Creating an Action Profile to Start Any Windows Service

Overview

This article describes how to create a single Action Profile that can start various Windows Services when Uptime Infrastructure Monitor observes that a Windows Service has stopped.

Background

Alerts and Actions can be triggered in Uptime Infrastructure Monitor when an Element changes state. This is discussed in the Uptime Infrastructure Monitor Administrator's Guide here: Alerts and Actions. Several actions are available out-of-the-box: Log to File, Recovery Script, Windows Service and SNMP trap.

The Windows Service action can Start, Stop or Restart a Windows Service identified by name. This is useful when you've created a Service Monitor using the Windows Service Check monitor to watch for a state change with a specific Windows Service. A typical scenario would be to set a Critical Status if a specified Windows Service is Stopped. In this case the Windows Service action in Uptime Infrastructure Monitor could be used to automatically try to start the Windows Service in an attempt to minimize service disruption.

The limitation to leveraging the Windows Service action on an Action Profile is that the Windows Service to be started must be identified by name. If you have a large number of Windows Services that could drive an action you will need to create a new Action Profile for each Windows Service and ensure you tie it to the correct Service Monitor. In environments where the number of Windows Services that fit this scenario is small this works well. However, in larger environments where the number of Windows Services that fit this scenario is large it might be desirable to create a single Action Profile that can restart any Windows Service.

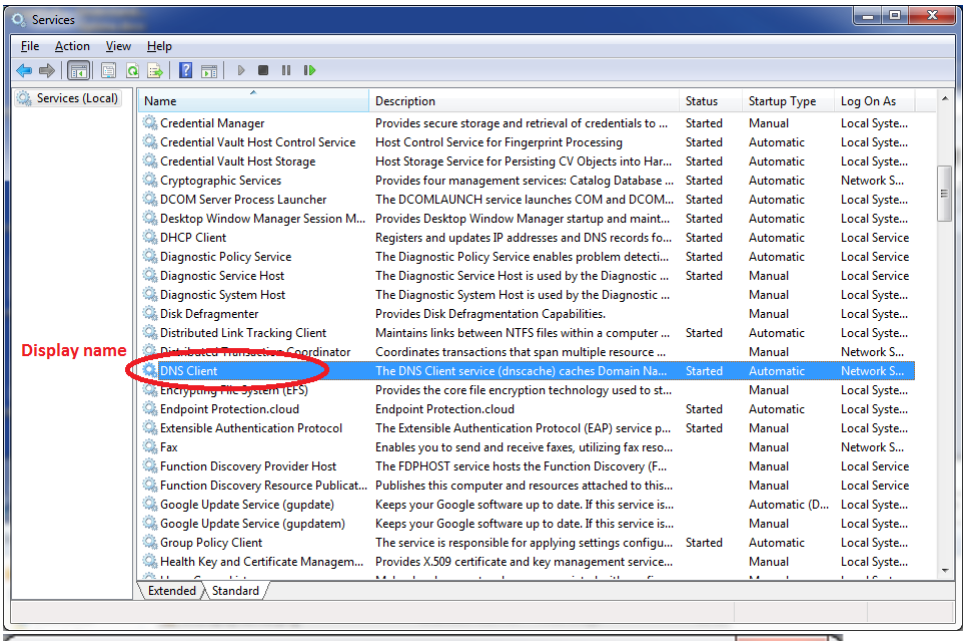
This article explores how to restart any Windows Service with an Action Profile as an alternative to creating a specific Action Profile for each Windows Service.

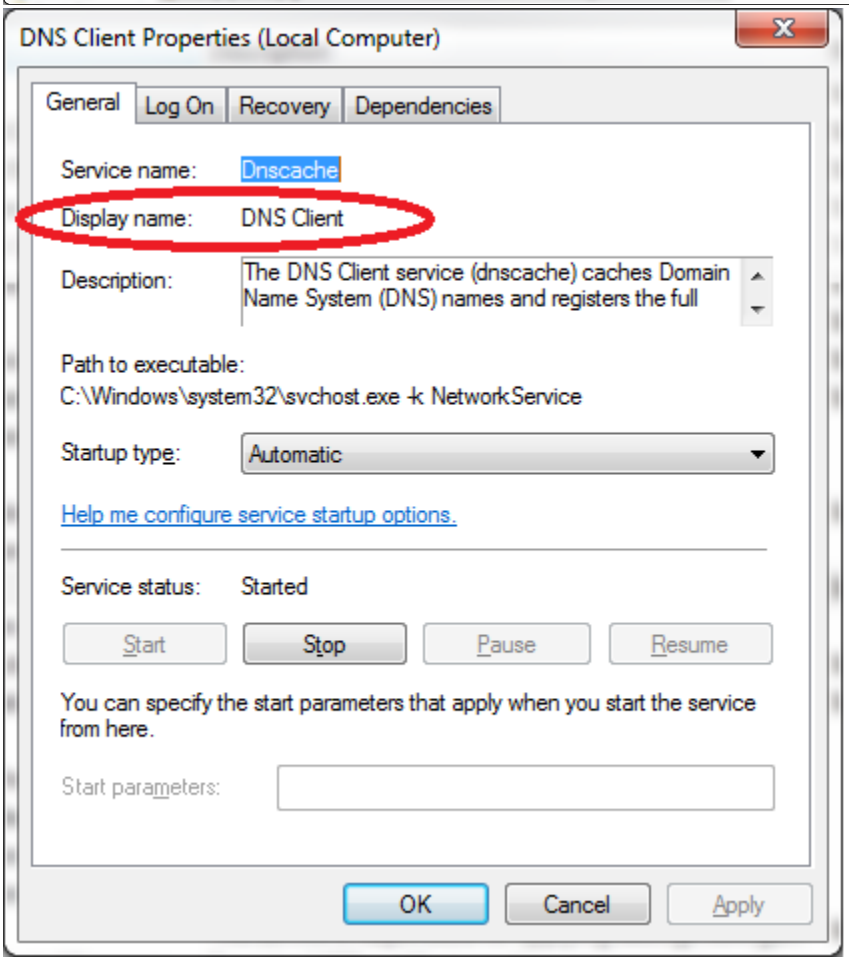
Create a Service Monitor

Start by creating a Service Monitor as follows:

- Create a Service Monitor selecting the Windows Service Check monitor.

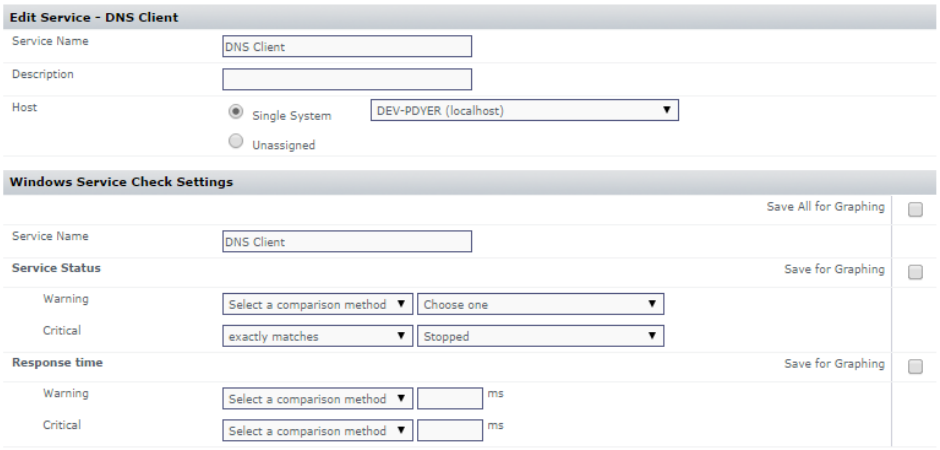
> IMPORTANT: You must match the Service Name to the Windows Service Display Name. This name will be leveraged later to select the Windows Service to start.
> Note that Microsoft has both a Display Name and a Service Name. We want the Display Name (see the images that follow).





- Complete the remainder of the Service Monitor making sure the Windows Service - Service Name is also the Display Name.
- Raise a Critical Alert when the Service Status is Stopped.

- Your Service Monitor should look something like the image that follows.



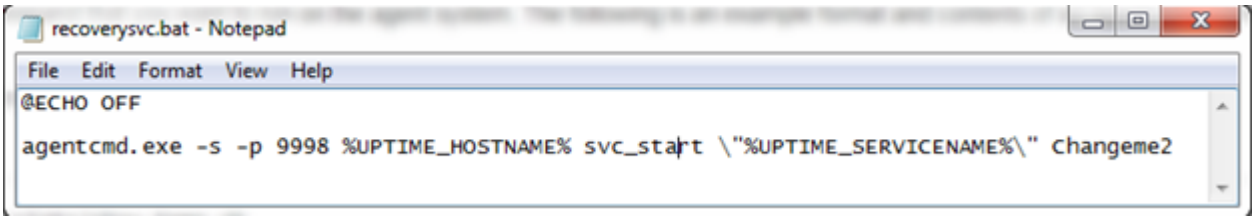
Create Recovery Script

We are going to leverage a Recovery Script in an Action Profile to Start the Windows Service. This will be a very simple script. We are going to leverage two key components built into Uptime Infrastructure Monitor:

- First, the ability to create a script on the Uptime Infrastructure Monitor Monitoring Station that directs an Uptime Infrastructure Monitor Agent to perform an action. In fact, the Uptime Infrastructure Monitor Agent exposes a method to the Uptime Infrastructure Monitor Monitoring Station to Start a Window Service – we simply have to supply the Windows Service name.
- Second, the use of Recovery Script Variables (variables available when creating Recovery Scripts). These are variables handed into our script by Uptime Infrastructure Monitor.

Here is what the Recovery Script looks like:

```
@ECHO OFF
agentcmd.exe -s -p 9998 %UPTIME_HOSTNAME% svc_start \"%UPTIME_SERVICENAME%\" Changeme2
```



Some items to note:

- %UPTIME_HOSTNAME% is the name of the system raising the Alert, also the system where the Windows Service will be restarted.
- %UPTIME_SERVICENAME% is the name of the Service Monitor we created earlier. Note that this is why we needed the Service Monitor name to match the Windows Service Display Name.
- The \" may look strange, this ensures the whole Windows Service name gets included regardless of spaces.
- Changeme2 is the Uptime Infrastructure Monitor Agent Password in this environment, your environment will differ.

This script is saved to the Scripts directory on the Uptime Infrastructure Monitor Monitoring Station (default: C:\Program Files\uptime software\uptime\scripts).

For the sake of brevity we will skip the remaining details. However, if you want more information you can find it online, here are two key articles:

Creating an Action Profile Recovery Script (for the purposes of this article focus on the Monitoring Station Script).

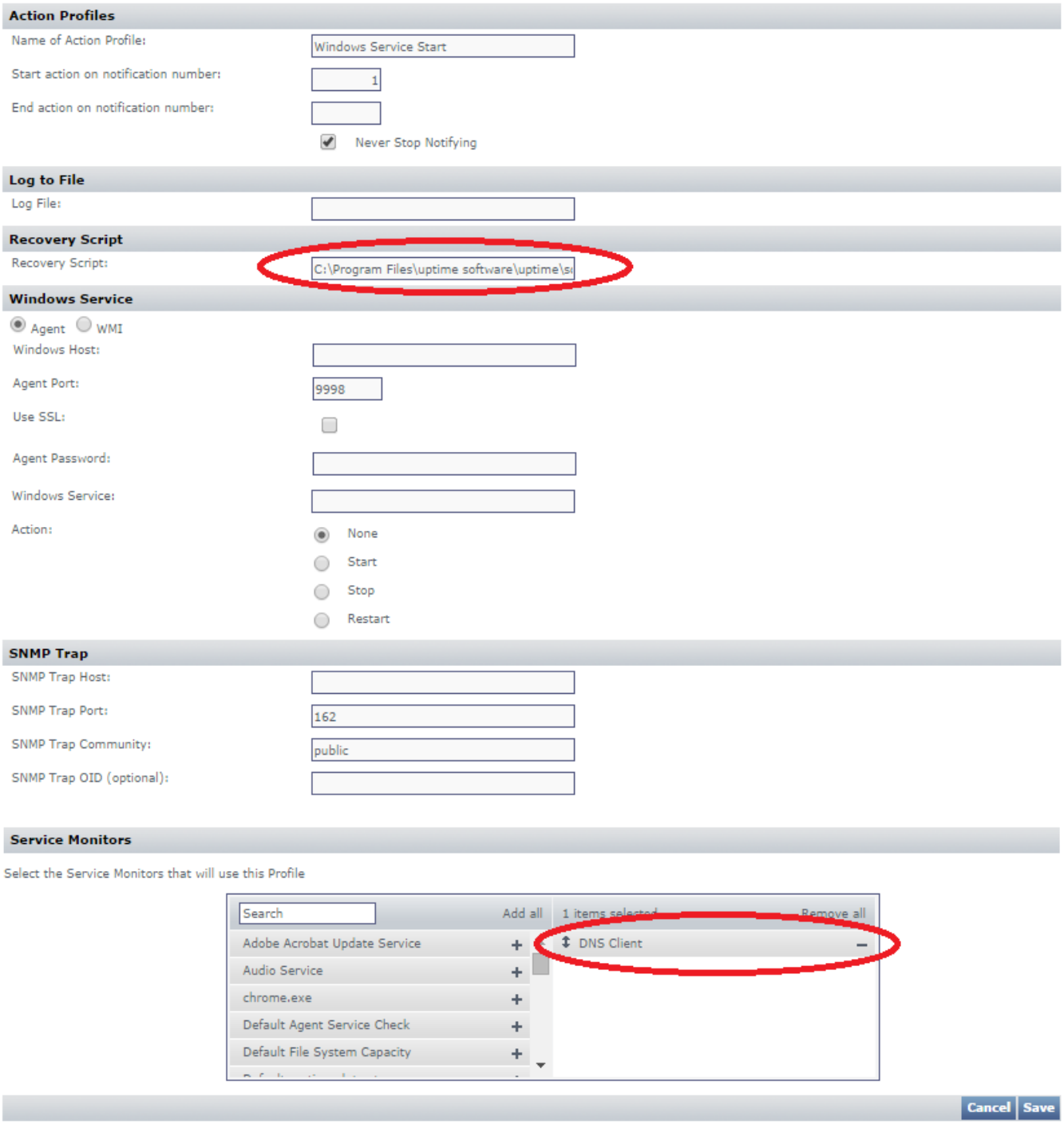
Alert Profile and Action Profile Variables (for the purposes of this article focus on the Recovery Script Variables):

Create an Action Profile

Lastly, we need to tie the Service Monitor and Recovery Script together in an Action Profile. Follow these steps:

- Create a new Action Profile and name it appropriately (no specific name is required).
- Populate the full path to the recovery script.
- Add the Service Monitor we created above.

It should look something like this:



Testing

The final test is to make sure that everything works as expected. Testing is as simple as stopping the Windows Service from your Service Monitor. You may want to adjust the Timing Settings on the Service Monitor to trigger the action sooner for the purposes of testing.

If you are familiar with Action Profiles you may know that Uptime Infrastructure Monitor has a built in Test Action Profile capability. However, that won't work in our case because the variables we are leveraging only get populated when a Service Monitor is triggered.

Notes

The Windows Service will be started in the security context of the Uptime Infrastructure Monitor agent running on the system where the Windows Service is being started (by default Local System).

This is only one way to accomplish this task, other viable options exist. You could create an Uptime Infrastructure Monitor Agent side script to perform the recovery, you could call a third party tool from the recovery script to start the service, etc, etc.