120: Application Management through the REST API

Thursday, November 19, 2015

1:55 PM

ACS exposes the REST endpoints but they are not open to the outside world. In order to use these endpoints you must first create an SSH tunnel through which calls will be made, see [108: Connecting to Orchestration REST API](onenote:#108%20Connecting%20to%20Orchestration%20Management%20Interfaces&section-id={CFB761B4-3A3C-488C-A3F0-A31102E5F7F5}&page-id={B28097A5-BEB4-47B7-B5F0-7680ED286660}&end&base-path=https://d.docs-df.live.net/d66b9407fb17d322/Documents/COntainer%20Service/Docs.one)s

Marathon REST API

Once you have a tunnel open you can run commands on your local machine to interact with the [Marathon API](https://mesosphere.github.io/marathon/docs/generated/api.html). Of course you can also SSH into your management endpoint and work from, but generally we prefer to open a tunnel, it prevents inadvertently changing something on the master. It also means you can use your preferred OS as your client.

Once your tunnel is open you can make queries against the endpoint. For example to query what apps are currently deployed use:

`curl localhost:8080/v2/apps`

Against a new cluster this will return:

`{"apps":[]}`

Deploying a Docker Container

To deploy a Docker container using Marathon you first need to create a `marathon.json` file. For this tutorial we'll use one from a Demo application I have built. Copy <https://raw.githubusercontent.com/rgardler/AzureDevTestDeploy/master/marathon/marathon.json> to a local file ant then run:

`curl -X POST <http://master0:8080/v2/groups> -d @marathon.json -H "Content-type: application/json"`

This should return something like:

'{"version":"2015-11-20T18:59:00.494Z","deploymentId":"b12f8a73-f56a-4eb1-9375-4ac026d6cdec"}'

You can query the apps this deployment created:

```

curl localhost:8080/v2/apps

```

And visit the application we just deployed at your agents public URL.