

Using APIs to delete a service or all host components on a host

At times you may need to delete a service or all host components on a host. This article talks about how to do that using APIs. *Be aware of losing data when performing any delete operations. Ensure that your data is backed up and decommissioning of nodes are performed as needed.*

From 1.4.3 onwards you need to add "X-Requested-By" header to all curl calls.

E.g. curl -i -u admin:admin -H "X-Requested-By: ambari" -d '{"HostRoles": {"state": "STARTED"}}' -X PUT 'http://localhost:8080/api/v1/clusters/c1/host_components?HostRoles/stale_configs=false'

Removing a Service (2.1.0)

Verified against releases 2.1.0

Note: These API calls do not uninstall the packages associated with the service and neither they remove the config or temp folders associated with the service components.

Before the PUT or DELETE calls, you can do a GET to ensure that the API is referring to a valid resource.

1. Note all the host components associated with the service.

```
curl -u admin:admin -H "X-Requested-By: ambari" -X GET http://AMBARI_SERVER_HOST:8080/api/v1/clusters/c1/services/SERVICENAME
```

2. Ensure the service is stopped (you can use the Ambari Web-UI to stop the service as well)

Stop the whole service (ensure correct values are provided for AMBARI_SERVER_HOST, SERVICE_NAME):

```
curl -u admin:admin -H "X-Requested-By: ambari" -X PUT -d '{"RequestInfo":{"context":"Stop Service"},"Body":{"ServiceInfo":{"state":"INSTALLED"}}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/c1/services/SERVICE_NAME
```

Stop individual components (ensure correct values are provided for AMBARI_SERVER_HOST, HOSTNAME, COMPONENT_NAME):

```
curl -u admin:admin -H "X-Requested-By: ambari" -X PUT -d '{"RequestInfo":{"context":"Stop Component"},"Body":{"HostRoles":{"state":"INSTALLED"}}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/c1/hosts/HOSTNAME/host_components/COMPONENT_NAME
```

Stop all component instances (ensure correct values are provided for AMBARI_SERVER_HOST, SERVICE_NAME, COMPONENT_NAME) - *just another way to stop Service members.*

```
curl -u admin:admin -H "X-Requested-By: ambari" -X PUT -d '{"RequestInfo":{"context":"Stop All Components"},"Body":{"ServiceComponentInfo":{"state":"INSTALLED"}}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/c1/services/SERVICE_NAME/components/COMPONENT_NAME
```

3. Delete the whole SERVICE

```
curl -u admin:admin -H "X-Requested-By: ambari" -X DELETE http://AMBARI_SERVER_HOST:8080/api/v1/clusters/c1/services/SERVICENAME
```

Removing a Service (1.2.3)

Verified against releases 1.2.3

Note: These API calls do not uninstall the packages associated with the service and neither they remove the config or temp folders associated with the service components.

Before the PUT or DELETE calls, you can do a GET to ensure that the API is referring to a valid resource.

1. Note all the host components associated with the service.

```
curl -u admin:admin -X GET http://AMBARI_SERVER_HOST:8080/api/v1/clusters/cl/services/SERVICENAME
```

2. Ensure the service is stopped (you can use the Ambari Web-UI to stop the service as well)

Stop the whole service (ensure correct values are provided for AMBARI_SERVER_HOST, SERVICE_NAME):

```
curl -u admin:admin -X PUT -d '{"RequestInfo":{"context":"Stop Service"},"Body":{"ServiceInfo":{"state":"INSTALLED"}}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/cl/services/SERVICE_NAME
```

Stop individual components (ensure correct values are provided for AMBARI_SERVER_HOST, HOSTNAME, COMPONENT_NAME):

```
curl -u admin:admin -X PUT -d '{"RequestInfo":{"context":"Stop Component"},"Body":{"HostRoles":{"state":"INSTALLED"}}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/cl/hosts/HOSTNAME/host_components/COMPONENT_NAME
```

3. Put MASTER and SLAVE components into MAINTENANCE mode

```
curl -u admin:admin -X PUT -d '{"HostRoles":{"state":"MAINTENANCE"}}' http://AMBARI_SERVER_HOST:8080/api/v1/clusters/cl/hosts/HOSTNAME/host_components/COMPONENTNAME
```

4. Delete MASTER and SLAVE components

```
curl -u admin:admin -X DELETE http://AMBARI_SERVER_HOST:8080/api/v1/clusters/cl/hosts/HOSTNAME/host_components/COMPONENTNAME
```

5. Delete the whole SERVICE

```
curl -u admin:admin -X DELETE http://AMBARI_SERVER_HOST:8080/api/v1/clusters/cl/services/SERVICENAME
```

Removing a Host (2.1.0)

Verified against releases 2.1.0

The preferred way to remove a host is to move the master services from the host, decommission the slave nodes and then remove the host *after* deleting all the host components. However, there are situations such as the host is lost and cannot be brought back online for graceful removal. Under this circumstances the following API calls can be used to clean up the host.

1. Get a list of host components mapped to the host

```
curl -u admin:admin -H "X-Requested-By: ambari" -X GET http://AMBARI_SERVER_HOST:8080/api/v1/clusters/cl/hosts/HOSTNAME
```

See the property, "**host_components**" for all the host.

Sample:

Output:

```
...
"host_components" : [
  {
    "href" : "http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTERNAME/hosts/HOSTNAME/host_components/DATANODE",
    "HostRoles" : {
      "cluster_name" : "CLUSTERNAME",
      "component_name" : "DATANODE",
      "host_name" : "HOSTNAME"
    }
  },
  {
    "href" : "http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTERNAME/hosts/HOSTNAME/host_components/GANGLIA_MONITOR",
    "HostRoles" : {
      "cluster_name" : "CLUSTERNAME",
      "component_name" : "GANGLIA_MONITOR",
      "host_name" : "HOSTNAME"
    }
  }
]
...
```

2. DELETE all host components mapped to this host

```
E.g. Delete DATANODE
curl -u admin:admin -H "X-Requested-By: ambari" -X DELETE http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTERNAME/hosts/HOSTNAME/host_components/DATANODE
```

3. DELETE the host

```
curl -u admin:admin -H "X-Requested-By: ambari" -X DELETE http://AMBARI_SERVER_HOST:8080/api/v1/clusters/CLUSTERNAME/hosts/HOSTNAME
```