App control

Astra Automation

NetApp
April 29, 2022

This PDF was generated from https://docs.netapp.com/us-en/astra-automation/workflows/wf_list_unman_apps.html on April 29, 2022. Always check docs.netapp.com for the latest.

Table of Contents

App control . . . 1
List the unmanaged apps . . . 1
List the managed apps . . . 5
Get a managed app . . . 6
Manage an app . . . 7
Unmanage an app . . . 7

App control

List the unmanaged apps

You can list the applications that are currently not managed by Astra. You might do this as part of selecting an app to be managed.

> The REST endpoint used in these workflows returns all the Astra applications by default. You can use the `filter` query parameter on the API call to request only the unmanaged apps be returned. As an alternative, you can omit the filter parameter to return all the apps and then examine the `managedState` field in the output to determine which apps are in the `unmanaged` state.

List only the apps with managedState equal to unmanaged

This workflow uses the `filter` query parameter to return only the unmanaged apps.

1. List the unmanaged applications

Perform the following REST API call.

HTTP method	Path
GET	/account/{accountID}/topology/v1/apps

Additional input parameters

In addition to the parameters common with all REST API calls, the following parameters are also used in the curl examples for this step.

Parameter	Type	Required	Description
filter	Query	No	Use a filter to specify which apps should be returned.
include	Query	No	Optionally select the values you want returned in the response.

Curl example: Return the name, id, and managedState for the unmanaged apps

```
curl --location -i --request GET
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/topology/v1/apps?filter=man
agedState%20eq%20'unmanaged'&include=name,id,managedState' --header
'Accept: */*' --header 'Authorization: Bearer <API_TOKEN>'
```

JSON output example

```
{
  "items": [
    [
      "maria",
      "eed19f78-0884-4792-bb7a-313258c6b0b1",
      "unmanaged"
    ],
    [
      "test-postgres-app",
      "1ee6235b-cda1-45cb-8d4c-630bdb8b41a5",
      "unmanaged"
    ],
    [
      "postgres1-postgresql",
      "e591ee59-ea90-4a9f-8e6c-d2b6e8647096",
      "unmanaged"
    ],
    [
      "kube-system",
      "077a2f73-4b51-4d04-8c6c-f63b3b069755",
      "unmanaged"
    ],
    [
      "trident",
      "5b6fc28f-e308-4653-b9d2-6d66a764d2e1",
      "unmanaged"
    ],
    [
      "postgres1-postgresql-clone",
      "06be05c5-763e-4d73-bd06-1f27f5f2e130",
      "unmanaged"
    ]
  ],
  "metadata": {}
}
```


List all the apps and select the unmanaged apps

This workflow returns all the apps. You must examine the output to determine which are unmanaged.

1. List all the applications

Perform the following REST API call.

HTTP method	Path
GET	/account/{accountID}/topology/v1/apps

Additional input parameters

In addition to the parameters common with all REST API calls, the following parameters are also used in the curl examples for this step.

Parameter	Type	Required	Description
include	Query	No	Optionally select the values you want returned in the response.

Curl example: Return all data for all apps

```
curl --location -i --request GET
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/topology/v1/apps' --header
'Accept: */*' --header 'Authorization: Bearer <API_TOKEN>'
```

Curl example: Return the name, id, and managedState for all apps

```
curl --location -i --request GET
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/topology/v1/apps?include=na
me,id,managedState' --header 'Accept: */*' --header 'Authorization: Bearer
<API_TOKEN>'
```

JSON output example

```
{
  "items": [
    [
      "maria",
      "eed19f78-0884-4792-bb7a-313258c6b0b1",
      "unmanaged"
    ],
    [
      "mariadb-mariadb",
      "8da20fff-c69c-4170-bb0d-e4f91c5a1333",
      "managed"
    ],
    [
      "test-postgres-app",
      "1ee6235b-cda1-45cb-8d4c-630bdb8b41a5",
      "unmanaged"
    ],
    [
      "postgres1-postgresql",
      "e591ee59-ea90-4a9f-8e6c-d2b6e8647096",
      "unmanaged"
    ],
    [
      "kube-system",
      "077a2f73-4b51-4d04-8c6c-f63b3b069755",
      "unmanaged"
    ],
    [
      "trident",
      "5b6fc28f-e308-4653-b9d2-6d66a764d2e1",
      "unmanaged"
    ],
    [
      "postgres1-postgresql-clone",
      "06be05c5-763e-4d73-bd06-1f27f5f2e130",
      "unmanaged"
    ],
    [
      "davidns-postgres-app",
      "11e046b7-ec64-4184-85b3-debcc3b1da4d",
      "managed"
    ]
  ],
  "metadata": {}
}
```

2. Select the unmanaged applications

Review the output of the API call and manually select the apps with `managedState` equal to `unmanaged`.

List the managed apps

You can list the applications that are currently managed by Astra. You might do this as part of finding the snapshots or backups for a specific app.

1. List the applications

Perform the following REST API call.

HTTP method	Path
GET	/account/{accountID}/k8s/v1/managedApps

Additional input parameters

In addition to the parameters common with all REST API calls, the following parameters are also used in the curl examples for this step.

Parameter	Type	Required	Description
include	Query	No	Optionally select the values you want returned in the response.

Curl example: Return all data for all apps

```
curl --location -i --request GET
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/k8s/v1/managedApps'
--header 'Accept: */*' --header 'Authorization: Bearer <API_TOKEN>'
```

Curl example: Return the name, id, and state for all apps

```
curl --location -i --request GET
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/k8s/v1/managedApps?include=
name,id,state' --header 'Accept: */*' --header 'Authorization: Bearer
<API_TOKEN>'
```

JSON output example

```
{
  "items": [
    [
      "test-postgres-app",
      "1ee6235b-cda1-45cb-8d4c-630bdb8b41a5",
      "running"
    ]
  ],
  "metadata": {}
}
```


Get a managed app

You can retrieve all the resource variables describing a single managed application.

Before you begin

You must have the ID of the managed app you want to retrieve. If needed you can use the workflow List the managed apps to locate the application.

1. Get the application

Perform the following REST API call.

HTTP method	Path
GET	/accounts/{account_id}/k8s/v1/managedApps/{managedApp_id}

Additional input parameters

In addition to the parameters common with all REST API calls, the following parameters are also used in the curl examples for this step.

Parameter	Type	Required	Description
managed app id	Path	Yes	ID value of the managed application to retrieve.

Curl example: Return all data for the application

```
curl --location -i --request GET
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/k8s/v1/managedApps/<MANAGED
_APP_ID>' --header 'Accept: */*' --header 'Authorization: Bearer
<API_TOKEN>'
```

Manage an app

You can create a managed application based on an application already known to Astra. When an application is managed, you can protect it by taking regular backups and snapshots.

Before you begin

You must have the ID of the discovered app you want to manage. If needed you can use the workflow List the unmanaged apps to locate the application.

1. Manage the application

Perform the following REST API call.

HTTP method	Path
POST	/account/{accountID}/k8s/v1/managedApps

Additional input parameters

In addition to the parameters common with all REST API calls, the following parameters are also used in the curl examples for this step.

Parameter	Type	Required	Description
JSON	Body	Yes	Provides the parameters needed to identify the application to be managed. See the example below.

JSON input example

```
{
  "type": "application/astra-managedApp",
  "version": "1.1",
  "id": "7da20fff-c69d-4270-bb0d-a4f91c5a1333"
}
```

Curl example: Manage an app

```
curl --location -i --request POST
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/k8s/v1/managedApps'
--header 'Content-Type: application/astra-managedApp+json' --header
'Accept: */*' --header 'Authorization: Bearer <API_TOKEN>'  --d @JSONinput
```

Unmanage an app

You can remove a managed app when it’s no longer needed. Removing a managed

application also deletes the associated schedules.

Before you begin

You must have the ID of the managed app you want to unmanage. If needed you can use the workflow List the managed apps to locate the application.

The application's backups and snapshots are not automatically removed when it is deleted. If you no longer need the backups and snapshots, you should delete them before removing the application.

1. Unmanaged the app

Perform the following REST API call.

HTTP method	Path
DELETE	/accounts/{account_id}/k8s/v1/managedApps/{managedApp_id}

Additional input parameters

In addition to the parameters common with all REST API calls, the following parameters are also used in the curl examples for this step.

Parameter	Type	Required	Description
managed app id	Path	Yes	Identifies the managed application to remove.

Curl example: Remove a managed app

```
curl --location -i --request DELETE
'https://astra.netapp.io/accounts/<ACCOUNT_ID>/k8s/v1/managedApps/<MANAGED
_APP_ID>' --header 'Accept: */*' --header 'Authorization: Bearer
<API_TOKEN>'
```

Copyright Information

Copyright © 2022 NetApp, Inc. All rights reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means-graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system-without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark Information

NETAPP, the NETAPP logo, and the marks listed at http://www.netapp.com/TM are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.