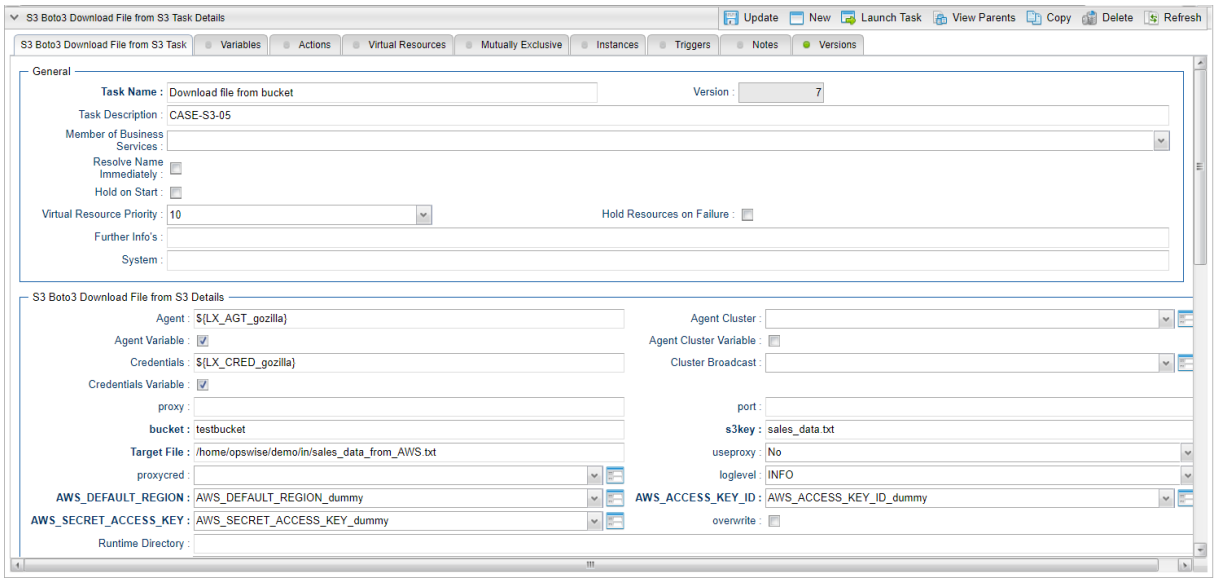
Field Description:

Field	Required	Description
Agent	Mandatory	The Linux Universal Agent, which runs the Python boto3 module to call the AWS S3 commands
Credentials	Optional	The Credentials used on the Linux Server
Use Proxy (yes/no)	Mandatory	Decide If a Proxy Server should be used yes or no (If “no” is selected the fields Proxy,Port and proxycred are ignored)
Proxy	Optional	Proxy Server IP or hostname (only used in case Use Proxy = yes)
proxycred	Optional	Proxy Server Credentials (only used in case Use Proxy = “yes” is selected)
Port	Optional	Proxy Server Port (only used in case Use Proxy = yes)
loglevel	Mandatory	logging settings DEBUG, INFO, WARNING, ERROR, CRITICAL
AWS_DEFAULT_REGION	Mandatory	AWS Region kept as credential
AWS_SECRET_ACCESS_KEY	Mandatory	AWS Secret Key
AWS_ACCESS_KEY_ID	Mandatory	AWS Access Key
bucket	Mandatory	List all AWS keys in the bucket

6.5 Download_file_from_bucket

Command	UT Name	Description
Download_file_from_bucket	ut_AWS_S3_Download_file_from_bucket_linux	Downloads an AWS S3 key to a local file

Task Screenshot:



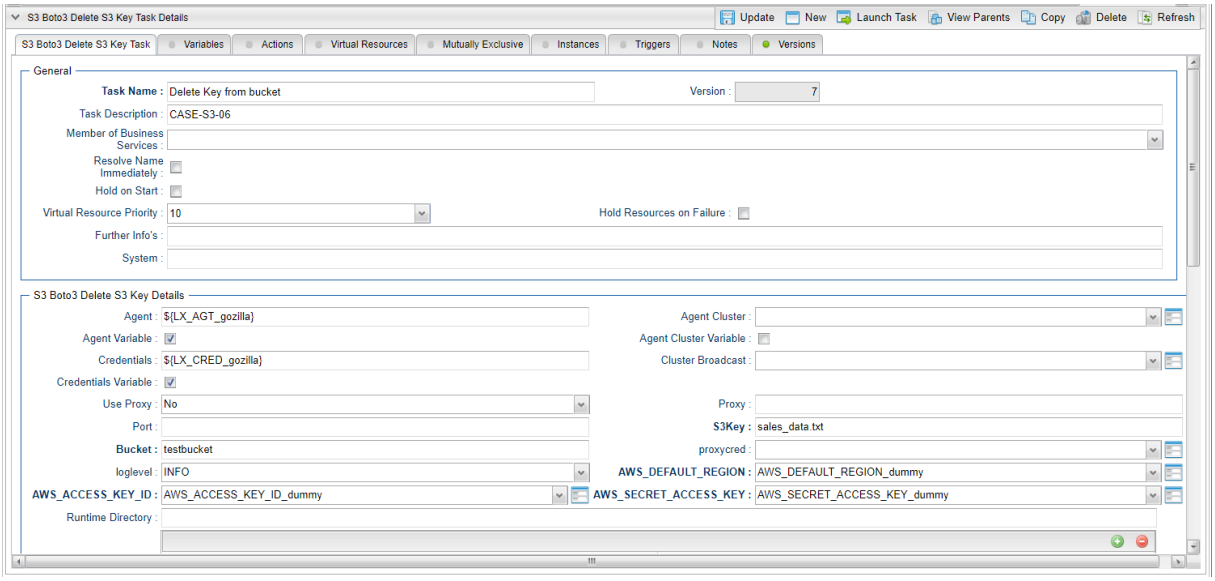
Field Description:

Field	Required	Description
Agent	Mandatory	The Linux Universal Agent, which runs the Python boto3 module to call the AWS S3 commands
Credentials	Optional	The Credentials used on the Linux Server
Use Proxy (yes/no)	Mandatory	Decide If a Proxy Server should be used yes or no (If “no” is selected the fields Proxy,Port and proxycred are ignored)
Proxy	Optional	Proxy Server IP or hostname (only used in case Use Proxy = yes)
proxycred	Optional	Proxy Server Credentials (only used in case Use Proxy = “yes” is selected)
Port	Optional	Proxy Server Port (only used in case Use Proxy = yes)
loglevel	Mandatory	logging settings DEBUG, INFO, WARNING, ERROR, CRITICAL
AWS_DEFAULT_REGION	Mandatory	AWS Region kept as credential
AWS_SECRET_ACCESS_KEY	Mandatory	AWS Secret Key
AWS_ACCESS_KEY_ID	Mandatory	AWS Access Key
Target File	Mandatory	Target file name and directory e.g. */home/opswise/demo/in/sales_data_from_AWS.txt*
S3key	Mandatory	Source Key file to download
bucket	Mandatory	Source bucket

6.6 Delete_key_from_bucket

Command	UT Name	Description
Delete_key_from_bucket	ut_AWS_S3_Delete_key_from_bucket_linux	Deletes a key from a bucket

Task Screenshot:



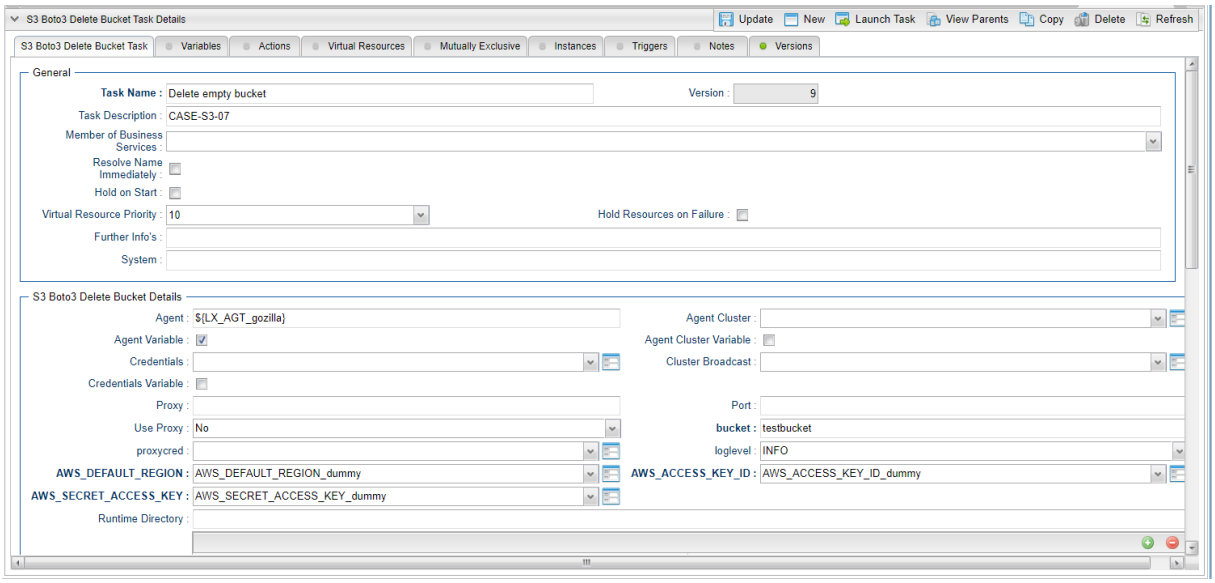
Field Description:

Field	Required	Description
Agent	Mandatory	The Linux Universal Agent, which runs the Python boto3 module to call the AWS S3 commands
Credentials	Optional	The Credentials used on the Linux Server
Use Proxy (yes/no)	Mandatory	Decide If a Proxy Server should be used yes or no (If “no” is selected the fields Proxy,Port and proxycred are ignored)
Proxy	Optional	Proxy Server IP/ hostname (used if Use Proxy = “yes”)
proxycred	Optional	Proxy Server Credentials (used if Use Proxy = “yes”)
Port	Optional	Proxy Server Port (only used in case Use Proxy = yes)
loglevel	Mandatory	logging settings DEBUG, INFO, WARNING, ERROR, CRITICAL
AWS_DEFAULT_REGION	Mandatory	AWS Region kept as credential
AWS_SECRET_ACCESS_KEY	Mandatory	AWS Secret Key
AWS_ACCESS_KEY_ID	Mandatory	AWS Access Key
S3key	Mandatory	Key to delete from the given bucket
Bucket	Mandatory	Bucket, which contains the key to delete

6.7 Delete_empty_bucket

Command	UT Name	Description
Delete_empty_bucket	ut_AWS_S3_Delete_empty_bucket_linux	Deletes an empty bucket

Task Screenshot:



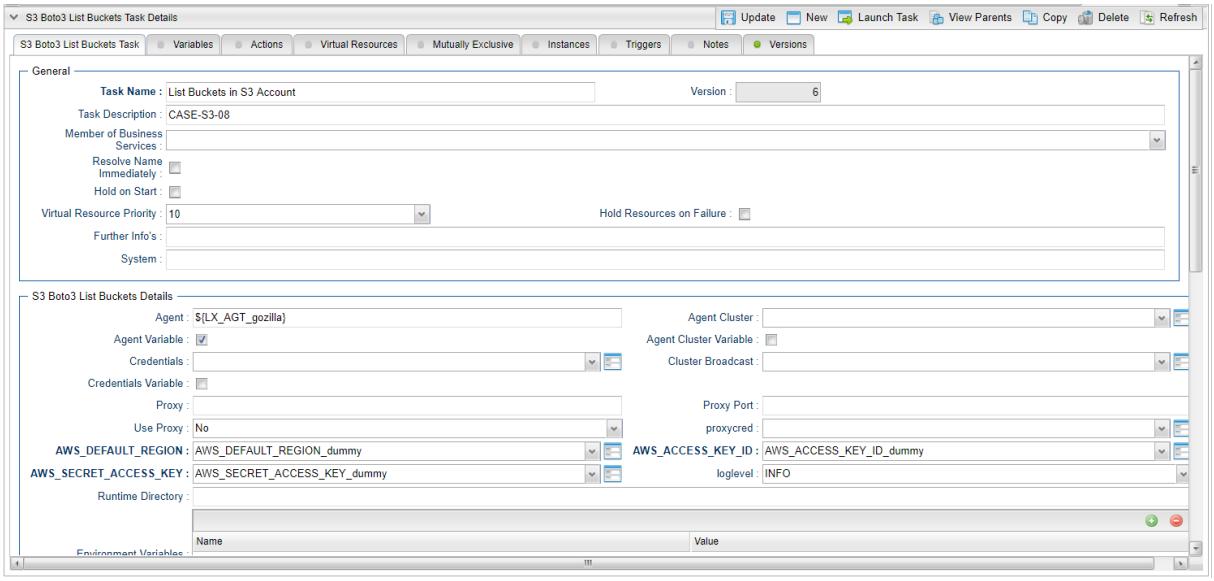
Field Description:

Field	Required	Description
Agent	Mandatory	The Linux Universal Agent, which runs the Python boto3 module to call the AWS S3 commands
Credentials	Optional	The Credentials used on the Linux Server
Use Proxy (yes/no)	Mandatory	Decide If a Proxy Server should be used yes or no (If “no” is selected the fields Proxy,Port and proxycred are ignored)
Proxy	Optional	Proxy Server IP/ hostname (used if Use Proxy = “yes”)
proxycred	Optional	Proxy Server Credentials (used if Use Proxy = “yes”)
Port	Optional	Proxy Server Port (only used in case Use Proxy = yes)
loglevel	Mandatory	logging settings DEBUG, INFO, WARNING, ERROR, CRITICAL
AWS_DEFAULT_REGION	Mandatory	AWS Region kept as credential
AWS_SECRET_ACCESS_KEY	Mandatory	AWS Secret Key
AWS_ACCESS_KEY_ID	Mandatory	AWS Access Key
bucket	Mandatory	Bucket to delete

6.8 List_buckets

Command	UT Name	Description
List_buckets	ut_AWS_S3_List_buckets_linux	List all buckets of an AWS account

Task Screenshot:



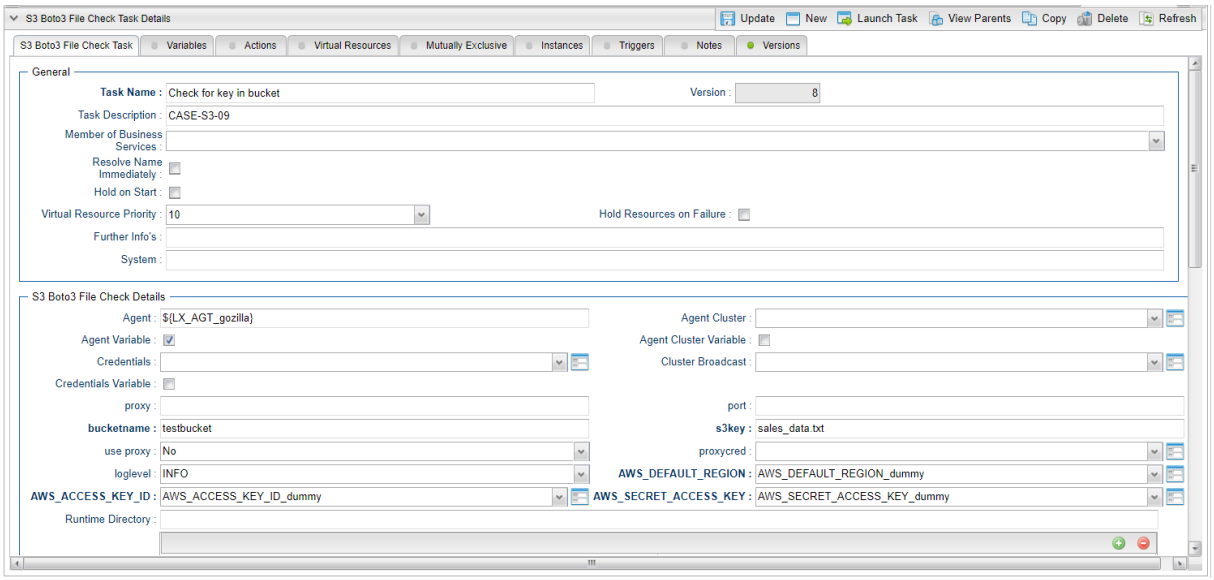
Field Description:

Field	Required	Description
Agent	Mandatory	The Linux Universal Agent, which runs the Python boto3 module to call the AWS S3 commands
Credentials	Optional	The Credentials used on the Linux Server
Proxy	Optional	Proxy Server IP/ hostname (used if Use Proxy = “yes”)
proxycred	Optional	Proxy Server Credentials (used if Use Proxy = “yes”)
Port	Optional	Proxy Server Port (only used in case Use Proxy = yes)
Port	Optional	Proxy Server Port (only used in case Use Proxy = yes)
loglevel	Mandatory	logging settings DEBUG, INFO, WARNING, ERROR, CRITICAL
AWS_DEFAULT_REGION	Mandatory	AWS Region kept as credential
AWS_SECRET_ACCESS_KEY	Mandatory	AWS Secret Key
AWS_ACCESS_KEY_ID	Mandatory	AWS Access Key

6.9 Check_for_key_in_buckets

Command	UT Name	Description
Check_for_key_in_buckets	ut_AWS_S3_Check_for_key_in_buckets_linux	Checks for the existence of a key in a bucket

Task Screenshot:



Field Description:

Field	Required	Description
Agent	Mandatory	The Linux Universal Agent, which runs the Python boto3 module to call the AWS S3 commands
Credentials	Optional	The Credentials used on the Linux Server
Use Proxy (yes/no)	Mandatory	Decide If a Proxy Server should be used yes or no (If “no” Proxy, Port and proxycred are ignored)
Proxy	Optional	Proxy Server IP/ hostname (used if Use Proxy = “yes”)
proxycred	Optional	Proxy Server Credentials (used if Use Proxy = “yes”)
Port	Optional	Proxy Server Port (only used in case Use Proxy = yes)
loglevel	Mandatory	logging settings DEBUG, INFO, WARNING, ERROR, CRITICAL
AWS_DEFAULT_REGION	Mandatory	AWS Region kept as credential
AWS_SECRET_ACCESS_KEY	Mandatory	AWS Secret Key
AWS_ACCESS_KEY_ID	Mandatory	AWS Access Key
S3key	Mandatory	Key to check for in the given bucket
bucketname	Mandatory	Bucket to check for the given key

7 Test Cases

The following basic test cases has been performed:

Case#	Assumed behavior	Result
Creates a bucket in AWS S3	Log message: INFO - bucket: *testbucket* created.	Correct
Creates a bucket in AWS S3 (bucket already exists)	Log message: ERROR - bucket: *testbucket* already exist	Correct
Monitors at a given interval (10s) for a key in a bucket	If the key is not available Monitor stays in running status. If the key is available Monitor goes to success Log message: INFO - Object: sales_data.txt found in bucket: *testbucket*	Correct
Copies a file to a bucket, if the key exists (flag overwrite is set)	Log message: INFO - Key: sales_data.txt exists and will be overwritten due to overwrite flag set INFO - Starting new HTTPS connection (1): s3.amazonaws.com	Correct
Copies a file to a bucket, if the key exists (flag overwrite is not set)	Log message: ERROR - Key: sales_data.txt exists and will not be overwritten, set overwrite flag	Correct
Copies a file to a bucket (file does not exist in bucket)	Log message INFO - Finished uplodaing file: /home/opswise/demo/sales_data.txt, to bucket: testbucket as key: sales_data.txt	Correct
Lists all keys in a bucket	Keys are listed in stdout: *sales_data.txt*	Correct
Downloads an AWS S3 key to a local file if file does not yet exist	Log message: INFO - Downloading File: sales_data.txt ,from bucket: testbucket to file: /home/opswise/demo/in/sales_data_from_AWS.txt finished	Correct

Downloads an AWS S3 key to a local file if already exist (flag overwrite is set)	Log message: INFO - File: /home/opswise/demo/in/sales_data_from_AWS.txt exists and will be overwritten due to overwrite flag set INFO - Downloading File: sales_data.txt ,from bucket: axa-secure-8 to file: /home/opswise/demo/in/sales_data_from_AWS.txt finished	Correct
Downloads an AWS S3 key to a local file if already exist (flag overwrite is not set)	Log message: ERROR - File: /home/opswise/demo/in/sales_data_from_AWS.txt exists and will not be overwritten, set overwrite flag	Correct
Downloads an AWS S3 key to a local file if file does not yet exist and overwrite flag is set.	Log message: INFO - Downloading File: sales_data.txt ,from bucket: testbucket to file: /home/opswise/demo/in/sales_data_from_AWS.txt finished	Correct
Deletes a key from a bucket	Log message: INFO - : Key deleted: sales_data.txt, from Bucket: *testbucket*	Correct
Deletes a key from a bucket (key does not exist)	Log message: ERROR - The object: sales_data.txt does not exists	
Deletes an empty bucket	Log message: INFO - bucket *testbucket* deleted	Correct
Deletes a not empty bucket	Log message: INFO - bucket *testbucket* does not exist INFO - bucket *testbucket* is not empty	Correct
Deletes a bucket, which does not exist	Log message: INFO - bucket *testbucket* does not exist	Correct
List all buckets of an AWS account	Log message: List of buckets is displayed	Correct
Checks for the existence of a key in a bucket	Log message: INFO - Object: sales_data.txt found in bucket: *testbucket*	Correct

Checks for the existence of a key in a bucket, if key does not exist	Log message: ERROR - Object: sales_dataa.txt not found in bucket: *testbucket*	Correct

8 Document References

There are no document references.