

APIC-EM 1.3. - Certificate Generation - Deletion via API

Contents

[Introduction](#)

[Background Information](#)

[How will you get to know what is the current state of the device?](#)

[How do you ensure if APIC-EM also has the same certificate or if APIC-EM has understood the same certificate or not?](#)

[How to delete the certificate from the device?](#)

[How to Apply Certificate from APIC - EM?](#)

[Sometimes APIC-EM has the certificate but the device does not. How can you resolve it?](#)

Introduction

This document describes how to use the Cisco Application Policy Infrastructure Controller (APIC) - Extension Mobility (EM) API to create - delete the certificate. With IWAN, it is all automatically configured. However, IWAN at this moment does not have any flow to recover automatically device from expired certificate.

The good part is that there is some sort of flow in automation in terms of RestAPI. But, that automation is per device and it needs some information on the device. The RestAPI flow which is outside of IWAN flow, uses some mechanism to automate the certificate for device.

Background Information

Usual Customer Topology.

SPOKE --- HUB ----- APIC_EM [Controller]

These are the three situations:

- Certificate is expired.
- Certificate is not renewing.
- Certificate is not at all available.

How will you get to know what is the current state of the device?

Run the command **Switch# sh cry pki cert.**

```

HUB2#sh cry pki cert
Certificate
  Status: Available
  Certificate Serial Number (hex): 3C276CE6B6ABFA8D
  Certificate Usage: General Purpose
  Issuer:
    cn=sdn-network-infra-subca
  Subject:
    Name: HUB2
    cn=ASR1001_SSI161908CX_sdn-network-infra-iwan
    hostname=HUB2
  Validity Date:
    start date: 06:42:03 UTC Mar 28 2017
    end date: 07:42:03 UTC Mar 28 2017
  Associated Trustpoints: sdn-network-infra-iwan

CA Certificate
  Status: Available
  Certificate Serial Number (hex): 04
  Certificate Usage: General Purpose
  Issuer:
    cn=ca
  Subject:
    cn=sdn-network-infra-subca
  Validity Date:
    start date: 06:42:03 UTC Mar 28 2017
    end date: 07:42:03 UTC Mar 28 2017
  Associated Trustpoints: sdn-network-infra-iwan

```

If you see, there are two certificates and here you need to check Associated Trustpoint .

End date will usually be of one year and it should be greater than the start date.

If it is sdn-network-infra-iwan then it means from APIC-EM that you have ID as well as CA Certificate registered.

How do you ensure if APIC-EM also has the same certificate or if APIC-EM has understood the same certificate or not?

a. Show version from device and collect the serial number:

If you require further assistance please contact us by sending email to export@cisco.com.

```

License Type: RightToUse
License Level: adventerprise
Next reload license Level: adventerprise

```

```

cisco ASR1001 (1RU) processor (revision 1RU) with 1062861K/6147K bytes of memory.
Processor board ID SSI161908CX
4 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
7741439K bytes of eUSB flash at bootflash:.

```

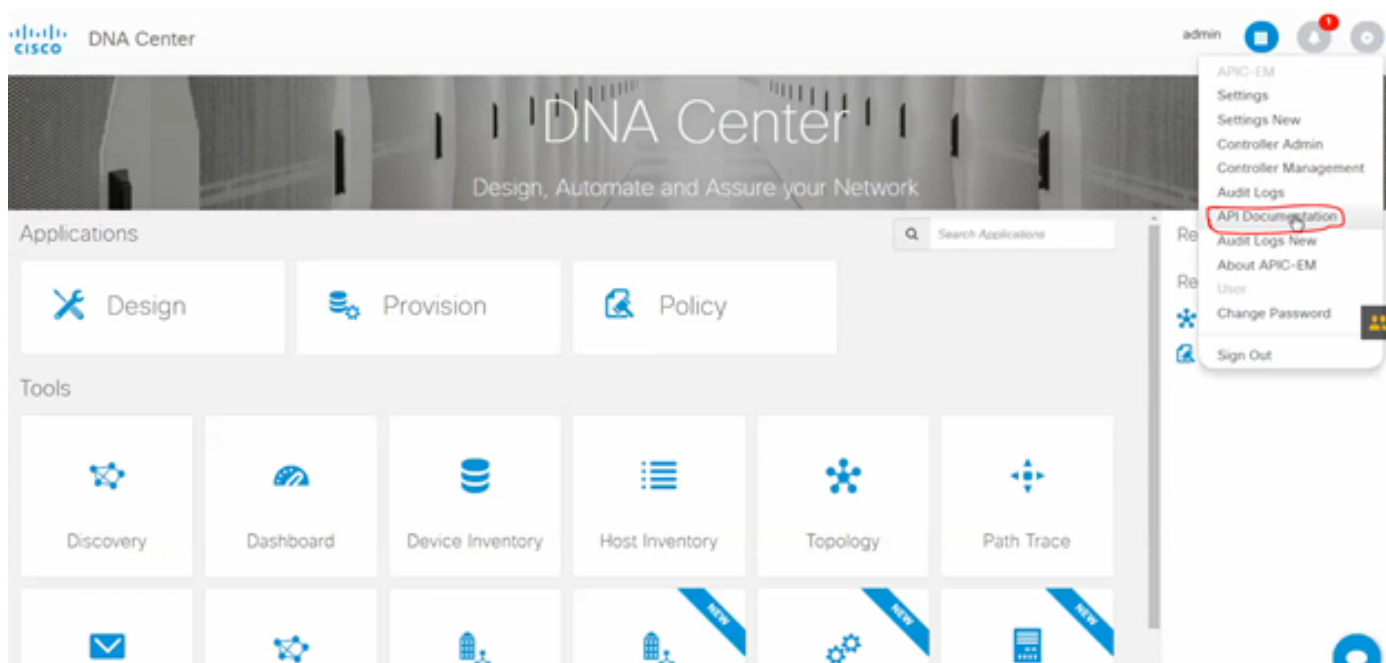
```

Configuration register is 0x0

```

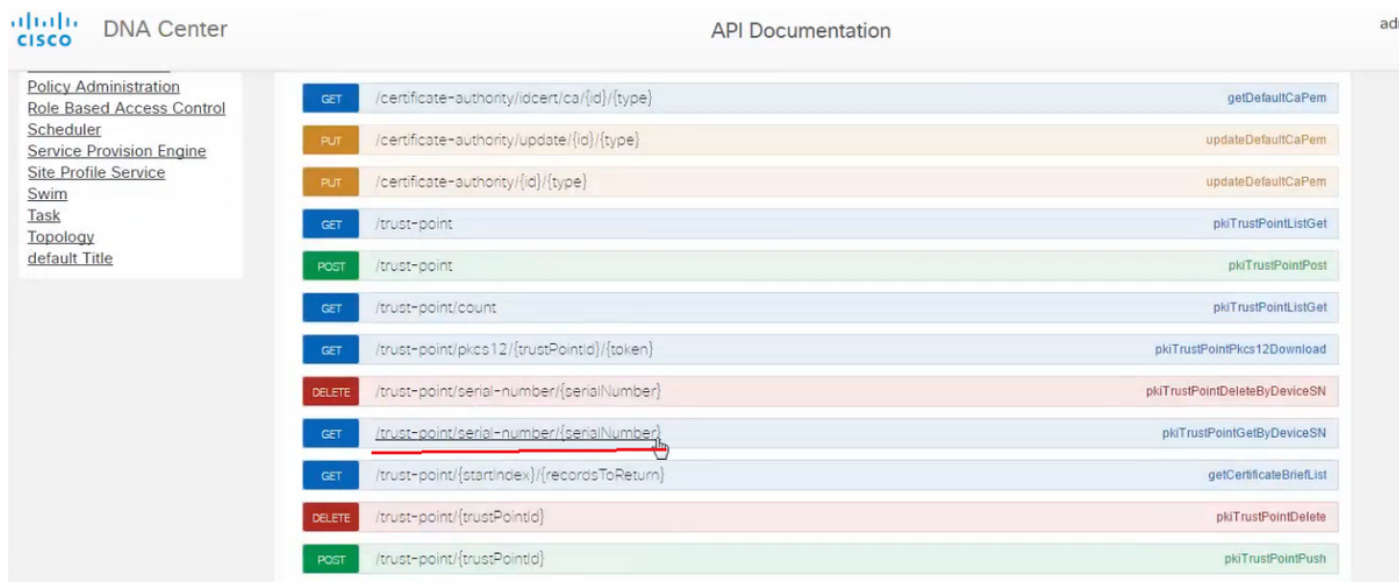
With the help of this serial number you can perform APIC-EM query to find out what APIC-EM thinks about this device.

b. Navigate to API Documentation.



c. Click on Public Key Infrastructure (PKI) Broker.

d. Click on First API which will help us know the status from the API side.



Click on **GET**.

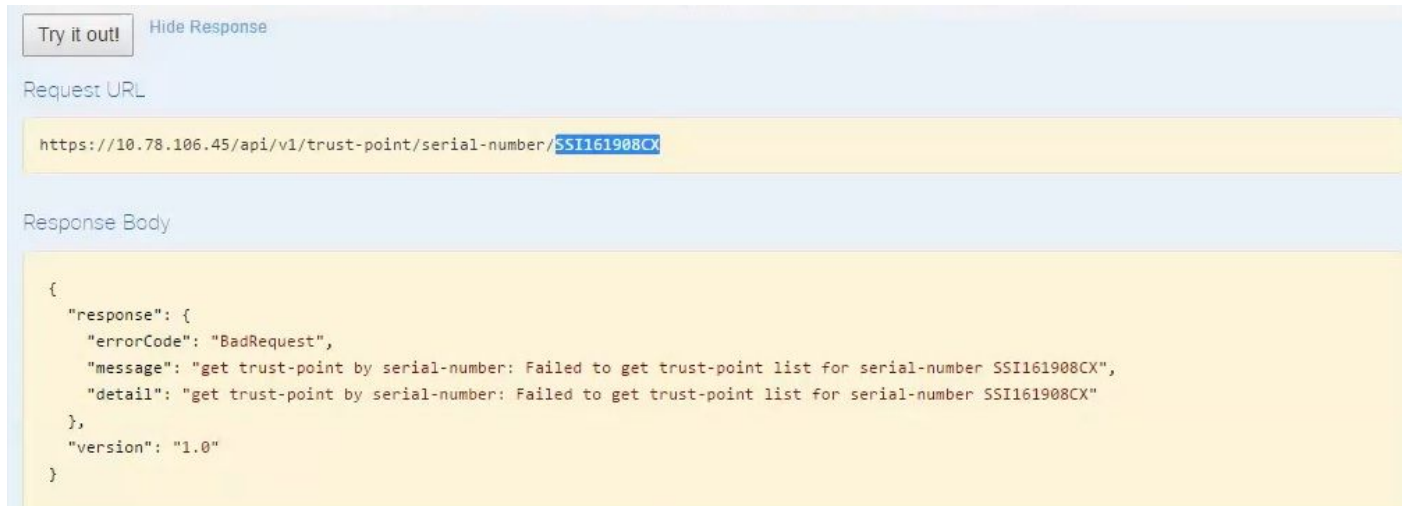
On one checkbox, click on serial number collected from show version output of Device.

Click on **Try it out!**.

Compare the output value with `sh crp pki cert` output of the device.

How to delete the certificate from the device?

It happens sometimes that on the device, certificate is there and in the APIC-EM it is not there. Which is why, when you run **GET API** you get an error message.



Try it out! Hide Response

Request URL

```
https://10.78.106.45/api/v1/trust-point/serial-number/SSI161908CX
```

Response Body

```
{
  "response": {
    "errorCode": "BadRequest",
    "message": "get trust-point by serial-number: Failed to get trust-point list for serial-number SSI161908CX",
    "detail": "get trust-point by serial-number: Failed to get trust-point list for serial-number SSI161908CX"
  },
  "version": "1.0"
}
```

The solution is only one and that is to delete the certificate from device:

a. **Switch# show run | I trustpoint**

```
HUB2#sh run | i trustpoint
crypto pki trustpoint zxz
crypto pki trustpoint sdn-network-infra-iwan
HUB2#
```

Run command **Switch# no crypto pki trustpoint <trustpoint name>**.

```
HUB2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
HUB2(config)#no crypto pki trustpoint sdn-network-infra-iwan
% Removing an enrolled trustpoint will destroy all certificates
received from the related Certificate Authority.

Are you sure you want to do this? [yes/no]: yes
% Be sure to ask the CA administrator to revoke your certificates.

HUB2(config)#
```

This command deletes all the Certificate on device associated with selected trustpoint.

Re-check if certificate is deleted.

Use the command: **Switch# sh cry pki cert.**

It should not show sdn trustpoint which was deleted.

b. Deletion of Key:

Run command on device: **Switch# sh cry key mypubkey all.**

Here you will see that the Key name starts with **sdn-network-infra**.

Command to delete the Key:

```
HUB2(config)#cry key zeroize rsa sdn-network-infra-iwan
& Keys to be removed are named 'sdn-network-infra-iwan'.
& All router certs issued using these keys will also be removed.
Do you really want to remove these keys? [yes/no]: yes
HUB2(config)#
```

2. Ensure that the APIC-EM interface which is connected to the device should be Pingable.

It might happen that APIC-EM has two interfaces out of which one is Public and the other is private. In that case, ensure that the APIC-EM interface which communicates to the device ping each other.

```
HUB2#ping 10.10.10.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.10, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
HUB2#
```

How to Apply Certificate from APIC - EM?

Under APIC-EM, when API Documentation is clicked and PKI Broker selected, this option is available.

[POST/trust-point](#)

- This will create certificate with-in APIC - EM.

The screenshot displays the API documentation for the PKI Broker Service. On the left, a navigation menu lists various services, with 'PKI Broker Service' selected. The main content area shows a list of API endpoints. The 'POST /trust-point' endpoint is circled in red. Below the endpoint list, the 'Implementation Notes' section states: 'This method is used to create a trust-point'. The 'Response Class' section shows the 'TaskIdResult' and 'TaskIdResponse' classes. The 'Model Schema' section shows the 'TaskId' class. The 'Response Content Type' is listed as 'application/json'.

Then you need have information on the device and click on try it out.

Response Class

Model | Model Schema

```

TaskIdResult {
  version (string, optional),
  response (TaskIdResponse, optional)
}
TaskIdResponse {
  taskId (TaskId, optional),
  url (string, optional)
}
TaskId {
}

```

Response Content Type: application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
pkitrustPointInput	<pre>{ "platformId": "ASR1001", "serialNumber": "SSI161908CX", "trustProfileName": "sdn-network-infra-iwan", "entityType": "router", "entityName": "HUB2" }</pre>	pkitrustPointInput	body	Model Model Schema PkitrustPoint { serialNumber (string): Devices serial-number, entityName (string): Devices hostname, id (string, optional): Trust-point identification. Automatically generated, platformId (string): Platform identification. Eg. ASR1000, trustProfileName (string): Name of trust-profile (must already exist). Default: sdn-network-infra-iwan, entityType (string, optional): Available options: router.

Parameter content type: application/json

Example:

```

{
  "platformId": "ASR1001",
  "serialNumber": "SSI161908CX",
  "trustProfileName": "sdn-network-infra-iwan",
  "entityType": "router",
  "entityName": "HUB2"
}

```

- The highlighted information is STATIC and rest of all is Dynamic.
- Entity name is Hostname of the device.
- Serial number you got from the show version of the device.
- Entity type you can change based on device type.
- This information is needed to tell APIC-EM to configure the device. Here APIC-EM understands the serial number.

Output of Try it out!:

Response Body

```
{
  "response": {
    "taskId": "1a395ed1-1730-43fa-9527-327ed3e6e12b",
    "url": "/api/v1/task/1a395ed1-1730-43fa-9527-327ed3e6e12b"
  },
  "version": "1.0"
}
```

Response Code

202

Response Headers

```
{
  "Pragma": "no-cache, no-cache",
  "Content-Security-Policy": "style-src 'self' 'unsafe-inline'; script-src 'self' 'unsafe-eval' 'unsafe-inline' 'nonce-2dcc163f-98f3-45e2-bd5b-",
  "X-Frame-Options": "SAMEORIGIN, SAMEORIGIN",
  "Date": "Tue, 28 Mar 2017 10:10:06 GMT",
  "Strict-Transport-Security": "max-age=31536000; includeSubDomains, max-age=31536000; includeSubDomains",
  "Content-Type": "application/json;charset=UTF-8",
  "Access-Control-Allow-Origin": "https://10.78.106.45",
  "Cache-Control": "no-cache, no-store, no-cache, no-store",
  "Transfer-Encoding": "chunked",
  "Access-Control-Allow-Credentials": "false"
}
```

This output means that the file is created internally by APIC-EM and is now ready to deploy on the device.

Next step is to push this device into the bundle. To push, you need to get trust point ID. This can be done via GET API CALL.

[GET/trust-point/serial-number/{serialNumber}](#) - Query

GET /trust-point/serial-number/{serialNumber} pkTrustPointGetByDeviceSN

Implementation Notes
This method is used to return a specific trust-point by its device serial-number

Response Class
Model | Model Schema

PkiTrustPointResult {
version (string, optional)
response (PkiTrustPoint, optional)
}

PkiTrustPoint {
serialNumber (string): Devices serial-number.
entityName (string): Devices hostname.
id (string, optional): Trust-point identification. Automatically generated.
platformId (string): Platform identification. Eg. ASR1006.
trustProfileName (string): Name of trust-profile (must already exist). Default: sdn-network-infra-iwan.
entityType (string, optional): Available options: router, switch. Currently not used.
networkDeviceId (string, optional): Device identification. Currently not used.
certificateAuthorityId (string, optional): CA identification. Automatically populated.
controllerIpAddress (string, optional): IP address device uses to connect to APIC-EM. Eg. Proxy server IP address. Automatically populated if not set.
attributeInfo (object, optional)
}

Response Content Type: application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
serialNumber	551161908CX	Device serial-number	path	string

Error Status Codes

It will give you this output. It means that the APIC-EM has the certificate with this to push on the device.

Response Body

```

{
  "response": {
    "platformId": "ASR1001",
    "serialNumber": "SSI161908CX",
    "trustProfileName": "sdn-network-infra-iwan",
    "entityName": "HUB2",
    "entityType": "router",
    "certificateAuthorityId": "f0bd5040-3f04-4e44-94d8-de97b8829e8d",
    "attributeInfo": {},
    "id": "2b832bf6-9061-44bd-a773-fb5256e544fb"
  },
  "version": "1.0"
}

```

Response Code

200

Push the certificate to the device.

[POST/trust-point/{trustPointId}](#) // trustPointId needs to be copied from GET Serial Number Query

```

{"response": { "platformId": "ASR1001", "serialNumber": "SSI161908CX", "trustProfileName": "sdn-network-infra-iwan", "entityName": "HUB2", "entityType": "router", "certificateAuthorityId": "f0bd5040-3f04-4e44-94d8-de97b8829e8d", "attributeInfo": {}, "id": "c4c7d612-9752-4be5-88e5-e2b6f137ea13" }, "version": "1.0" }

```

This will push the certificate to device – provided there is proper connectivity.

POST	/trust-point/{trustPointId}	pkiTrustPointPush
GET	/trust-point/{trustPointId}	pkiTrustPointGet
GET	/trust-point/{trustPointId}/config	pkiTrustPointConfigGet
GET	/trust-point/{trustPointId}/downloaded	checkPKCS12Downloaded

[BASE URL: https://10.78.106.45/api/v1/api-docs/pki-broker-service . API VERSION: 1.0]

Parameters

Parameter	Value	Description	Parameter Type	Data Type
trustPointId	2b832bf6-9061-44bd-a773-fb5256e544fb	Trust-point ID	path	string

Error Status Codes

HTTP Status Code	Reason
200	The request was successful. The result is contained in the response body.
201	The POST/PUT request was fulfilled and a new resource has been created. Information about the resource is in the response body.
202	The request was accepted for processing, but the processing has not been completed.
204	The request was successful, however no content was returned.
206	The GET request included a Range Header, and the server responded with the partial content matching the range.
400	The client made a request that the server could not understand (for example, the request syntax is incorrect).
401	The client's authentication credentials included with the request are missing or invalid.
403	The server recognizes the authentication credentials, but the client is not authorized to perform this request.
404	The client made a request for a resource that does not exist.
500	The server could not fulfill the request.
501	The server has not implemented the functionality required to fulfill the request.
503	The server is (temporarily) unavailable.
504	The server did not respond inside time restrictions and timed-out.
409	The target resource is in a conflicted state (for example, an edit conflict where a resource is being edited by multiple users). Retrying the request later might succeed.
415	The client sent a request body in a format that the server does not support (for example, XML to a server that only accepts JSON).

Try it out!

Response Success Message:

Try it out! Hide Response

Request URL

```
https://10.78.106.45/api/v1/trust-point/2b832bf6-9061-44bd-a773-fb5256e544fb
```

Response Body

```
{
  "response": {
    "taskId": "f10022bd-8f45-4597-8160-bcc07fd55898",
    "url": "/api/v1/task/f10022bd-8f45-4597-8160-bcc07fd55898"
  },
  "version": "1.0"
}
```

Response Code

```
202
```

Response Headers

Recheck on device:

You see that both the certificates are now pasted:

```
HUB2#sh cry pki cert
Certificate
  Status: Available
  Certificate Serial Number (hex): 2AD39646370CACC7
  Certificate Usage: General Purpose
  Issuer:
    cn=sdn-network-infra-ca
  Subject:
    Name: HUB2
    cn=ASR1001_SSI161908CX_sdn-network-infra-iwan
    hostname=HUB2
  Validity Date:
    start date: 10:00:07 UTC Mar 28 2017
    end   date: 10:00:07 UTC Mar 28 2018
    renew date: 10:00:06 UTC Jan 14 2018
  Associated Trustpoints: sdn-network-infra-iwan

CA Certificate
  Status: Available
  Certificate Serial Number (hex): 5676260082D447A3
  Certificate Usage: Signature
  Issuer:
    cn=sdn-network-infra-ca
  Subject:
    cn=sdn-network-infra-ca
  Validity Date:
    start date: 09:20:26 UTC Mar 28 2017
    end   date: 09:20:26 UTC Mar 27 2022
  Associated Trustpoints: sdn-network-infra-iwan

HUB2#
```

Sometimes APIC-EM has the certificate but the device does not. How can you resolve it?

There is some background task through which you can delete certificate from only APIC-EM. Sometimes, the customer by mistake deletes the certificate from the device but in APIC-EM, it is still there. Click on **DELETE**.

[DELETE/trust-point/serial-number/{serialNumber}](#) - Delete.

GET	/trust-point/count	pkITrustPointListGet
GET	/trust-point/pkcs12/{trustPointId}/{token}	pkITrustPointPkcs12Download
DELETE	/trust-point/serial-number/{serialNumber}	pkITrustPointDeleteByDeviceSN
GET	/trust-point/serial-number/{serialNumber}	pkITrustPointGetByDeviceSN

Implementation Notes

This method is used to return a specific trust-point by its device serial-number

Response Class

Model Model Schema

PkiTrustPointResult {
 version (string, optional),
 response (PkiTrustPoint, optional)
}

Enter the serial number and click **Try it out!**.

Parameters

Parameter	Value	Description	Parameter Type	Data Type
serialNumber	SSI161908CX	Device serial-number	path	string

Error Status Codes

HTTP Status Code	Reason
200	The request was successful. The result is contained in the response body.
204	The request was successful, however no content was returned.
206	The GET request included a Range Header, and the server responded with the partial content matching the range.
400	The client made a request that the server could not understand (for example, the request syntax is incorrect).
401	The client's authentication credentials included with the request are missing or invalid.
403	The server recognizes the authentication credentials, but the client is not authorized to perform this request.
404	The client made a request for a resource that does not exist.
500	The server could not fulfill the request.
501	The server has not implemented the functionality required to fulfill the request.
503	The server is (temporarily) unavailable.
504	The server did not respond inside time restrictions and timed-out.
409	The target resource is in a conflicted state (for example, an edit conflict where a resource is being edited by multiple users). Retrying the request later might succeed.
415	The client sent a request body in a format that the server does not support (for example, XML to a server that only accepts JSON).

Try it out!

```
{
  "response": {
    "taskId": "33ab0da8-9be1-40b7-86c2-cf2e501ebbb5",
    "url": "/api/v1/task/33ab0da8-9be1-40b7-86c2-cf2e501ebbb5"
  },
  "version": "1.0"
}
```

Response Code

202

Response Headers

```
{
  "Pragma": "no-cache, no-cache",
  "Content-Security-Policy": "style-src 'self' 'unsafe-inline'; script-src 'self' 'unsafe-eval' 'unsafe-inline' 'nonce-f59e75bb-2a28-4fe8-a954-",
  "X-Frame-Options": "SAMEORIGIN, SAMEORIGIN",
  "Date": "Tue, 28 Mar 2017 10:15:23 GMT",
  "Strict-Transport-Security": "max-age=31536000; includeSubDomains, max-age=31536000; includeSubDomains",
  "Content-Type": "application/json;charset=UTF-8",
  "Access-Control-Allow-Origin": "https://10.78.106.45",
  "Cache-Control": "no-cache, no-store, no-cache, no-store",
  "Transfer-Encoding": "chunked",
  "Access-Control-Allow-Credentials": "false"
}
```