Happy New Year! Here is a little tip on how we used the check_http and check_bigip_pool to eliminate our need for WebTrends. The servers being monitored are in the DMZ and the Nagios host is inside our firewall. I have deployed the NRPE daemon to our Solaris 9 servers, so we will also need to modify the nrpe.cfg file. The check_bigip_pool is run against our F5 appliance, which we use for load balancing. The check_bigip_pool command is run from the Nagios host against the F5. The following information is step-by-step on how I set these hosts and services up.

Using check_http is very straight forward. I would suggest that you concentrate on receiving a string or file size in addition to the straight HTTP service check. You also want to check the results by running the command form the command line.

Using check_bigip_pool is also straight forward. You will need a hostname (hostname for F5), community (generally public), software version and pool name. The check_bigip_pool will return the number of nodes found running on the F5. So, you will need to work with the warning and critical percentages. HINT: If check_bigip_pool finds one of two nodes running and you only require one node running, set the warning to 49 and critical to 25. When both nodes are down you will receive a critical alert and notification.

Here are the configuration steps for this setup:

Define what is to be monitored (NOTE: this is just an outline, my organizational setup monitors 5 clients and 7 different ports):

Client	URL	Port	HTTP Response
webstore.<your domain>.com	http://webstore	80	OK
webstore.<your domain>.com	http://webstore.com/WatchDog.tem	80	OK – I am here
f5host.<your domain>.com	your pool name		count of nodes up

**Service definition are placed within a template file. I prefer to place service definitions within a file that is descriptive of the function. You will also need to create a separate service definition for each service check, because you need to create distinct service_descriptions section. You can leave blanks in the service description; however, I do not because it will remove your ability to acknowledge problems via email. I also used the "hosts" flag instead of hostgroups, because I have 4 webstore servers, but not all of them are looking at the same URLs and ports. Also include a notes section, we will be using it later for our notification email.*

Service Definition (nagios_host:/path-to-nagios/etc/objects/services/srv_check_http.cfg):

```
define service{
        use                     generic-service
        hosts                   hostname
        notes                   description of the service being monitored.
        service_description     check_webstore
        check_command           check_nrpe!check_webstore
        max_check_attempts       3
        normal_check_interval    5
        retry_check_interval     1
        check_period             24x7
        contact_groups           webstore_admins
        }
```

**Notice the variation in the check_command definition? This is so you can run the command on the Nagios host and be listed under the defined host.*

F5 Service Definition (nagios_host:/path-to-nagios/etc/objects/services/f5_checks.cfg):

```
define service{
        use                     generic-service
        hosts                   f5host
        service_description      webstore_business_station
        notes                   There are 0 nodes online for http://business-station.coat.com.
        check_command            check_webstore_business_station_pool!
        max_check_attempts        3
        normal_check_interval     5
        retry_check_interval      1
        notification_period        24x7
        check_period               24x7
        contact_groups             webstore_admins
        }
```

**Host definition is placed within a template. I have created a separate template for each classification of hosts, which makes it easier to find and update hosts. The host definition would be the same for the f5 server.*

Host Definition (nagios_host:/path_to_nagios/etc/objects/hosts/webstore_servers.cfg):

```
define host{
        use                     webstore-servers   ; Name of host template to use
        host_name               your host's name
        address                 ip_address
        alias                   short name for your host
        notes                   <b>What does the host do<br>LOCATION:         <br>CONSOLE:
</b><br>
        icon_image              sun_logo.gif
        icon_image_alt          Sun Host
        hostgroups              Add your host groups for standard monitoring and specialty monitoring
        notification_period     24x7
        }
```

**Here is how I defined the commands. For separation purposes, I prefer to the definitions within a distinct configuration file and give each command name a distinct definition. This way if one port is not accessible, you can address the one port issue and not assume the entire web application is down.*

Command Definition (nagios_host:/path_to_nagios/etc/objects/commands/cmd_http.cfg):

```
define command{
        command_name            check_webstore
        command_line            $USER1$/check_http  -H $ARG1$ -u $ARG2$ -p $ARG3$
        }
```

F5 Command Definition (nagios_host:/path_to_nagios/etc/objects/commands/cmd_check_bigip_pool.cfg):

```
define command{
        command_name            check_webstore_pool
        command_line            $USER1$/check_bigip_pool -H $HOSTADDRESS$ -C $ARG1$ -S
$ARG2$ -P $ARG3$ -w $ARG4$ -c $ARG5$
        }
```

**I combine the contactgroup with the contact definition in its own file. You may not have the luxury of email aliases, so this helps me manage the contacts for a specific host or service.*

Contact Definition (nagios_host:/path_to_nagios/etc/objects/contacts/webstore_admins.cfg):

```
define contactgroup{
        contactgroup_name               webstore_admins
        alias                           WEBSTORE ADMINS
        members                         webstore_admins
        }
define contact{
        contact_name                    webstore_admins
        alias                           WEBSTORE Admins
        contact_groups                  webstore_admins
        host_notifications_enabled      0
        service_notifications_enabled   1
        service_notification_period     24x7
        host_notification_period        24x7
        service_notification_options    w,u,c,r
        host_notification_options       d,r
        service_notification_commands   notify-webstore-service-by-email
        host_notification_commands       notify-linux-host-by-email
        email                           webstore_support@your_domain.com
        can_submit_commands             1
        }
```

**Notifications are fun. I keep separate command files, because different groups need to receive different information. Notice that the service_notification_commands in the contacts matches the notification definition command name. Within the definition, you will find the notes section after the "-s" flag and also the method I use for enabling problem acknowledgments via a link within email.*

Notification Definition (nagios_host:/path_to_nagios/etc/objects/commands/webstore_admins_email_notification.cfg):

```
# Linux 'notify-webstore-service-by-email' command definition
define command{
        command_name    notify-webstore-service-by-email
        command_line    /usr/bin/printf "%b" "If this is a CRITICAL problem, please insure the
ecommerce on-call person is
notified.\n\nHostname:\t\t$HOSTNAME$\nService:\t\t$SERVICEDESC$\nState:\t\t\t$SERVICESTAT
E$\nDate/Time:\t\t$LONGDATETIME$\nAdditional
Info:\t$SERVICEOUTPUT$\nAcknowledge:\t\thttp://nagios_host.your_domain.com/nagios/cgi-
bin/cmd.cgi?
cmd_typ=34&host=$HOSTNAME$&service=$SERVICEDESC$\nAuthor:\t\t$SERVICEACKAUTH
OR$\nAcknowledgement:\t$SERVICEACKCOMMENT$" | /usr/bin/mailx -s "$HOSTALIAS$:
$SERVICENOTES$" $CONTACTEMAIL$ -r core.admin@coat.com
    }
```

Email Message:

Subject: <hostname alias>:<service notes section>.

Body:

```
If this is a CRITICAL problem, please insure the ecommerce on-call person is
notified.

Hostname:               webstore1.coat.com
Service:                check_webstore_8150
State:                  CRITICAL
Date/Time:              Wed Jan 6 02:24:39 EST 2010
Additional Info:        CRITICAL - Socket timeout after 10 seconds
Acknowledge:            http://nagios_host.your_domain.com/nagios/cgi-bin/cmd.cgi?
cmd_typ=34&host=<host_name>$service=service_name
Author:
Acknowledgement:
```

Hopefully this helps with your monitoring of web sites. Any questions please feel free to email me at mikhail@ebusinessjuncture.com.