

Add a host and deploy components using APIs

If you are using APIs to add host to an existing cluster involves the following steps:

- Install ambari-agent on the host to be added to the cluster
 - Ensure that the agent installed be the same version as the server
 - Configure the agent to register with the server (edit /etc/ambari-agent/conf/ambari-agent.ini)

Adding a host and then installing components

Verified against releases 1.4.1/1.2.5

Starting 1.4.2/1.4.3 you will have to add -H option to the curl calls. E.g. -H "X-Requested-By: ambari"

1. Ensure the host is registered properly.

```
curl -u admin:admin http://AMBARI_SERVER_HOST:8080/api/v1/hosts

Sample OUTPUT
{
  "href" : "http://AMBARI_SERVER_HOST:8080/api/v1/hosts",
  "items" : [
    {
      "href" : "http://AMBARI_SERVER_HOST:8080/api/v1/hosts/NEW_HOST_ADDED",
      "Hosts" : {
        "host_name" : "NEW_HOST_ADDED"
      }
    }
  ]
}
```

If you do not see the host resource then verify that ambari-agent registered successfully with the ambari-server. See through the logs to check what host-name was used for registration.

2. Add the host to the cluster.

```
curl --user admin:admin -i -X POST http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/hosts/NEW_HOST_ADDED
```

3. Ensure the host is added to the cluster.

```
curl --user admin:admin -i -X GET http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/hosts/NEW_HOST_ADDED
```

4. Add the necessary host components to the host.

These steps are applicable when the components being added are member of service already added. If you are adding hosts to install components of a new service then you need to add the service itself to the installed stack first. This may require adding new config types as well.

Ensure that ganglia monitor is also installed on the new hosts.

```
curl --user admin:admin -i -X POST http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/hosts/NEW_HOST_ADDED/host_components/DATANODE
curl --user admin:admin -i -X POST http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/hosts/NEW_HOST_ADDED/host_components/GANGLIA_MONITOR
```

5. Install the components.

This step will create requests that you can monitor for progress and result.

```
curl --user admin:admin -i -X PUT -d '{"HostRoles": {"state": "INSTALLED"}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/hosts/NEW_HOST_ADDED/host_components/GANGLIA_MONITOR
curl --user admin:admin -i -X PUT -d '{"HostRoles": {"state": "INSTALLED"}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/hosts/NEW_HOST_ADDED/host_components/DATANODE
```

Each install request will return a request id that can be used to monitor progress.

Sample OUTPUT

```
{
  "href" : "http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/requests/9",
  "Requests" : {
    "id" : 9,
    "status" : "InProgress"
  }
}
```

Monitor the request and tasks within the request for progress

```
curl --user admin:admin -i -X GET http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/requests/9
curl --user admin:admin -i -X GET http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/requests/9/tasks/101

{
  "href" : "http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/requests/9/tasks/101",
  "Tasks" : {
    ...
    "status" : "COMPLETED",
    ...
  }
}
```

Use the following to access all tasks for a given request

```
curl -u admin:admin http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/requests/9?fields=tasks/Tasks/*
```

6. Start the components.

This step will create requests that you can monitor for progress and result. Start the components after install has been successful.

```
curl --user admin:admin -i -X PUT -d '{"HostRoles": {"state": "STARTED"}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/hosts/NEW_HOST_ADDED/host_components/GANGLIA_MONITOR
curl --user admin:admin -i -X PUT -d '{"HostRoles": {"state": "STARTED"}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/hosts/NEW_HOST_ADDED/host_components/DATANODE
```

Each install request will return a request id that can be used to monitor progress.

Client components, such as HDFS_CLIENT, need not be started. Nagios server may need to be restarted to ensure it picks up the new host and reports its status.

7. Restart Nagios.

```
curl --user admin:admin -i -X PUT -d '{"RequestInfo": {"context": "Stop Nagios"}, "ServiceInfo": {"state": "INSTALLED"}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/services/NAGIOS
curl --user admin:admin -i -X PUT -d '{"RequestInfo": {"context": "Start Nagios"}, "ServiceInfo": {"state": "STARTED"}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTER_NAME/services/NAGIOS
```