# Server Status Tool

The tool is intended to show the status of various server processes that need to be brought up or down depending on the job being done.

These processes will be defined in a JSON file. It is very important to define the processes in the order in which they need to be brought down. So, a service that can come up or go down later needs to be defined after one that can be started earlier.

This tool is still in development so be a bit cautious in the order in which you change the state of the processes.

## Setup

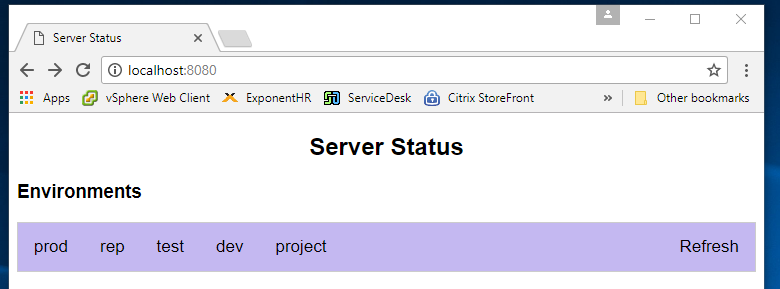
Unzip the file ServerStatus.zip to a directory somewhere on your hard disk.

Double click on the file ServerStatus.exe.

A command window should open. This window needs to remain open while you are using the application.

Give the application time to scan the JSON file and retrieve the various servers and services. This should take about a minute. Once done you will see a message on the screen that says “*Ready for web*”.

When you open a browser and enter in the URL <http://localhost:8080> you should see the main screen as well as the environments that are available to manage.



## Process

You can view the services available for each environment. Typically, that means Prod, Rep, Test, Dev & Project.

On the screen below you will see several different services. The services are color coded showing that they are UP, Down, or we were unable to connect to the Service or Server.

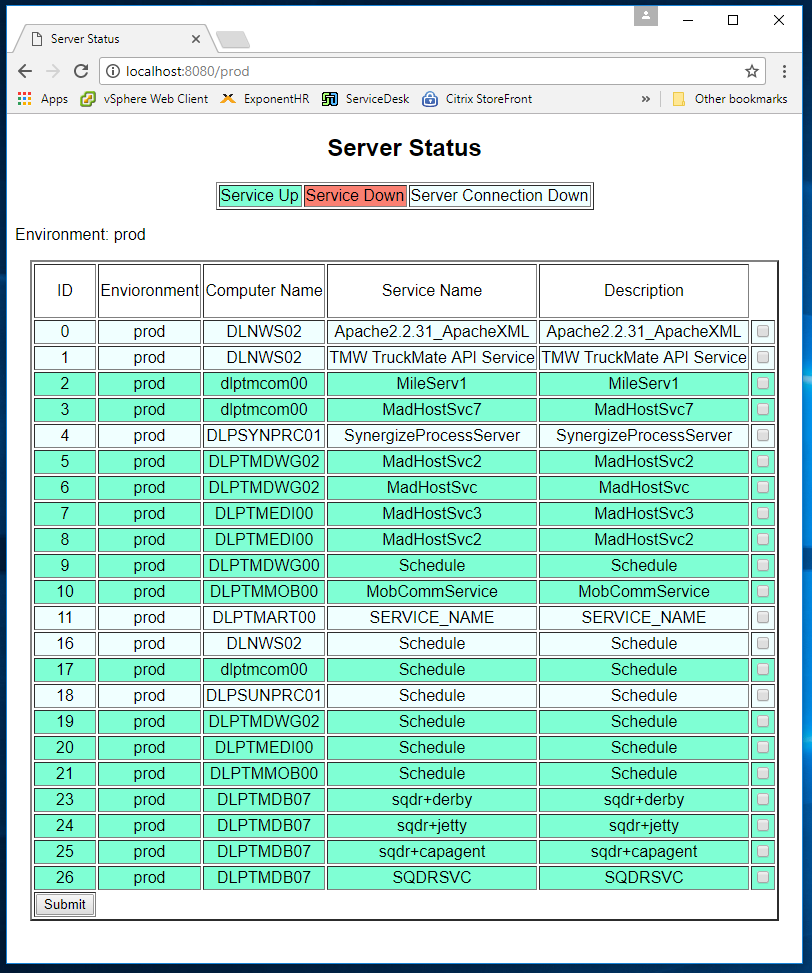
You can click on the checkbox for a process to bring it up or down. Presently you may only click on services that are in the same category, Up or Dn or Unknown.

***If you try to click on several objects that are in different categories the system will fail.***

Check the box(s) that you want to work with and click on Submit. The service will be either brought down or started up.

***If you click on a process that is unknown the application will try to bring it up. However, it will most likely be unsuccessful.***

Clicking on the “Refresh” button on the home page will re-query the states for each service listed in the JSON file. This could take a min or two.



## JSON file

The following is a code snippet from the JSON file:

|  |
| --- |
| [  {  "id": "0",  "env": "prod",  "ComputerName": "DLNWS02",  "Service": "Apache2.2.31\_ApacheXML",  "Description": " Apache2.2.31\_ApacheXML "  },  {  "id": "1",  "env": "prod",  "ComputerName": "DLNWS02",  "Service": "TMW TruckMate API Service",  "Description": " TMW TruckMate API Service "  }  ] |

***The “id” field must start with 0 and increment up one for each service.***

The computer name is just the Name. No need for anything else

The service is the name identifier from the services screen on windows. If you get a list of services and click on properties you will get the “Name”, which is different than the description field

## Logging

By default logging is done to a file in the home directory of the application. It is called “ServerStatus.log”.

You can turn on debug by opening the command window and entering the following initial command, “ServerStatus -d”. This will begin the application in debug mode which will provide a lot more internal tracing