



# CTX-Backup-Flows User Guide

## Contents

---

CTX-Backup-Flows User Guide .....	1
Contents .....	2
Versions .....	4
Document Revisions .....	4
Module Versions .....	4
Preface .....	5
About this Manual .....	5
Audience .....	5
Related Material .....	5
Abbreviations used in this Document .....	5
Requirements .....	6
Integration .....	7
Integration with Third-Party Systems .....	7
Integration with Existing Infrastructure .....	7
1 Backup Flows Overview .....	8
1.1 Backup Types .....	8
1.1.1 Backup Flow(s) .....	9
1.1.2 Backup Group(s) .....	9
1.1.3 Backup System .....	9
1.2 Retention period .....	9
1.3 Config files .....	10
1.3.1 General config file .....	10
1.3.2 Flows config file .....	12
1.3.3 Groups config file .....	14
1.3.4 System config file .....	16
1.4 Using the Module .....	17
1.4.1 Global variables to set .....	17
1.4.2 Config Files .....	17
1.4.3 How to Create backups .....	18
1.4.4 Exception Handling .....	19
2 Backup Module Flows .....	20
2.1 BF-Backup-Flows-From-Config .....	20
2.1.1 Overview .....	20
2.2 BF-Backup-Flows .....	20
2.2.1 Overview .....	20
3 Backup Module Subtasks .....	21

3.1	BF-GCAT-Get-Cortex-Authentication-Token.....	21
3.1.1	Overview .....	21
3.1.2	Input variables.....	21
3.1.3	Output variables .....	21
3.2	BF-PPR-Process-PowerShell-Response .....	22
3.2.1	Overview .....	22
3.2.2	Input variables.....	22
3.2.3	Output variables .....	22
3.3	BF-RMFFB-Remove-Missing-Flow-from-Backup.....	23
3.3.1	Overview .....	23
3.3.2	Input variables.....	23
3.3.3	Output variables .....	23

## Versions

---

### Document Revisions

The following revisions have been made to this document

Date	Revision	Notes
07/03/2019	1.0	First Release
05/08/2021	1.1	Expanding on config files requirements
27/09/2021	1.2	Reformatted Removed references to email module

### Module Versions

The following revisions have been made to this document

Module Version	Release Date	Comments
1.0	08/03/2019	Creation of: * BF-Backup-Flows-From-Config * BF-Backup-Flows * BF-GCAT-Get-Cortex-Authentication-Token * BF-PPR-Process-PowerShell-Response * BF-RMFFB-Remove-Missing-Flow-from-Backup
1.1	5/8/2021	Updated: * BF-Backup-Flows-From-Config * BF-Backup-Flows * BF-GCAT-Get-Cortex-Authentication-Token
1.2	05/10/2021	Updated: * BF-Backup-Flows-From-Config * BF-Backup-Flows * BF-GCAT-Get-Cortex-Authentication-Token

## Preface

---

### About this Manual

This document is a user guide for the CTX-Backup-Flows module.

### Audience

The audience for this document is those wanting to understand how to configure and use CTX-Backup-Flows module.

### Related Material

Document
CTX-Backup-Flows – Deployment Plan
CTX-Backup-Flows.studiopkg

### Abbreviations used in this Document

<b>DB</b>	Database
<b>JSON</b>	JavaScript Object Notation
<b>OCI</b>	Orchestration Communication Interface

## Requirements

---

The CTX-Backup-Flows module requires the following:

- Minimum Cortex v7.1 installed on the Cortex Application Server
- Cortex PowerShell OCI

## Integration

---

### Integration with Third-Party Systems

None

### Integration with Existing Infrastructure

To use this module, Cortex needs to be able to connect to the 'CortexWeb' DB which stores details of all the flows within Gateway. For this to work, the service account set to run the "Cortex Database Interface service" must be granted Read permissions to that DB.

For information on how to do this, please see Section 4 of the CTX-Backup-Flows Deployment Guide.

## 1 Backup Flows Overview

---

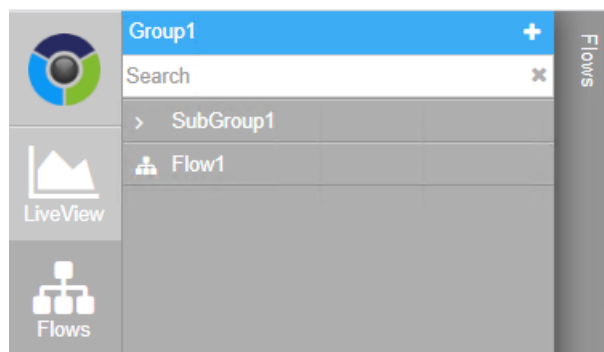
The CTX-Backup-Flows module is intended to provide automated backing up of flows from Cortex environments. It is possible to back up single or multiple specified flows, groups or multiple specified groups, or the entire system.

The flow should run on a schedule and check which items are due to be backed up. The flow will also handle clearing out any old backups if appropriate, based on the specified data retention period.

### 1.1 Backup Types

3 different backup types are supported, and are executed based on the existence of a config file for that type existing on the server.

- **Backup Flows** - One or more Cortex flows
- **Backup Groups** - Backup of one or more Cortex groups, including all flows and subgroups within the group



*A Group containing Sub-Groups and Flows in Cortex Gateway*

- **Backup System** - Backup of all flows on a Cortex system

How to configure these and other customisations are detailed later in the document.



### 1.1.1 Backup Flow(s)

This configuration allows the user to specify a Flow / Subtask, or a list of Flows / Subtasks to backup.

In the event of an exception, any individual flows which cause an issue will be excluded and the backup will be re-run.

### 1.1.2 Backup Group(s)

This configuration allows the user to specify a group or list of groups to backup. Any Flows / Subtasks within this group will be backed up.

On the event of an exception, any individual flows which cause an issue will be excluded and the entire backup process will be re-run.

### 1.1.3 Backup System

This configuration will back up every flow and subtask in Cortex Studio.

On the event of an exception when backing up a flow, any flow which cause an exception will be excluded, and the backup will be re-run.

## 1.2 Retention period

In the general config file, a parameter named RetentionPeriod is set to the number of days a backup should be kept for. When the backup flow is run, any backup studio packages older than the retention period will be replaced with a new backup studio package.

## 1.3 Config files

The General config file, described in detail below, is required. Also, at least one of the Flows, Groups or System config files are also required. This means there will be between two and four config files on the application server to be backed up.

### 1.3.1 General config file

The general config file should be named 'GeneralConfig.json' and put onto the Cortex server. The filepath of the general config file should be stored in the Global Variable 'g\_backup-config-path', located in the flow 'BF-Backup-Flows-From-Config'. This config file is required.

#### 1.3.1.1 General config parameters

Config Parameter	Details	Example
GatewaySourceHost	The Cortex Gateway URL	https://CTXAppServer/gateway
SQLServer	The fully qualified Database Server Name	CTXDBServer.MyDomain.com
SQLUser	The Cortex Database Interface User	MyDomain\\CTX_DB  Note: json format requires escaping of backslashes, i.e. the double backslash represents a single backslash character in this format
SQLPass	The Password for SQLUser. This may be encrypted using the Cortex Encryptor if necessary, found in c:\Cortex\admin by default on a Cortex application server	Encrypted: #_213207061128133!076198044053108150073058139151206~060068237177!225210117133027107049186117027253#  Non-Encrypted: CTXPA55W0rd
BackupConfigPath	Full path to where either flows, groups or system config file is stored	C:\\Backup\\Config  Note: json format requires escaping of backslashes, i.e. the double backslash represents a single backslash character in this format
RetentionPeriod	The number of days to keep backups for	30

### 1.3.1.2 General config File Contents

```
{  
    "GatewaySourceHost": " https://CTXAppServer/gateway ",  
    "SQLServer": " CTXDBServer.MyDomain.com ",  
    "SQLUser": " MyDomain\\CTX_DB ",  
    "SQLPass": "#_1701471951570260202215102247064219146090002233079~  
213007178118043!126188102240252116240177225175155#",  
    "RetentionPeriod": 30,  
    "BackupConfigPath": "C:\\Backups\\Config"  
}
```

### 1.3.2 Flows config file

The location of this config file should be added into the parameter 'BackupConfigPath' in the general config file. If specific flows are required to be backed up (referenced by name as opposed to a group of them), then this file is required.

The flows config file should contain the 'Flows' parameter with a list of the flows to be backed up.

#### 1.3.2.1 Flows config parameters

Config Parameter	Details	Example
File	The base name of the studiopkg file produced	FlowsBackup
Path	The path to where the studiopkg file will be put. <b>This must exist prior to running the flows and may need to be created</b>	C:\\Backup\\BackupPackages  Note: json format requires escaping of backslashes, i.e. the double backslash represents a single backslash character in this format
GatewayUser	A valid username for a Cortex Gateway User. This could be the user of the module themselves, the user under which the gateway application pool is run, etc  <b>This must have studio authorisation rights to view and edit any flows to be backed up</b>	CTX_Gateway
GatewayPass	The Password for GatewayUser. This may be encrypted using the Cortex Encryptor if necessary, found in c:\\Cortex\\admin by default on a Cortex application server	Encrypted: #_213207061128133!076198044053108150073058139151206~060068237177!225210117133027107049186117027253#  Non-Encrypted: CTXPA55W0rd
Flows	Comma Separated Json list of the Flow(s) to be backed up	["Test-REST-Integration","Test-SOAP-Integration"]

### 1.3.2.2 Flows File Contents

```
{  
  "File": "FlowsBackup",  
  "Path": "C:\\\\Backups\\\\BackupFlows",  
  "GatewayUser": "example",  
  "GatewayPass": "#_9889898989832223190148110224148027167023169~24  
6118108026071!086164216218065000158171001088103#",  
  "Flows": ["Test-flow"]  
}
```

### 1.3.3 Groups config file

The location of this config file should be added into the parameter 'BackupConfigPath' in the general config file. If flow groups are required to be backed up (referenced by name), then this file is required.

The flows config file should contain the 'Groups' parameter with a list of the flow groups to be backed up. For any groups listed all flows within the group, or subgroups within the group, will be included in the package.

#### 1.3.3.1 Groups config parameters

Config Parameter	Details	Example
File	The base name of the studiopkg file produced.	GroupsBackup
Path	The path to where the studiopkg file will be put <b>This must exist prior to running the flows and may need to be created</b>	C:\\Backup\\BackupPackages  Note: json format requires escaping of backslashes, i.e. the double backslash represents a single backslash character in this format
GatewayUser	A valid username for a Cortex Gateway User. This could be the user of the module themselves, the user under which the gateway application pool is run, etc <b>This must have studio authorisation rights to view and edit any flows to be backed up</b>	CTX_Gateway
GatewayPass	The Password for GatewayUser. This may be encrypted using the Cortex Encryptor if necessary, found in c:\\Cortex\\admin by default on a Cortex application server	Encrypted: #_213207061128133!0761980 4405310815007305813915120 6~060068237177!2252101171 33027107049186117027253#  Non-Encrypted: CTXPA55W0rd
Groups	Comma Separated Name of Names of the Group(s) to backup	["Project 1","Generic Subtasks", "Reporting Flows"]

### 1.3.3.2 Groups File Contents

```
{  
  "File": "GroupsBackup",  
  "Path": "C:\\\\Backups\\\\BackupFlows",  
  "GatewayUser": "example",  
  "GatewayPass": "#_23700411060105!0322231901481102241480271670231  
69~246118108026071!0861642162180600000158171001088103#",  
  "Groups": ["CTX-Excel, CTX-Email"]  
}
```

### 1.3.4 System config file

The location of this config file should be added into the parameter 'BackupConfigPath' in the general config file. If all flows on an application server are to be backed up, then this config file is required

#### 1.3.4.1 System config parameters

Config Parameter	Details	Example
File	The base name of the studiopkg file produced.	SystemBackup
Path	The path to where the studiopkg file will be put <b>This must exist prior to running the flows and may need to be created</b>	C:\\Backup\\BackupPackages Note: json format requires escaping of backslashes.
GatewayUser	A valid username for a Cortex Gateway User. This could be the user of the module themselves, the user under which the gateway application pool is run, etc <b>This must have studio authorisation rights to view and edit any flows to be backed up</b>	CTX_Gateway
GatewayPass	The Password for GatewayUser. This may be encrypted using the Cortex Encryptor if necessary, found in c:\\Cortex\\admin by default on a Cortex application server	Encrypted: #_213207061128133!076198044053108150073058139151206~060068237177!225210117133027107049186117027253#  Non-Encrypted: CTXPA55W0rd

#### 1.3.4.2 System File Contents

```
{
  "File": "SystemBackup",
  "Path": "C:\\Backups\\BackupFlows",
  "GatewayUser": "example",
  "GatewayPass": "#_23700411060105!032223190148110224148027167023169~246118108026071!0861642162900000158171001088103#"
}
```



## 1.4 Using the Module

### 1.4.1 Global variables to set

There are two global variables which need to be set in the flow 'BF-Backup-Flows-From-Config':

- `g_backup-config-path`- Path to general config file. For example `<C:\Cortex\Admin>`
- `g_ps-credentials` - Structure containing Domain, Username, and password for PowerShell block. This should look as follows with the parameters filled in:

```
{  
    "Domain": "<domain of the Cortex Application Server>",  
    "Username": "<User that is authorised to execute PowerShell>",  
    "Password": "<Password for the above user>"  
}
```

- ❖ These are hard coded in the flow since it is the minimum information required for the flow to locate and import the information in the general config file, where other config items are stored.

### 1.4.2 Config Files

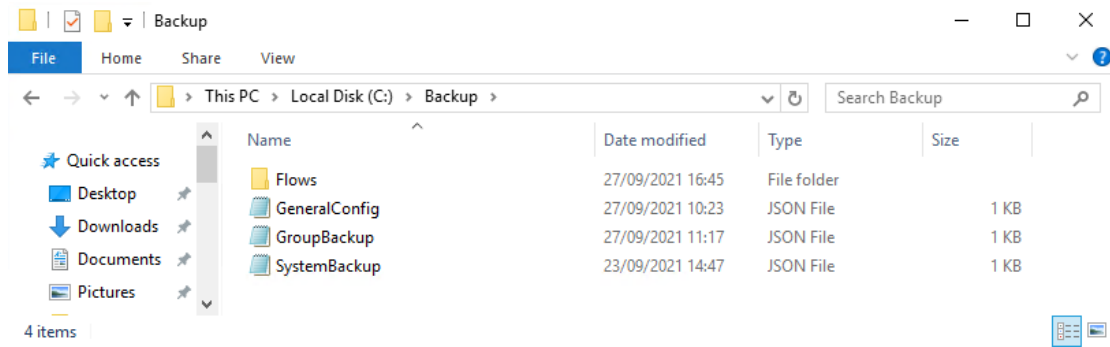
In order for the CTX-Backup-Flows flows to function at all, the general config file is required, as in section 1.3.1.

Then, in order for the CTX-Backup-Flows to back up the desired flows, at least one of the flow, group or system config file are also required. What exact combination are needed depends on what needs to be backed up

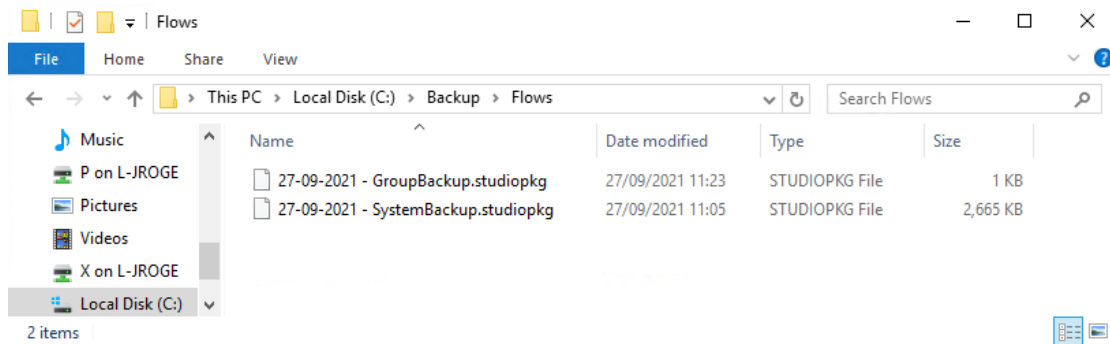
1. If a full system back up of all flows on the Cortex Application Server is needed, then a system config file should be created, as in section 1.3.4.
2. If a backup of a specific flow group or list of groups is needed, then a group config file should be created, as in section 1.3.3.
3. If a backup of a specific flow or list of flows is needed, then a flow config file should be created, as in section 1.3.2.

Any combination of flow, group or system config files may exist at the location specified in the `BackupConfigPath` parameter of the general config file. Each one will generate a separate backup `studiopkg` containing what is specified in the file.

For example, if a system config file and a group config file exist there, then every flow on the application server will be backed up into one `studiopkg`, and the groups specified will be backed up into another, as below



*An example of config files*



*An example of flows having been backed up based on the above config files*

If neither a system, backup nor flow config file is created then no backups will be created.

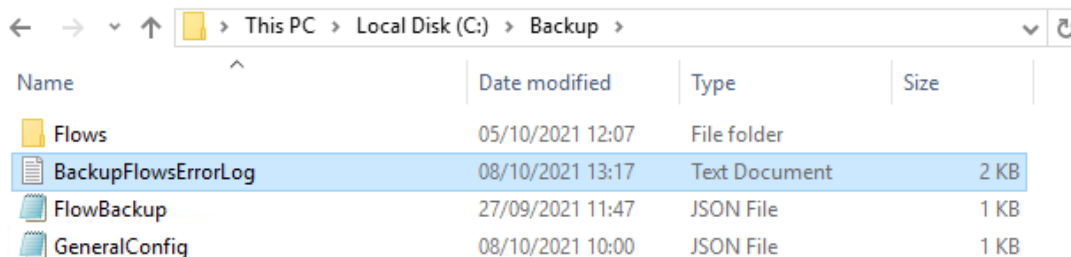
### 1.4.3 How to Create backups

After the Global variables have been set and the relevant config files created, run the 'BF-Backup-Flows-From-Config' flow. The backup studio packages resulting from the config files that exist on the server will appear in the location specified in variable 'Path' in the flow, group or system config files.

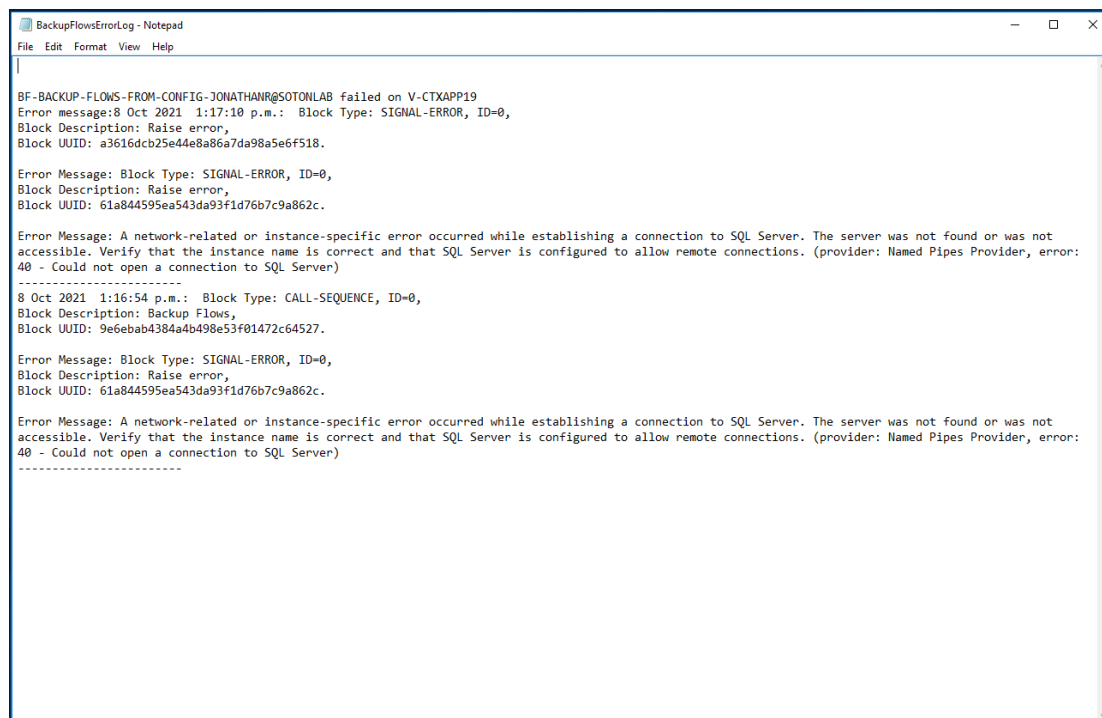
The flow 'BF-Backup-Flows-From-Config' may be scheduled to produce automated backups periodically, and if any changes to what is being backed up are required then the config files should be updated as needed.

### 1.4.4 Exception Handling

In the event of an exception, it will be logged to a text file named BackupFlowsErrorLog, which will be placed in the location specified for the general config file, as in section 1.3.1. See below for an example:



Name	Date modified	Type	Size
Flows	05/10/2021 12:07	File folder	
BackupFlowsErrorLog	08/10/2021 13:17	Text Document	2 KB
FlowBackup	27/09/2021 11:47	JSON File	1 KB
GeneralConfig	08/10/2021 10:00	JSON File	1 KB



```

BF-BACKUP-FLOWS-FROM-CONFIG-JONATHANR@SOTONLAB failed on V-CTXAPP19
Error message: 8 Oct 2021 1:17:10 p.m.: Block Type: SIGNAL-ERROR, ID=0,
Block Description: Raise error,
Block UUID: a3616dcb25e44e8a86a7da98a5e6f518.

Error Message: Block Type: SIGNAL-ERROR, ID=0,
Block Description: Raise error,
Block UUID: 61a844595ea543da93f1d76b7c9a862c.

Error Message: A network-related or instance-specific error occurred while establishing a connection to SQL Server. The server was not found or was not
accessible. Verify that the instance name is correct and that SQL Server is configured to allow remote connections. (provider: Named Pipes Provider, error:
40 - Could not open a connection to SQL Server)
-----
8 Oct 2021 1:16:54 p.m.: Block Type: CALL-SEQUENCE, ID=0,
Block Description: Backup Flows,
Block UUID: 9e6ebab4384a4b498e53f01472c64527.

Error Message: Block Type: SIGNAL-ERROR, ID=0,
Block Description: Raise error,
Block UUID: 61a844595ea543da93f1d76b7c9a862c.

Error Message: A network-related or instance-specific error occurred while establishing a connection to SQL Server. The server was not found or was not
accessible. Verify that the instance name is correct and that SQL Server is configured to allow remote connections. (provider: Named Pipes Provider, error:
40 - Could not open a connection to SQL Server)
-----

```

## 2 Backup Module Flows

---


### 2.1 BF-Backup-Flows-From-Config

#### 2.1.1 Overview

This flow handles the backup of Cortex flows based on the JSON file configurations.

The flow will get the required flow(s) to be backed up based on the config files that exist.

This will then validate each against the CortexWeb DB and pass the details into the flow 'BF-Backup-Flows'.

 Note that the variable 'g\_backup-config-path' needs to be initialised with the folder path containing all the backup config files. Also the global structure variable 'g\_ps-credentials' needs to be initialised with Domain, Username and Password elements for PowerShell block credentials, as in Section 1.4.1.

### 2.2 BF-Backup-Flows

#### 2.2.1 Overview

This flow will back up all the inputted flows and is called from BF-Backup-Flows-From-Config.

The following variables are passed into the flow from BF-Backup-Flows-From-Config:

- g\_export-location
- g\_export-filename
- g\_list-of-flows
- g\_sql-server
- g\_gateway-username
- g\_gateway-encrypted-password
- g\_SourceHost

This is handled from the BF-Backup-Flows-From-Config flows and should not be changed.

## 3 Backup Module Subtasks

---

### 3.1 BF-GCAT-Get-Cortex-Authentication-Token

#### 3.1.1 Overview

This subtask logs in to Cortex Gateway using a REST request and returns a Bearer Token which can be used to authenticate subsequent REST requests such as Exporting / Publishing flows.

This subtask is used by the Backup-Flows and Backup-Flows-From-Config flows and requires no configuration or setup.

#### 3.1.2 Input variables

Name	Type	Comments
GCAT_i_HostURL	Text	the Cortex Gateway URL
GCAT_i_GatewayUser	Text	A valid User to login with
GCAT_i_GatewayPassword	Text	The encrypted password for the user
GCAT_i_PsCredentials	Structure	PowerShell domain, username, and password  <pre>{   "Domain": "&lt;domain of the Cortex Application Server&gt;",   "Username": "&lt;User that is authorised to execute PowerShell&gt;",   "Password": "&lt;Password for the above user&gt;" }</pre>

#### 3.1.3 Output variables

Name	Type	Comments
GCAT_o_Bearer-Auth-Token	Text	The Bearer Token from the Login

## 3.2 BF-PPR-Process-PowerShell-Response

### 3.2.1 Overview

This subtask processes the PowerShell response. If there were any error records it appends all these and raises an error. If not, it can either return the output as a list or as a text depending on the user's choice

### 3.2.2 Input variables

Name	Type	Comments
PPR_i_ps-response	Structure	The response from the PowerShell block
PPR_i_convert-to-text-bool	Boolean	Truth value where 'true' indicates to return it as text. If false, a list is returned.

### 3.2.3 Output variables

Name	Type	Comments
PPR_o_ps-list	List	Variable containing the output of the command
PPR_o_ps-text	Text	Variable containing the output of the command

### 3.3 BF-RMFFB-Remove-Missing-Flow-from-Backup

#### 3.3.1 Overview

This subtask removes an erroring flow from the backups. This is normally caused by a missing or deleted flow which still has an entry in the DB.

The erroring flow name is used to find the UUID of the flow, which is then removed from the list of flows to backup so the process can be retried.

This subtask is used by the Backup-Flows and Backup-Flows-From-Config flows and requires no configuration or setup.

#### 3.3.2 Input variables

Name	Type	Comments
RMFFB_i_missing-flow	Text	The name of the flow to remove
RMFFB_i_flows-json	Text	The JSON list of flows to backup
RMFFB_i_backup-db-server	Text	The DB Server containing CortexWeb

#### 3.3.3 Output variables

Name	Type	Comments
RMFFB_o_backup-flows-json	Text	The JSON list of flows without the flow to remove