TOTD: New Configuration Archive feature - Export Device configurations to an encrypted zip file

Prepared for: Cisco DNA Customer, Solutions Support
Prepared by: Tomas de Leon, Technical Leader
September 21, 2020
Document number: 09212020_v1

CISCO SYSTEMS - CX CENTERS ENTERPRISE NETWORKING

TECHNOTE OF THE DAY (TOTD) -- NEW CONFIGURATION ARCHIVE FEATURE - EXPORT DEVICE CONFIGURATIONS TO AN ENCRYPTED ZIP FILE IN CISCO DNA CENTER VERSION 2.1.2.X

Objective

The objective of this document is to show users the new Configuration Archive feature. A new Rest API was added to export device configurations to an encrypted zip file. The new API was added in Cisco DNA Center version 2.1.2.0.

Goals

Provide an awareness of the new configuration archive feature and to provide an example of using the API with the Postman Rest API Client.

The following technote is written against the Release of Cisco DNA Center version 2.1.2.0.

Reference Information:

- Cisco DNA Center version 2.1.2.x0
- Postman Version 7.32.0 (7.32.0)

CISCO SYSTEMS - CX CENTERS ENTERPRISE NETWORKING

CONFIGURATION ARCHIVE FEATURE

In the Release of Cisco DNA Center version 2.1.2.0, there is only one API URL task provided for the Configuration Archive feature. The current API task is to export device configurations to an encrypted zip file. My understanding is that more API URL task will be coming in future releases but this is always subject to change.

Platform . Developer Toolkit . APIs

Configuration Archive

Name	Description
Export Device configurations	*Export Device configurations to an encrypted zip file.*

Method	URL
POST	https://<_dnac.ip.address_>/dna/intent/api/v1/network-device-archive/cleartext

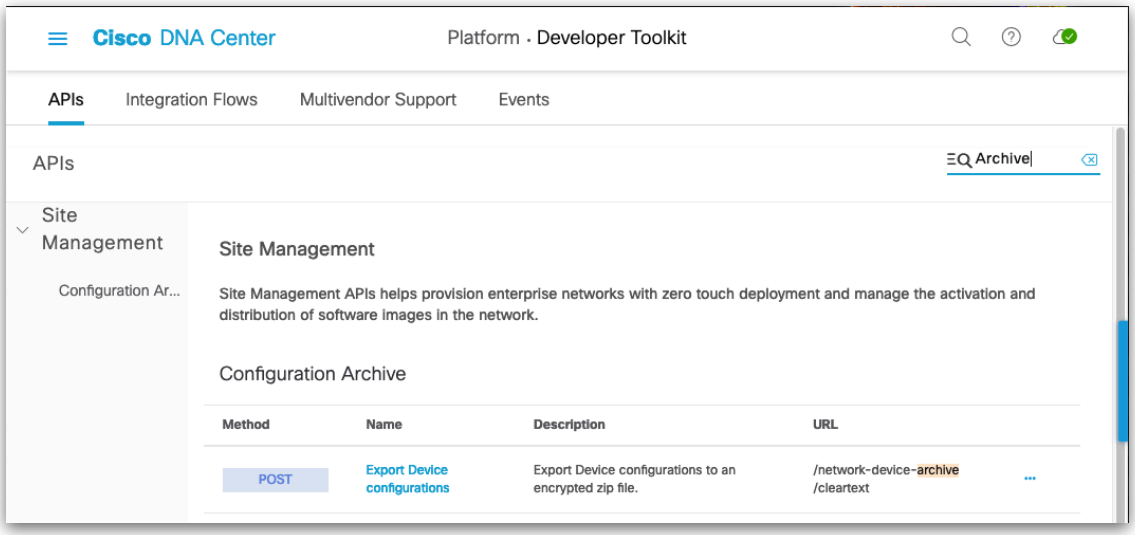
Model Schema
payload:

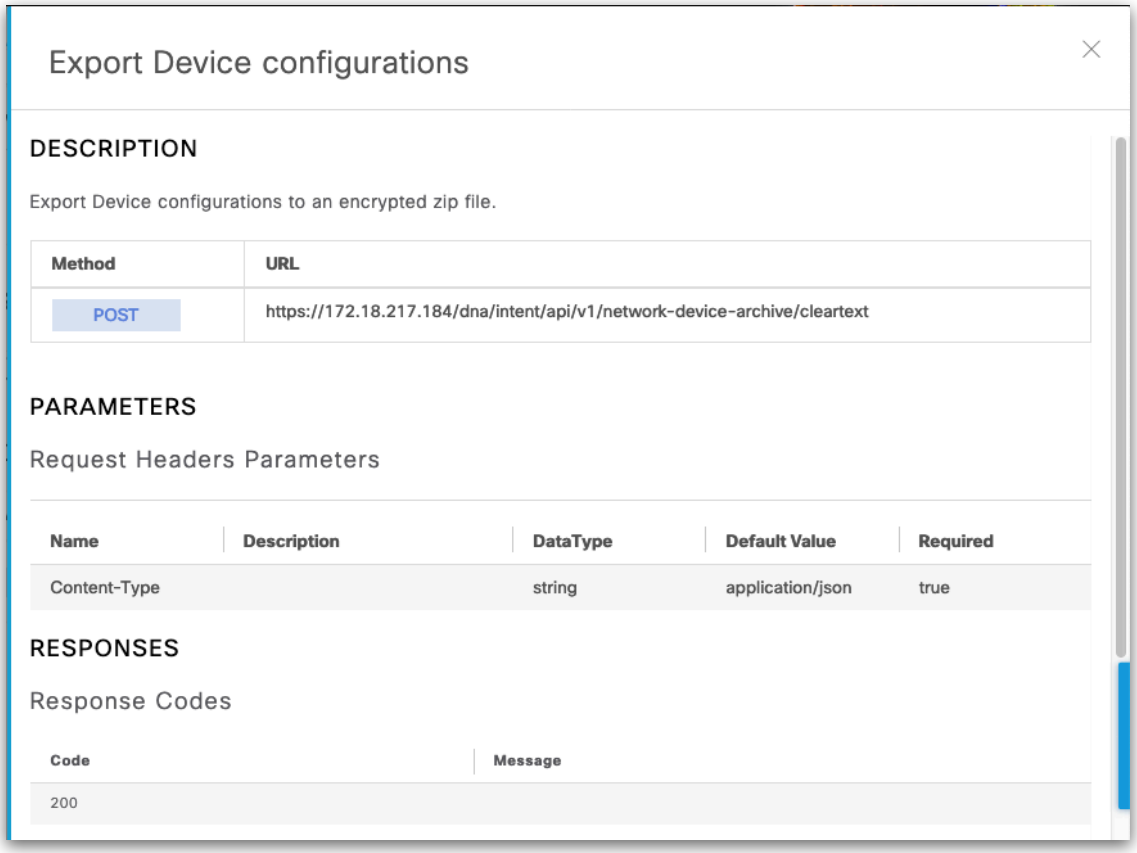
```
{

"deviceId": "<List of device uuids to be exported>",

"password": "<Password to export the file. Same should be used to view the file>"

}
```

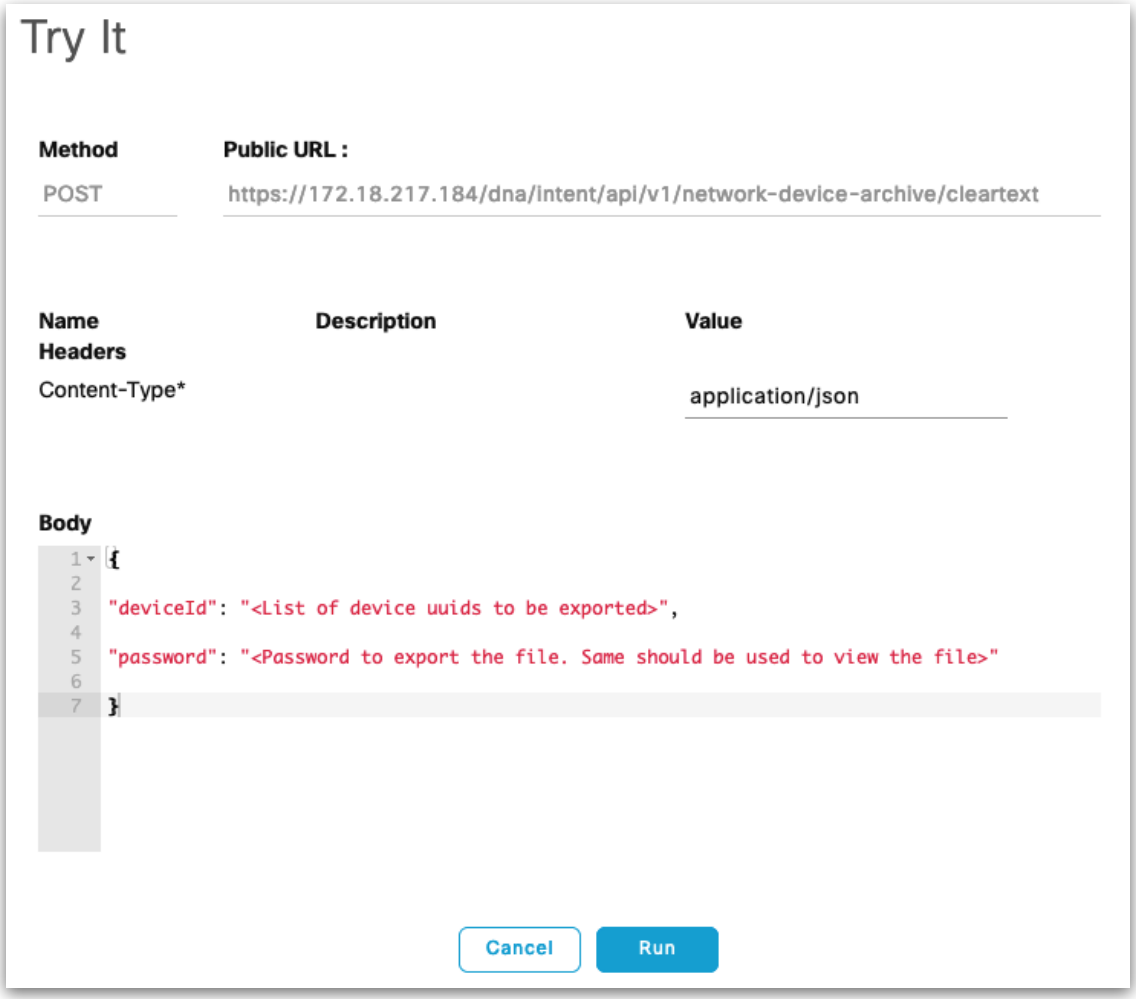
Note: The schema payload is not listed in the Cisco DNA Center screens or the help. A CDET has been created to add this documentation to the UI in an upcoming release.

Platform . Developer Toolkit . APIs (cont.)

Cisco DNA Center UI Example:







Note: The schema payload is not listed in the Cisco DNA Center screens or the help. A CDET has been created to add this documentation to the UI in an upcoming release.

USING THE POSTMAN REST API CLIENT FOR EXPORTING DEVICE CONFIGURATIONS

Use Case Scenario - Export Device configurations to an encrypted zip file using the Postman Rest API Client.

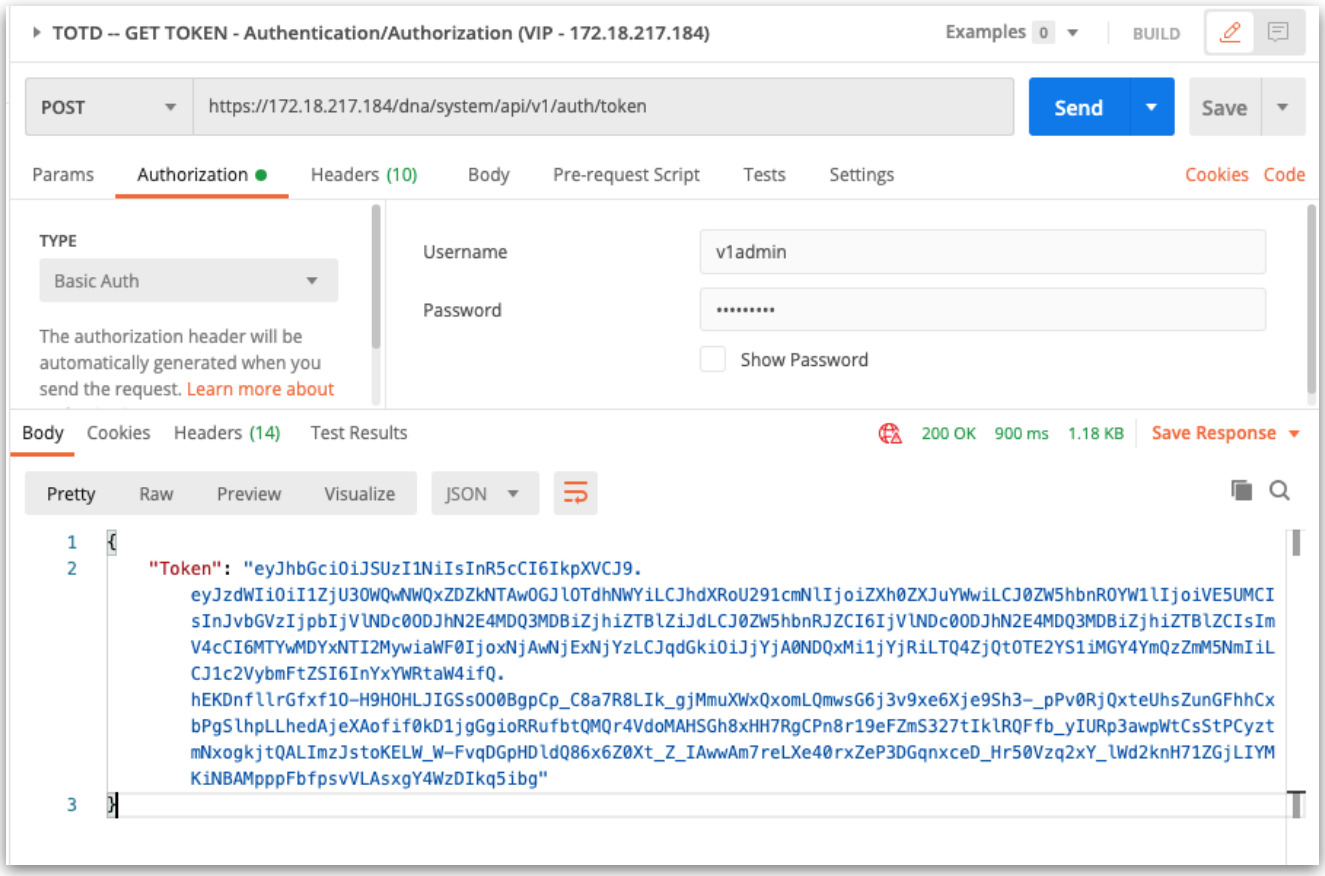
- Cisco DNA Center version 2.1.2.0
- Postman Version 7.32.0 (7.32.0)

STEP 1 - GET TOKEN - Authentication/Authorization

Method	URL
POST	https://<_dnac.ip.address_>/dna/system/api/v1/auth/token

Response:

eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiI1ZjU3OWQwNWQxZDZkNTAwOGJlOTdhNWYiLCJhdXRoU291cmNlIjoiZXh0ZXJuYWwiLCJ0ZW5hbnROYW1lIjoiVE5UMCIsInJvbGVzIjpbIjVlNDc0ODJhN2E4MDQ3MDBiZjhiZTBlZiJdLCJ0ZW5hbnRJZCI6IjVlNDc0ODJhN2E4MDQ3MDBiZjhiZTBlZCIsImV4cCI6MTYwMDYxNTI2MywiaWF0IjoxNjAwNjExNjYzLCJqdGkiOiJjYjA0NDQxMi1jYjRiLTQ4ZjQtOTE2YS1iMGY4YmQzZmM5NmIiLCJ1c2VybmFtZSI6InYxYWRtaW4ifQ.hEKDnfllrGfxf1O-H9HOHLJIGSsOO0BgpCp_C8a7R8LIk_gjMmuXWxQxomLQmwsG6j3v9xe6Xje9Sh3-_pPv0RjQxteUhsZunGFhhCxbPgSlhpLLhedAjeXAofif0kD1jgGgioRRufbtQMQr4VdoMAHSGh8xHH7RgCPn8r19eFZmS327tIklRQFfb_yIURp3awpWtCsStPCyztmNxogkjtQALImzJstoKELW_W-FvqDGpHDldQ86x6Z0Xt_Z_IAwwAm7reLXe40rxZeP3DGqnxceD_Hr50Vzq2xY_lWd2knH71ZGjLIYMKiNBAMpppFbfpsvVLAsxgY4WzDIkq5ibg



STEP 2 - GET DEVICE INFO using managementIpAddress

Method

GET

URL

https://<_dnac.ip.address_>/dna/intent/api/v1/network-device?managementIpAddress=<_device.mgmt.ip.address_>

- Test Case Parameters

NETWORK DEVICE INFORMATION

NAME	MGMT_IP	PLATFORM	SERIAL_NUM
deadbeef-1.dna.local	192.168.0.101	C9200L-48T-4G	JAE22490HPJ
deadbeef-2.dna.local	192.168.0.102	C9200L-48T-4G	JAE22490HGT

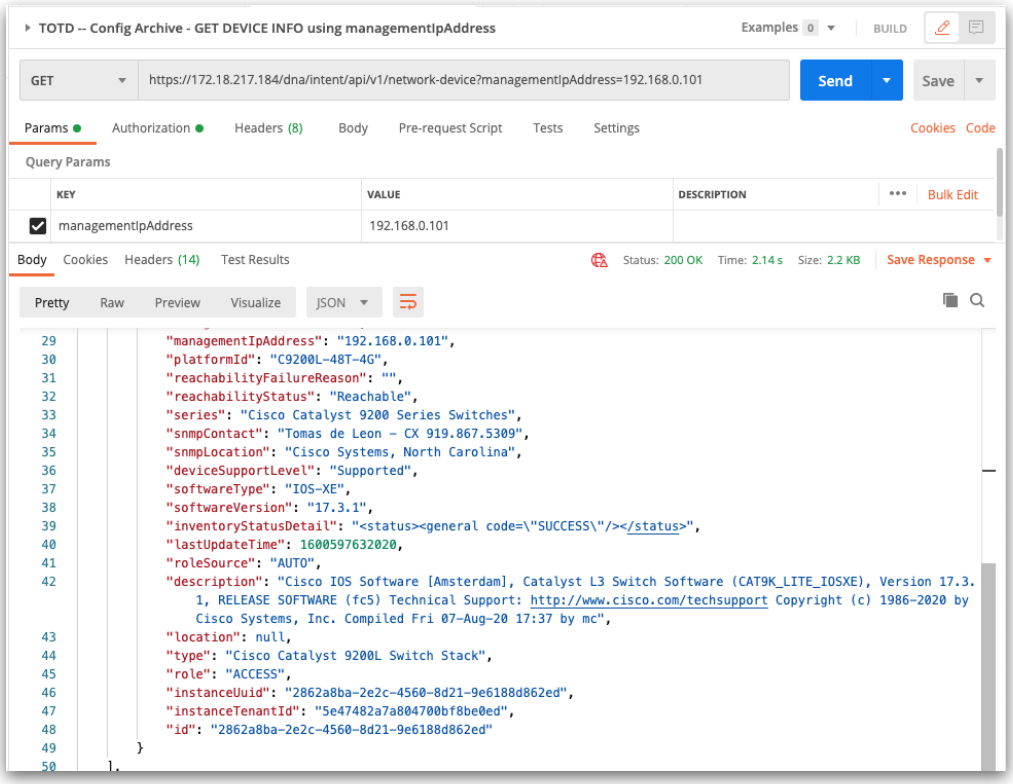
Response: (abbreviated)

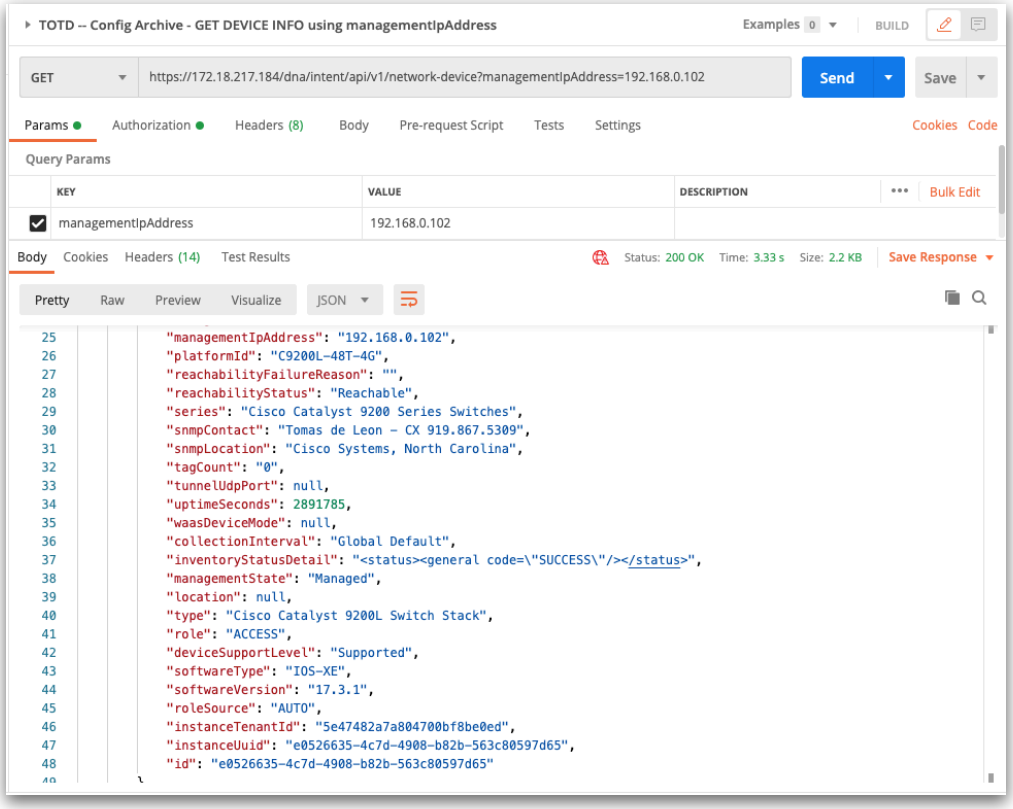

```
{
   "response": [
      {
         "family": "Switches and Hubs",
         "collectionStatus": "Managed",
         "serialNumber": "JAE22490HPJ",
          "macAddress": "08:ec:f5:88:0d:80",
          "hostname": "deadbeef-1.dna.local",
          "managementState": "Managed",
         "managementIpAddress": "192.168.0.101",
          "platformId": "C9200L-48T-4G",
         "reachabilityStatus": "Reachable",
          "series": "Cisco Catalyst 9200 Series Switches",
          "snmpContact": "Tomas de Leon - CX 919.867.5309",
          "snmpLocation": "Cisco Systems, North Carolina",
          "deviceSupportLevel": "Supported",
          "softwareType": "IOS-XE",
          "softwareVersion": "17.3.1",
          "inventoryStatusDetail": "<status><general code=\"SUCCESS\"/></status>",
         "type": "Cisco Catalyst 9200L Switch Stack",
          "role": "ACCESS",
```

```
            "instanceUuid": "2862a8ba-2e2c-4560-8d21-9e6188d862ed",
            "instanceTenantId": "5e47482a7a804700bf8be0ed",
            "id": "2862a8ba-2e2c-4560-8d21-9e6188d862ed"
        }
    ],
    "version": "1.0"
}
```


Response: (abbreviated)


```
{
    "response": [
        {
            "family": "Switches and Hubs",
            "hostname": "deadbeef-2.dna.local",
            "macAddress": "08:ec:f5:93:fb:00",
            "serialNumber": "JAE22490HGT",
            "collectionStatus": "Managed",
            "managementIpAddress": "192.168.0.102",
            "platformId": "C9200L-48T-4G",
            "reachabilityStatus": "Reachable",
            "series": "Cisco Catalyst 9200 Series Switches",
            "snmpContact": "Tomas de Leon - CX 919.867.5309",
            "snmpLocation": "Cisco Systems, North Carolina",
            "inventoryStatusDetail": "<status><general code=\"SUCCESS\"/></status>",
            "managementState": "Managed",
            "type": "Cisco Catalyst 9200L Switch Stack",
            "role": "ACCESS",
            "deviceSupportLevel": "Supported",
            "softwareType": "IOS-XE",
            "softwareVersion": "17.3.1",
            "instanceTenantId": "5e47482a7a804700bf8be0ed",
            "instanceUuid": "e0526635-4c7d-4908-b82b-563c80597d65",
            "id": "e0526635-4c7d-4908-b82b-563c80597d65"
        }
    ],
    "version": "1.0"
}
```


TOTD -- Config Archive - GET DEVICE INFO using managementIpAddress
Examples 0
BUILD
GET
https://172.18.217.184/dna/intent/api/v1/network-device?managementIpAddress=192.168.0.101
Send
Save
Params
Authorization
Headers (8)
Body
Pre-request Script
Tests
Settings
Cookies
Code
Query Params
KEY
VALUE
DESCRIPTION
Bulk Edit
managementIpAddress
192.168.0.101
Body
Cookies
Headers (14)
Test Results
Status: 200 OK
Time: 2.14 s
Size: 2.2 KB
Save Response
Pretty
Raw
Preview
Visualize
JSON
"managementIpAddress": "192.168.0.101",
"platformId": "C9200L-48T-4G",
"reachabilityFailureReason": "",
"reachabilityStatus": "Reachable",
"series": "Cisco Catalyst 9200 Series Switches",
"snmpContact": "Tomas de Leon - CX 919.867.5309",
"snmpLocation": "Cisco Systems, North Carolina",
"deviceSupportLevel": "Supported",
"softwareType": "IOS-XE",
"softwareVersion": "17.3.1",
"inventoryStatusDetail": "<status><general code=\"SUCCESS\"/></status>",
"lastUpdateTime": 1600597632020,
"roleSource": "AUTO",
"description": "Cisco IOS Software [Amsterdam], Catalyst L3 Switch Software (CAT9K_LITE_IOSXE), Version 17.3.1, RELEASE SOFTWARE (fc5) Technical Support: http://www.cisco.com/techsupport Copyright (c) 1986-2020 by Cisco Systems, Inc. Compiled Fri 07-Aug-20 17:37 by mc",
"location": null,
"type": "Cisco Catalyst 9200L Switch Stack",
"role": "ACCESS",
"instanceUuid": "2862a8ba-2e2c-4560-8d21-9e6188d862ed",
"instanceTenantId": "5e47482a7a804700bf8be0ed",
"id": "2862a8ba-2e2c-4560-8d21-9e6188d862ed"


TOTD -- Config Archive - GET DEVICE INFO using managementIpAddress
Examples 0
BUILD
GET
https://172.18.217.184/dna/intent/api/v1/network-device?managementIpAddress=192.168.0.102
Send
Save
Params
Authorization
Headers (8)
Body
Pre-request Script
Tests
Settings
Cookies
Code
Query Params
KEY
VALUE
DESCRIPTION
Bulk Edit
managementIpAddress
192.168.0.102
Body
Cookies
Headers (14)
Test Results
Status: 200 OK
Time: 3.33 s
Size: 2.2 KB
Save Response
Pretty
Raw
Preview
Visualize
JSON
"managementIpAddress": "192.168.0.102",
"platformId": "C9200L-48T-4G",
"reachabilityFailureReason": "",
"reachabilityStatus": "Reachable",
"series": "Cisco Catalyst 9200 Series Switches",
"snmpContact": "Tomas de Leon - CX 919.867.5309",
"snmpLocation": "Cisco Systems, North Carolina",
"tagCount": "0",
"tunnelUdpPort": null,
"uptimeSeconds": 2891785,
"waasDeviceMode": null,
"collectionInterval": "Global Default",
"inventoryStatusDetail": "<status><general code=\"SUCCESS\"/></status>",
"managementState": "Managed",
"location": null,
"type": "Cisco Catalyst 9200L Switch Stack",
"role": "ACCESS",
"deviceSupportLevel": "Supported",
"softwareType": "IOS-XE",
"softwareVersion": "17.3.1",
"roleSource": "AUTO",
"instanceTenantId": "5e47482a7a804700bf8be0ed",
"instanceUuid": "e0526635-4c7d-4908-b82b-563c80597d65",
"id": "e0526635-4c7d-4908-b82b-563c80597d65"

STEP 3 - EXPORT CONFIGURATION using DEVICE_ID

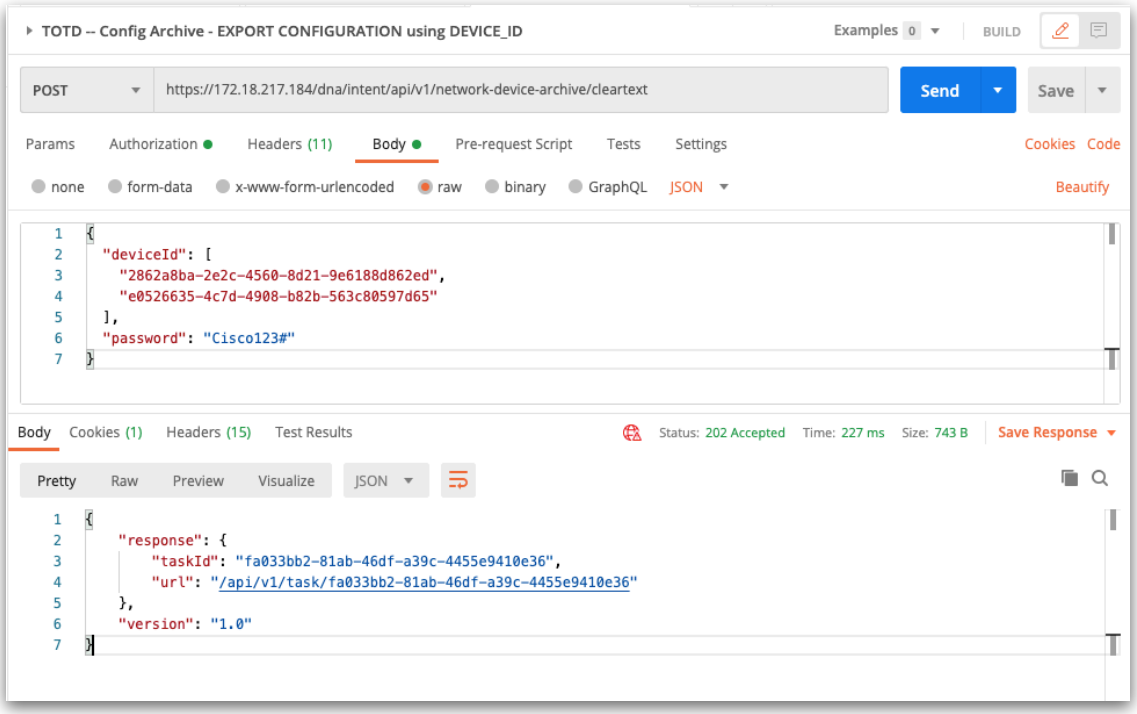
Method URL
POST https://<_dnac.ip.address_>/dna/intent/api/v1/network-device-archive/cleartext

Body

```
{
  "deviceId": [
    "2862a8ba-2e2c-4560-8d21-9e6188d862ed",
    "e0526635-4c7d-4908-b82b-563c80597d65"
  ],
  "password": "Cisco123#"
}
```

Response:

```
{
    "response": {
        "taskId": "fa033bb2-81ab-46df-a39c-4455e9410e36",
        "url": "/api/v1/task/fa033bb2-81ab-46df-a39c-4455e9410e36"
}
```



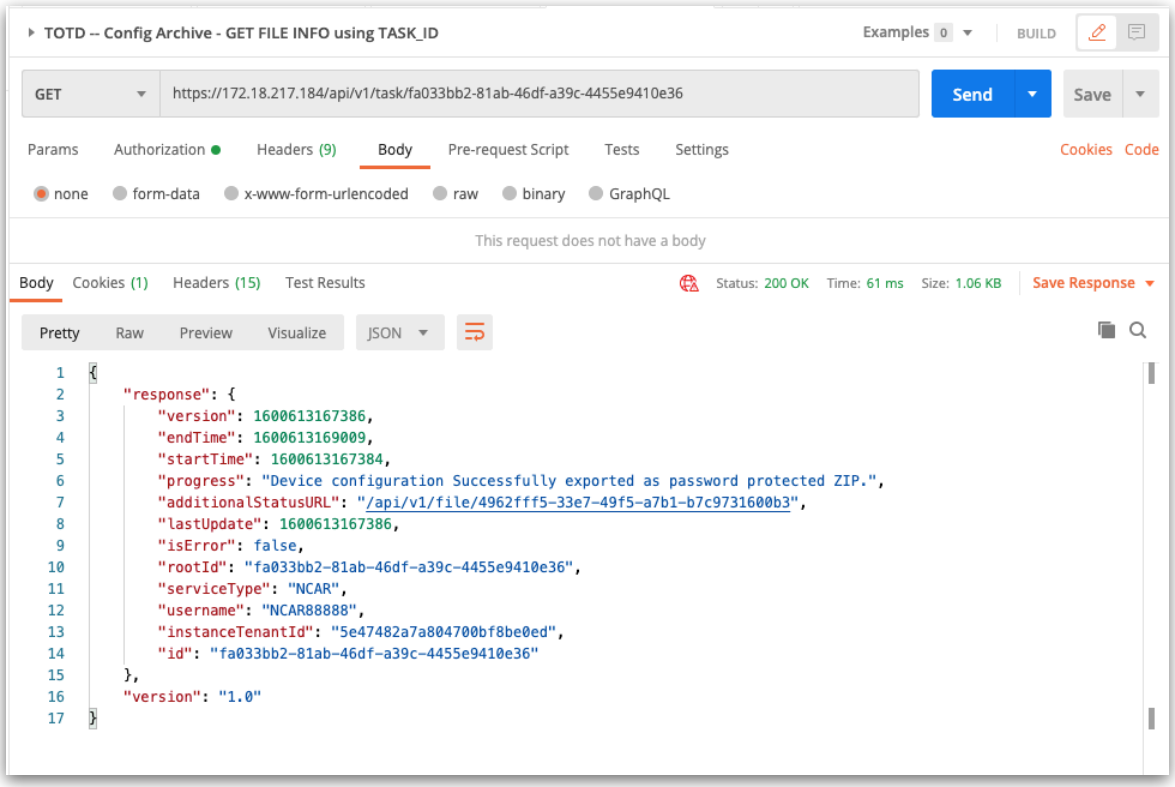
STEP 4 - GET FILE INFO using TASK_ID

Method	URL
GET	https://<_dnac.ip.address_>/api/v1/task/<_task.id_>

From Step 3:
"taskId": "**fa033bb2-81ab-46df-a39c-4455e9410e36**"

Response: (abbreviated)


```
{
   "response": {
      "version": 1600613167386,
      "endTime": 1600613169009,
      "startTime": 1600613167384,
      "progress": "Device configuration Successfully exported as password protected ZIP.",
      "additionalStatusURL": "/api/v1/file/4962fff5-33e7-49f5-a7b1-b7c9731600b3",
      "lastUpdate": 1600613167386,
}
```

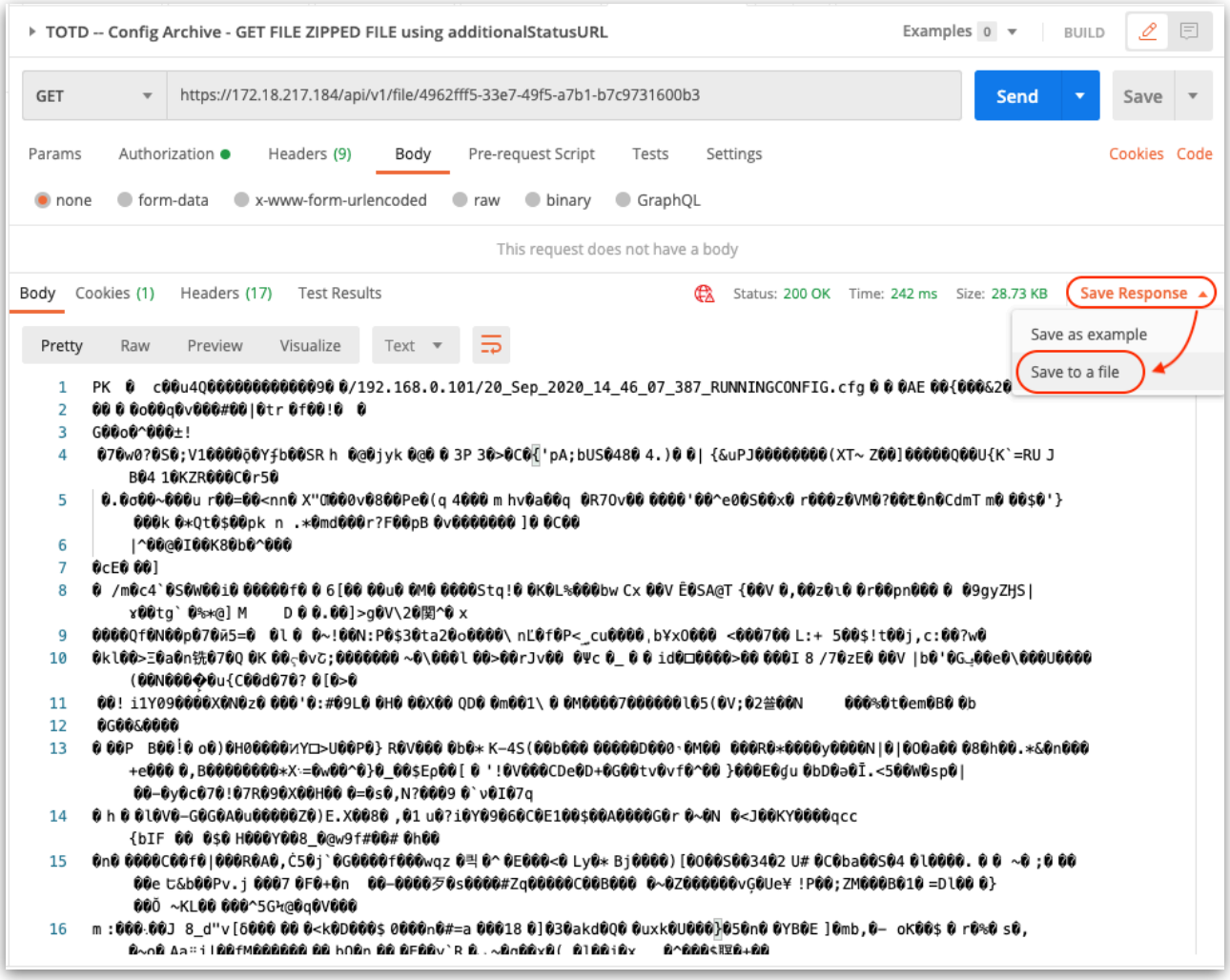
STEP 5 - GET FILE ZIPPED FILE using additionalStatusURL

Method	URL
GET	https://<_dnac.ip.address_>/<_additionalStatusURL.string_>

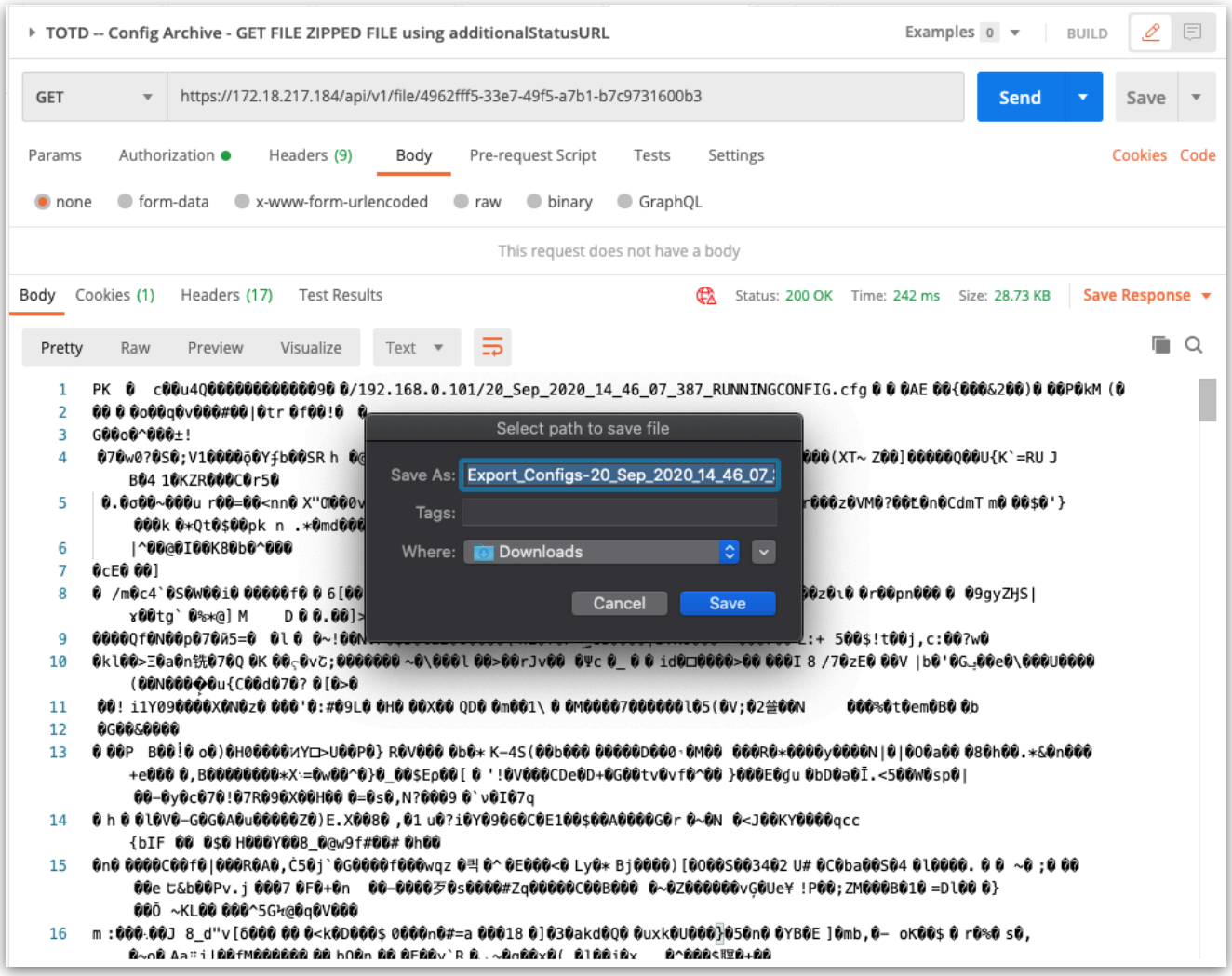
From Step 4:
"additionalStatusURL": "**/api/v1/file/4962fff5-33e7-49f5-a7b1-b7c9731600b3**",

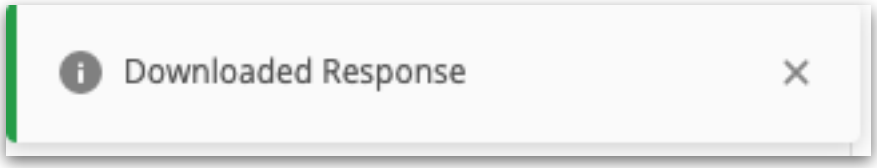
Response: (encoded text & symbols)

- Click on the "Save Response" Dropdown Arrow and Select "Save To a file"

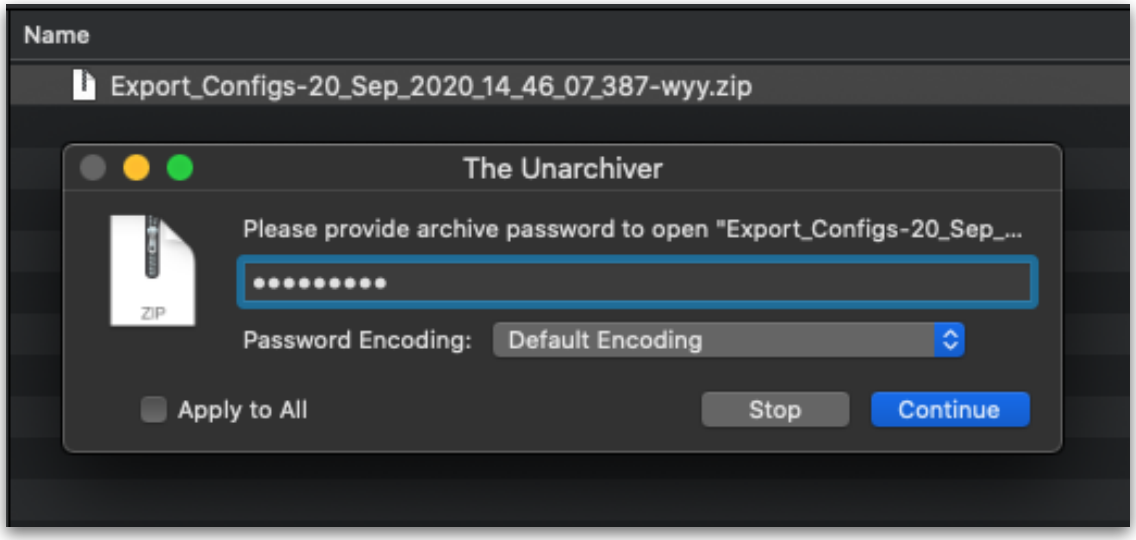


- Select the File Name, Select the Download location of the Zipped File, and Save





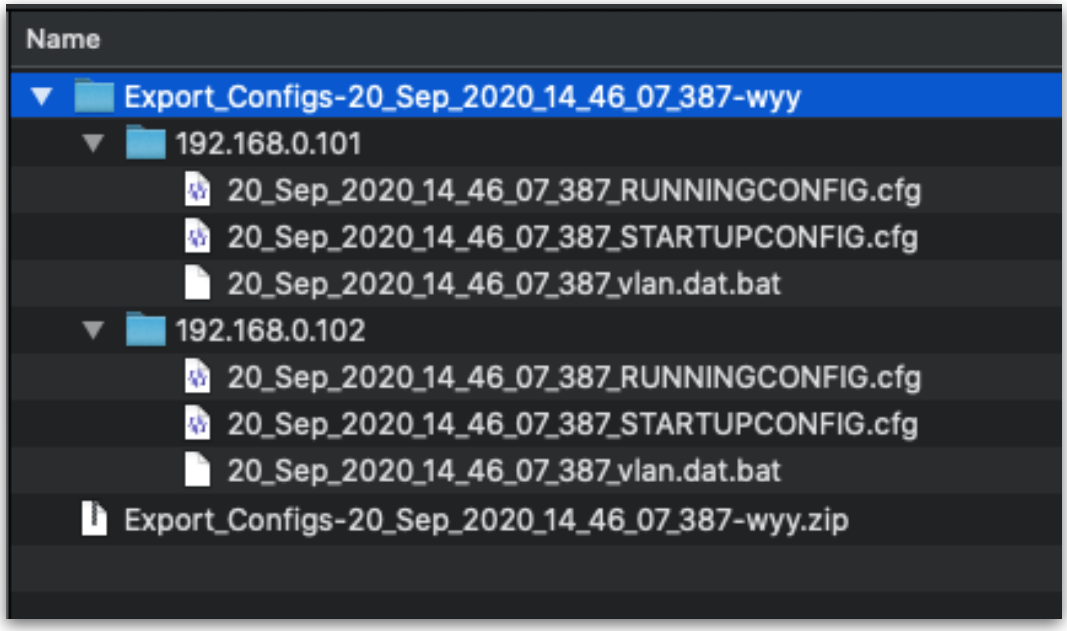
- Expand the downloaded zipped file
- Enter the Password used in Step 3



- Enclosed in the Directory for each Device will be the RUNNING CONFIG, STARTUP CONFIG, and VLAN DATA

Export_Configs-20_Sep_2020_14_46_07_387-wyy.zip

```
Configuration-Archive $ ls -lR *
-rw-r--r--@ 1 tdeleon  staff  28681 Sep 20 10:52 Export_Configs-20_Sep_2020_14_46_07_387-wyy.zip

Export_Configs-20_Sep_2020_14_46_07_387-wyy:
total 0
drwxr-xr-x@ 5 tdeleon  staff  160 Sep 20 10:58 192.168.0.101
drwxr-xr-x@ 5 tdeleon  staff  160 Sep 20 10:58 192.168.0.102

Export_Configs-20_Sep_2020_14_46_07_387-wyy/192.168.0.101:
total 112
-rwxr-xr-x@ 1 tdeleon  staff  28539 Sep 20  2020 20_Sep_2020_14_46_07_387_RUNNINGCONFIG.cfg
-rwxr-xr-x@ 1 tdeleon  staff  22623 Sep 20  2020 20_Sep_2020_14_46_07_387_STARTUPCONFIG.cfg
-rwxr-xr-x@ 1 tdeleon  staff    556 Sep 20  2020 20_Sep_2020_14_46_07_387_vlan.dat.bat

Export_Configs-20_Sep_2020_14_46_07_387-wyy/192.168.0.102:
total 112
-rwxr-xr-x@ 1 tdeleon  staff  28504 Sep 20  2020 20_Sep_2020_14_46_07_387_RUNNINGCONFIG.cfg
-rwxr-xr-x@ 1 tdeleon  staff  22588 Sep 20  2020 20_Sep_2020_14_46_07_387_STARTUPCONFIG.cfg
-rwxr-xr-x@ 1 tdeleon  staff    804 Sep 20  2020 20_Sep_2020_14_46_07_387_vlan.dat.bat
```

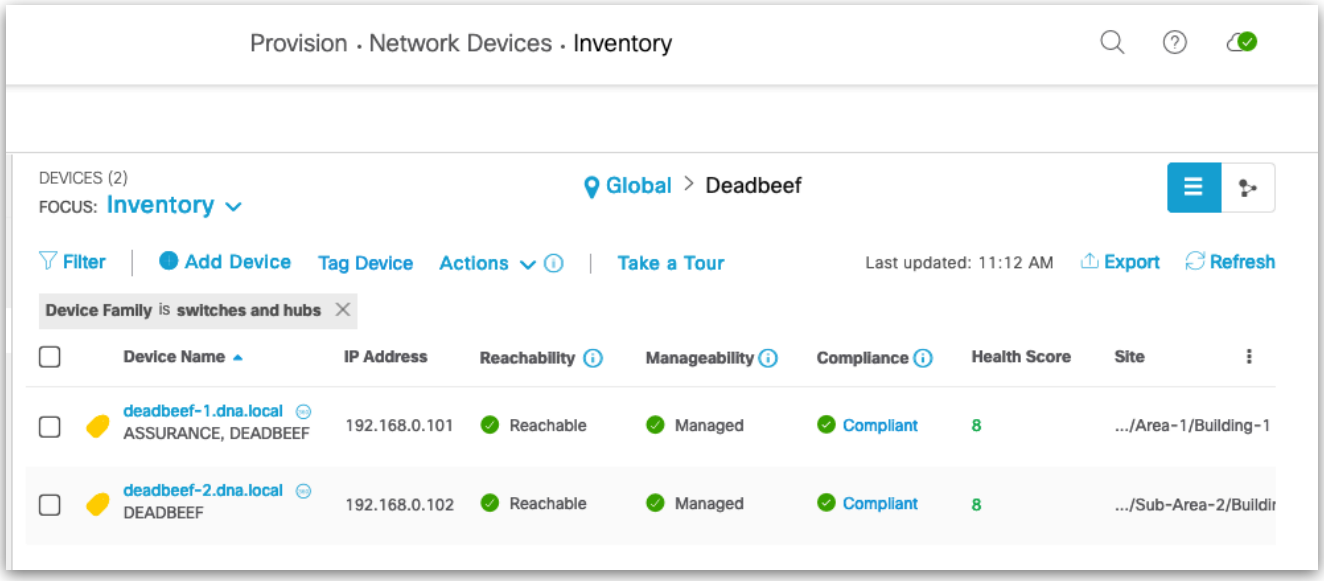



```
Configuration-Archive $ ls -lR *
-rw-r--r--@ 1 tdeleon  staff  28681 Sep 20 10:52 Export_Configs-20_Sep_2020_14_46_07_387-wyy.zip

Export_Configs-20_Sep_2020_14_46_07_387-wyy:
total 0
drwxr-xr-x@ 5 tdeleon  staff  160 Sep 20 10:58 192.168.0.101
drwxr-xr-x@ 5 tdeleon  staff  160 Sep 20 10:58 192.168.0.102

Export_Configs-20_Sep_2020_14_46_07_387-wyy/192.168.0.101:
total 112
-rwxr-xr-x@ 1 tdeleon  staff  28539 Sep 20  2020 20_Sep_2020_14_46_07_387_RUNNINGCONFIG.cfg
-rwxr-xr-x@ 1 tdeleon  staff  22623 Sep 20  2020 20_Sep_2020_14_46_07_387_STARTUPCONFIG.cfg
-rwxr-xr-x@ 1 tdeleon  staff    556 Sep 20  2020 20_Sep_2020_14_46_07_387_vlan.dat.bat

Export_Configs-20_Sep_2020_14_46_07_387-wyy/192.168.0.102:
total 112
-rwxr-xr-x@ 1 tdeleon  staff  28504 Sep 20  2020 20_Sep_2020_14_46_07_387_RUNNINGCONFIG.cfg
-rwxr-xr-x@ 1 tdeleon  staff  22588 Sep 20  2020 20_Sep_2020_14_46_07_387_STARTUPCONFIG.cfg
-rwxr-xr-x@ 1 tdeleon  staff    804 Sep 20  2020 20_Sep_2020_14_46_07_387_vlan.dat.bat
Configuration-Archive $
```

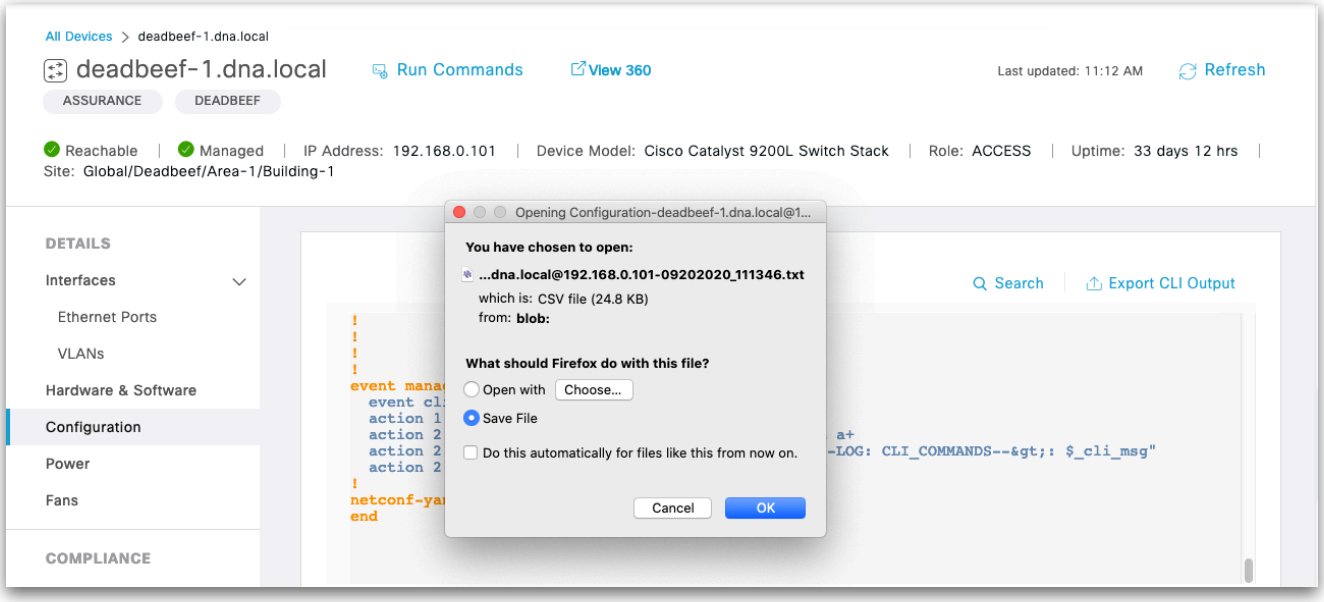
OTHER WAYS TO GET DEVICE CONFIGURATION INFO FROM THE CISCO DNA CENTER UI

Cisco DNA Center UI provides some other ways to get device configuration information:

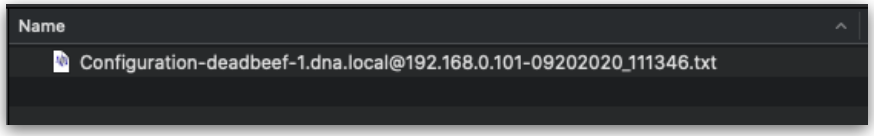
- Provision.Inventory.Device - **Export CLI Output**
- Provision.Inventory.Device - **Command Runner**



Export CLI Output

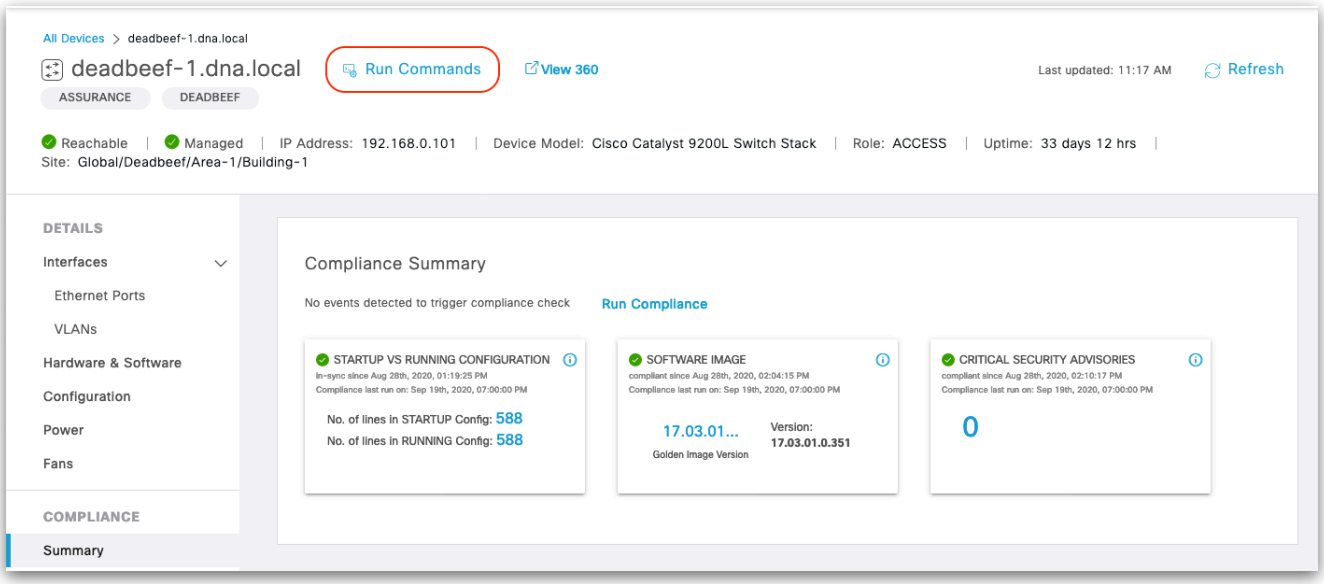


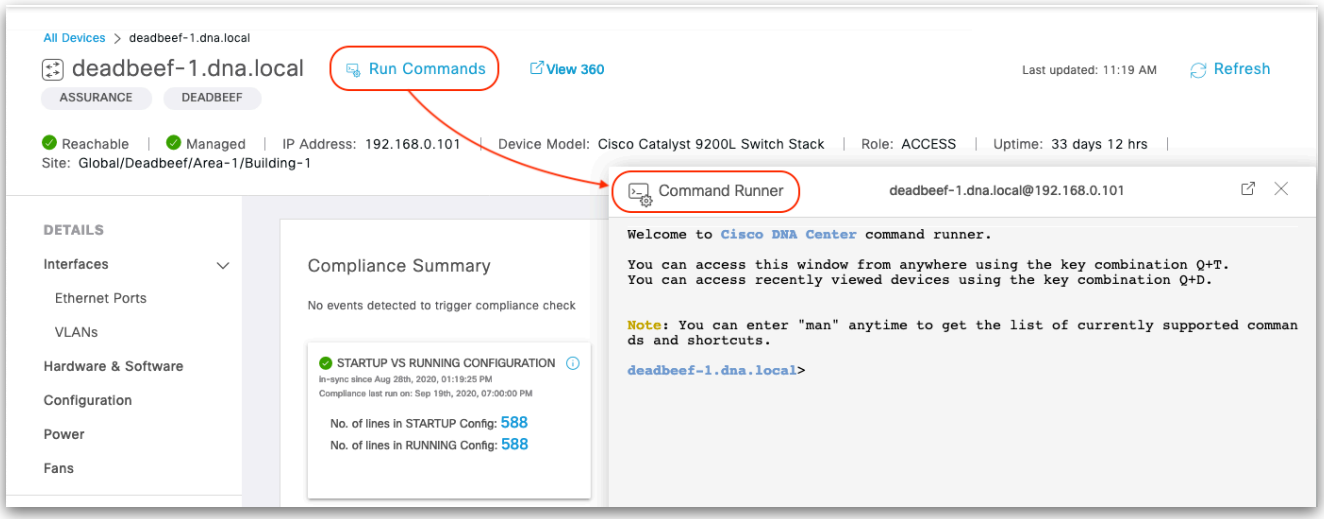
Export CLI Output (cont.)



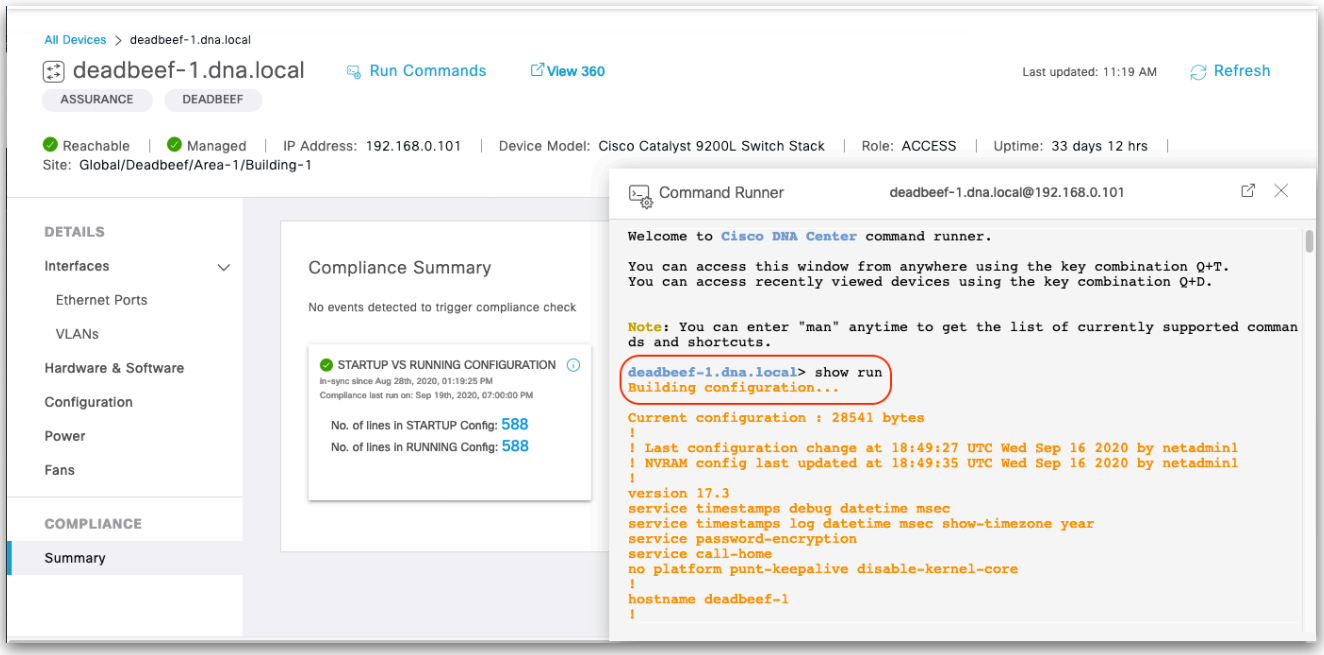

```
Configuration-Archive $ cat Configuration-deadbeef-1.dna.local@192.168.0.101-09202020_111346.txt


Building configuration...

Current configuration : 28541 bytes
!
! Last configuration change at 18:49:27 UTC Wed Sep 16 2020 by netadmin1
! NVRAM config last updated at 18:49:35 UTC Wed Sep 16 2020 by netadmin1
!
version 17.3
service timestamps debug datetime msec
service timestamps log datetime msec show-timezone year
service password-encryption
service call-home
no platform punt-keepalive disable-kernel-core
!
hostname deadbeef-1
!
!
vrf definition Mgmt-vrf
  !
  address-family ipv4
  exit-address-family
  !
  address-family ipv6
  exit-address-family
!
enable secret 8 xxxxxxxx
enable password 7 xxxxxxxx
!
!
!
!
aaa new-model
!
!
aaa group server tacacs+ dnac-network-tacacs-group
  server name dnac-tacacs_192.168.111.85
!
aaa group server radius dnac-client-radius-group
  server name dnac-radius_192.168.111.85
  ip radius source-interface Loopback0
!
aaa authentication login default local
aaa authentication login dnac-cts-list group dnac-client-radius-group local
aaa authentication login VTY_authen group dnac-network-tacacs-group local
aaa authentication dot1x default group dnac-client-radius-group
aaa authorization exec default local
aaa authorization exec VTY_author group dnac-network-tacacs-group local if-authenticated
aaa authorization network default group dnac-client-radius-group
aaa authorization network dnac-cts-list group dnac-client-radius-group
aaa accounting update newinfo periodic 2880
aaa accounting identity default start-stop group dnac-client-radius-group
aaa accounting exec default start-stop group dnac-network-tacacs-group
!
!
```

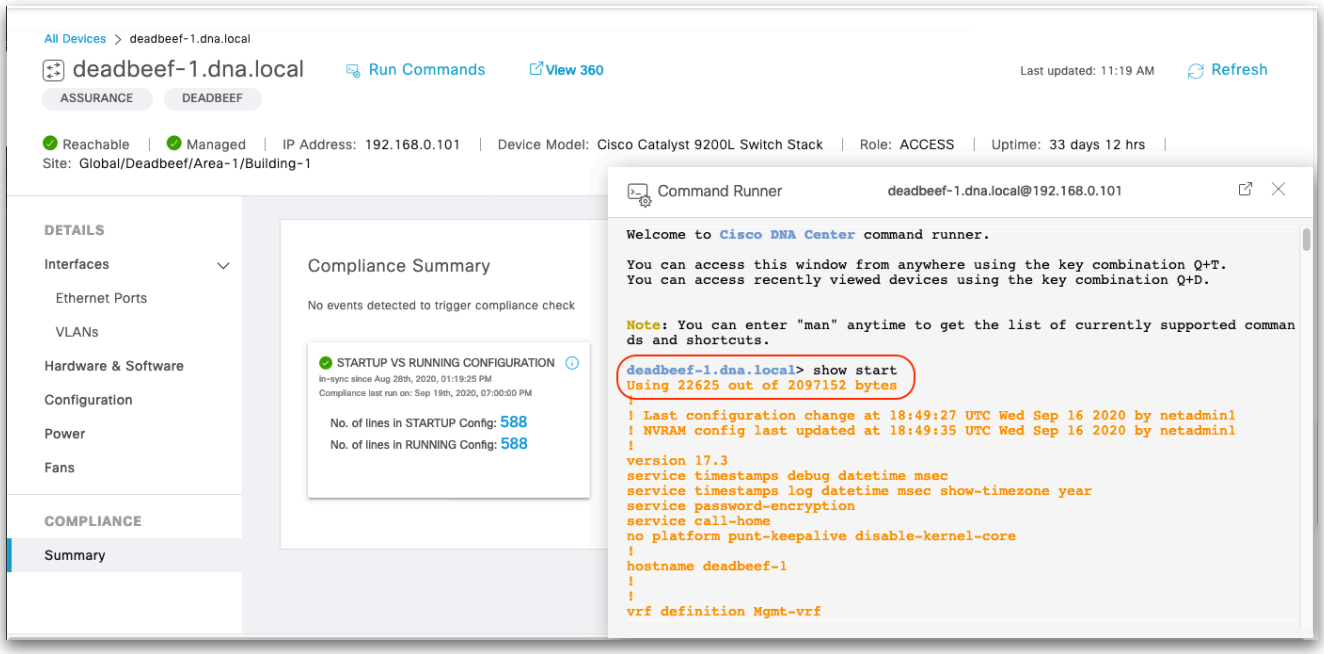
Command Runner

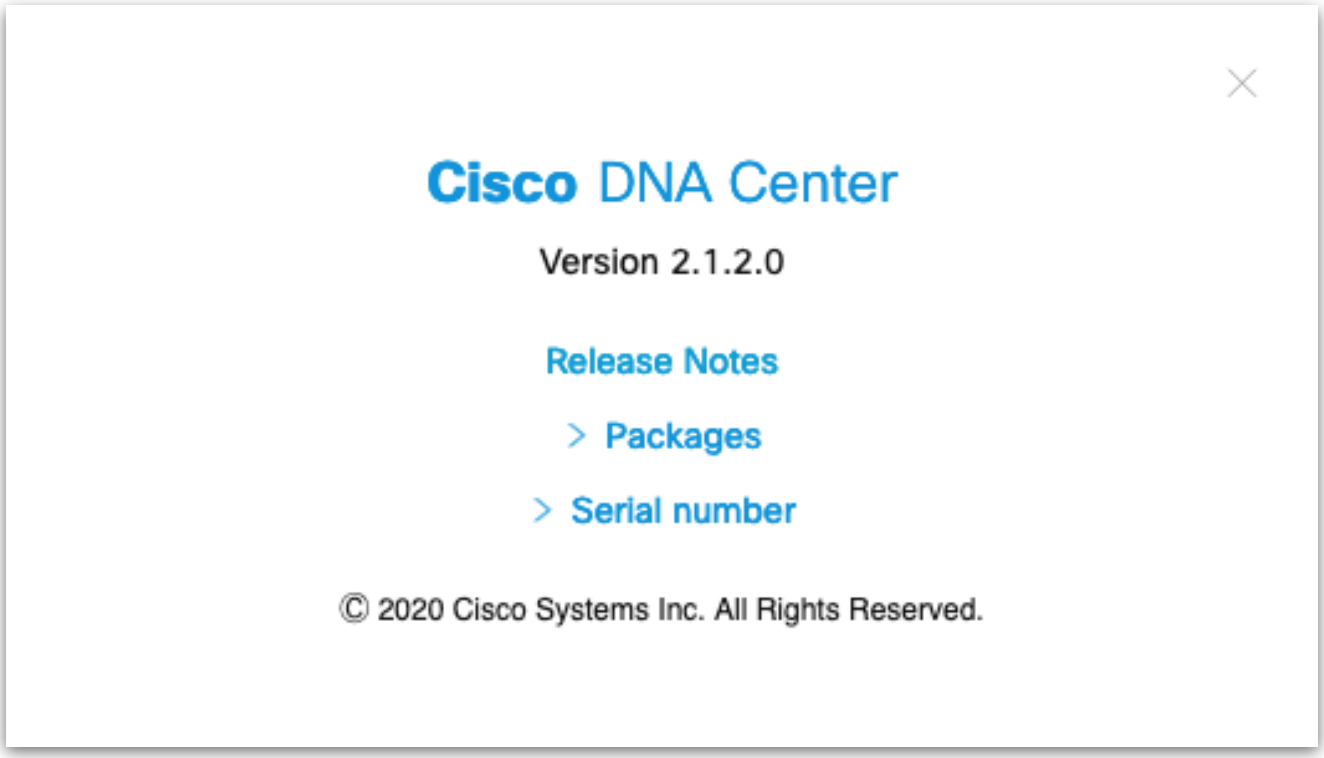




Command Runner (cont.)






Cisco DNA Center
Version 2.1.2.0
Release Notes
> Packages
> Serial number
© 2020 Cisco Systems Inc. All Rights Reserved.

The End.