SPK
and Associates

www.spkaa.com
Ph: 888-310-4540

SPK and Associates
900 E Hamilton Ave, Ste.100
Campbell, CA 95008

Monitoring Web applications using WebInject

Nagios is a wonderful tool. However, there are limits to what you can do with it directly. As Brad discussed, using NRPE is a powerful tool for Windows. For complicated web checks that require multi-step operation, we use: WebInject.

WebInject enhances Nagios in multiple ways:

1) Creates stateful requests by defining variables
2) Allows multi-step processing
3) Provides both GET and POST
4) Supports unlimited customization in each step

Each step in the configuration is a “case” with a unique “id”. WebInject also allows for setting variables such as the “SSO_USER” and “SSO_PASS” variable.

Each webinject starts with a configuration file that ‘s pretty simple. You have to estimate the length of time in seconds. If this “globaltimeout” is reached, then the script will fail so you need to accommodate for busy times on the servers. The “reporttype” needs to be nagios in order for it to integrate. The “testcasefile” is the filename of the actual request:

```
<globaltimeout>120</globaltimeout>
<globalhttplog>onfail</globalhttplog>
<testcasefile>sso.xml</testcasefile>
<reporttype>nagios</reporttype>
```

Here’s the testcasefile contents. The following script is a multi-step login to a Oracle SSO server that tests getting content from the application server after a successful user/password authentication.

```
<testcases>
<!-- The production version of the sso.xml script -->
<testvar varname="SSO_USER">user1@company1.com</testvar>
<testvar varname="SSO_PASS">password</testvar>
<!-- testvar varname="SSO_DOMAIN">@DIVISION.COMPANY1CORP.COM</testvar-->
<case
   id="1"
   description1="Accessing SSO Login Page"
   method="get"
   url="http://app1.company1.com"
   parseresponse1="Site2pstoreToken=|\n"
   errormessage="Unable to access the SSO Login server"
/>
<case
```

```
    id="2"
    description2="Performing SSO Login Redirect"
    method="get"

url="https://ssoserv1.company1.com/sso/pages/login.jsp?site2pstoretoken={PARSEDRESULT1}&p_error_code=&p_submit_url=http%3A%2F%2Fssoserv1.company1.com%3A7777%2Fsso%2Fauth&p_cancel_url=http%3A%2F%2Fapp1.company1.com&ssousername=&subscribername="
    errormessage="Unable to perform a successful SSO Login redirect"
/>
<case
   id="3"
   description3="Performing Actual SSO Login"
   method="post"
   url="https://ssoserv1.company1.com/sso/auth"

postbody="v=v1.4&site2pstoretoken={PARSEDRESULT1}&locale=&ssousername=${SSO_USER}&userid=${SSO_USER}&password=${SSO_PASS}"
   errormessage="Unable to perform a successful SSO Login"
/>
<case
   id="4"
   description4="Parsing SSO Login Redirect"
   method="get"

url="https://ssoserv1.company1.com/pls/orasso/orasso.wwsso_app_admin.ls_login?Site2pstoreToken={PARSEDRESULT1}"
   errormessage="Failure to Parse SSO Login Redirect"
   parseresponse2="\<TITLE>Redirect to |\</TITLE>"
/>
<case
   id="5"
   description5="Following Login Redirect"
   method="get"
   url="{PARSEDRESULT2}"
   errormessage="Failure to follow Login Redirect"
/>
<case
   id="6"
   description6="Redirect to Content - step 1"
   method="get"
   url="http://app1.company1.com/wpslaunch_1.html"
   errormessage="Failure to Redirect to Content - step 1"
/>
<case
   id="7"
   description7="Redirect to Content step - 2"
   method="get"
   url="http://app1.company1.com/wps/myportal/"
   errormessage="Failure to Redirect to Content step - 2"
```

```
/>
<case
  id="9"
  description9="Logout"
  method="get"
  url="http://app1.company1.com/osso_logout"
  errrormessage="Failure to perform logout"
/>
</testcases>
```

Tying into Nagios, is as simple as adding this to the services.cfg:

```
define service{
      use                     generic-service
      host_name                   webhost
      service_description           SSO Login Check
      is_volatile              0
      check_period                24x7
      max_check_attempts               3
      normal_check_interval           3
      retry_check_interval          1
      contact_groups               admins, admin_pagers
      notification_interval        15
      notification_period          24x7
      notification_options         w,u,c,r
      check_command                    webinject!config.xml!sso.xml
      }
```