# TBSB Hawk Setup in PPE / Production

**New Files:**

runSecondaryEMSHealthCheck.sh.sh – copy of runEMSHealthCheck, control script for Secondary EMS Health Checker

hawktolog.sh - script to direct hawk alerts to a log file. This log file is monitored by NSM. *Path: /tbsb/scripts*

hawkvariables.txt - defines global variables in use by Hawk Rulebases. *Path: /tbsb/tibco/tra/domain/TBSB/*

TBSB\_BW\_TBSBBW01\_PPE\_v01.hrb – rule base for ukirsbl4.

TBSB\_BW\_TBSBBW02\_PPE\_v01.hrb – rule base for ukirsbl5.

TBSB\_BW\_TBSBBW51\_PPE\_v01.hrb – rule base for ukirsbl6.

TBSB\_BW\_TBSBBW52\_PPE\_v01.hrb – rule base for ukirsbl7.

TBSB\_BW\_TBSBBW01\_PRD\_v01.hrb – rule base for ukirpbl4.

TBSB\_BW\_TBSBBW02\_PRD\_v01.hrb – rule base for ukirpbl5.

TBSB\_BW\_TBSBBW51\_PRD\_v01.hrb – rule base for ukirpbl6.

TBSB\_BW\_TBSBBW52\_PRD\_v01.hrb – rule base for ukirpbl7.

TBSB\_ALL\_EMS\_v05.hrb – rule base for all EMS Brokers and EMS Servers in PPE and Production.

TBSB\_RDC\_EMS\_v02.hrb – rule base for RDC only EMS Brokers and EMS Servers in PPE and Production.

TBSB\_CORE\_v04.hrb – rule base for all TBSB Hosts

TBSB\_ADMIN\_v04.hrb – rule base for all TBSB Administrators

TBSB\_POLAGT\_v04.hrb – rule base for all TBSB Policy Agents

**Required Updates:**

hawkagent.cfg – uncomment –variables option. Specifiy path to hawkvariables.txt. Path /tbsb/tibco/tra/domain/TBSB

hawkagent\_TBSB.log – log file to which hawk alerts are sent. Monitored by NSM. Path /tbsb/logs

tibjmsadmin.hma – -server option should be tcp://localhost:8888. Path: /tbsb/tibco/tra/domain/TBSB/plugin

runEMSHealthCheck.sh – change return status of a status query from ‘2’ to ‘0’.

## 1.0 Business Works Server Rule Base

**1.1 Process – Business Works Engine**

Test: Process Count of process *bwengine* = 0

Actions If True:

Create High Alert: ‘*Process Count’* instances of *‘Process Name’* running on ‘*Hostname’* since ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’*  Hawk: BW Process Engine Process bwengine is not running

Actions when cleared:

Write to log file: *‘Date , Timestamp’*  Hawk: 1 or more BW Process Engine Processes are running

**1.2 Process – Security Engine**

Test: Process Count of process *java* = 0

Actions If True:

Create High Alert: ‘*Process Count’* instances of *‘Process Name’* running on ‘*Hostname’* since ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’*  Hawk: Security Engine Process is not running

Actions when cleared:

Write to log file: *‘Date , Timestamp’*  Security Engine Process is running

**1.3 BW MicroAgent – GetExecInfo**

Test: Status of the BW Engine != ACTIVE

Actions If True:

Create High Alert: TBSB Process Engine on ‘*Hostname’* not ACTIVEsince ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’*  Hawk: process engine is not active

Actions when cleared:

Write to log file: *‘Date , Timestamp’*  Hawk: process engine is active

**1.4 BW MicroAgent – getHostInformation**

Test: State of TBSB Application != RUNNING

Actions If True:

Create High Alert: TBSB Application on ‘*Hostname’* not in RUNNING statesince ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’*  Hawk: TBSB-Archive application instance is not running

Actions when cleared:

Write to log file: *‘Date , Timestamp’*  Hawk: TBSB-Archive application instance is running

**1.5 BW MicroAgent – GetMemoryUsage**

Test: GetMemoryUsage() method returns >= 85%

Actions If True:

Create High Alert: TBSB Process Engine on ‘*Hostname’* memory usage is >=85% since ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’*  Hawk: BW Engine memory usage is over 85%

Actions when cleared:

Write to log file: *‘Date , Timestamp’*  Hawk: BW Engine memory usage is below 85%

**1.6 BW MicroAgent – getStatus**

Test: GetStatus() method returns Total Errors != 0

Actions If True:

Create High Alert: TBSB Business Works Engine on ‘*Hostname’* reporting 1 or more ERRORSsince ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’*  Hawk: BW Engine is reporting 1 or more Errors

Actions when cleared:

Write to log file: *‘Date , Timestamp’*  Hawk: BW Engine is reporting 0 errors

**1.7 BW MicroAgent - GetProcessStarters**

Test: GetStatus() method returns Total Errors != 0

Actions If True:

Create High Alert: TBSB Business Works Engine on ‘*Hostname’* reporting 1 or more ERRORSsince ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’*  Hawk: A Process within the TBSB-Archive application instance is not active. Process: *ProcessName*

Actions when cleared:

Write to log file: *‘Date , Timestamp’*  Hawk: The Process within the TBSB-Archive application is now active. Process: *ProcessName*

## 2.0 EMS Rule Base

**2.1 Process – EMS Daemon**

Test: Process Count of process *tibemsd64* = 0

Actions If True:

Create High Alert: EMS Server: ‘*Process Count’* instances of *‘Process Name’* running on ‘*Hostname’* since ‘*Timestamp’*

Write to log file*: ‘Date , Timestamp’* Hawk: The TIBCO EMS Daemon process is not running

Actions when cleared:

Write to log file:  *‘Date , Timestamp’* Hawk: The TIBCO EMS Daemon process is running

**2.2 Process – EMS Health Check**

Test: Process Count of command *java -Dlog4j.configuration=file:log4j.xml -jar start.jar OPTIONS=jndi jetty.port=8090* = 0

Actions If True:

Create High Alert: Primary EMS Healthcheck: ‘Process Count’ instances of ‘Process Name’ running on ‘Hostname’ since ‘Timestamp’

Write to log file: *‘Date , Timestamp’* Hawk: A required instance of EMS Health Checker process is not running. Currently Running: *‘Process Count’*

Actions when cleared:

Write to log file: *‘Date , Timestamp’* Hawk: The required number of EMS Health Checker processes are running. Currently Running: *‘Process Count’*

**2.3 Microagent – JMS\_controller**

Test: isRunning() method != true

Actions If True:

Create High Alert: EMS Server status on ‘*Hostname’ not running* since ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’*  Hawk: EMS Server is not in running state"

Actions when cleared:

Write to log file: *‘Date , Timestamp’*  Hawk: EMS Server is running

**2.4 Microagent – JMS\_controller**

Test: if the getQueues method returns a queue(s) with >= 100 pending messages

Actions If True:

Create High Alert: EMS Server *‘Hostname’* has *‘Pending Message Count’* pending messages on *‘QueueName’* since ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’*  Hawk: EMS Server has more than 100 pending messages of *‘Pending Message Count’ ‘Queue name’*

Actions when cleared:

Write to log file: *‘Date , Timestamp’*  Hawk: EMS Server has less than 100 pending messages" on *‘Queue name’*

**2.5 Microagent – JMS\_controller**

Test: if the getTopics method returns a topic(s) with >= 100 pending messages

Actions If True:

Create High Alert: EMS Server *‘Hostname’* has *‘Pending Message Count’* pending messages on *‘TopicName’* since ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’*  Hawk: EMS Server has more than 100 pending messages of *‘Pending Message Count’ ‘Topic name’*

Actions when cleared:

Write to log file: *‘Date , Timestamp’*  Hawk: EMS Server has less than 100 pending messages" on *‘Topic name’*

**2.6 Microagent – Custom Script – EMS Control Script**

Test1 : if the EMS Control Script ‘status’ method does not return the string *“STATUS: ems\_TBSB running”*

Actions If True:

Create High Alert: EMS Daemon Status script on *‘Hostname’* is returning *‘Return String’* since ‘*Timestamp’*

Test2: if the EMS Control Script ‘status’ method returns the string *“STATUS: ems\_TBSB not running”*

Actions If True:

Create Notification: Attempting to start EMS Daemon on *‘Hostname’ at* ‘*Timestamp’*

Attempt to start the EMS Daemon: Execute: *‘EMS Control Script’* start (note this will be attempted 3 times, not sooner than after 30seconds)

**2.7 Microagent – Custom Script – Primary EMS Health Check Control Script**

Test1 : if the EMS Health Checker Control Script ‘status’ method does not return the string *“EmsHealthCheck is running…”*

Actions If True:

Create High Alert: Primary EMS Health Check: EMS HealthCheck Status script on *‘Hostname’* is returning *‘Return String’* since ‘*Timestamp’*

Test2: if the EMS Control Script ‘status’ method returns the string *“EmsHealthCheck is not running…””*

Actions If True:

Create Notification: Attempting to start Primary EMS Healthcheck on *‘Hostname’ at* ‘*Timestamp’*

Attempt to start the EMS Health Check: Execute: *‘EMS Control Script’* start (note this will be attempted 3 times, not sooner than after 30seconds)

**2.8 Microagent – Logfile – EMS Health Check**

Test1 : if the EMS Health Checker log file does not return the string *“Successfully tested JMS”*

Actions If True:

Create Medium Alert: Primary EMS Health Check Log File on *‘Hostname’* is not reporting success since ‘*Timestamp’.*

## 3.0 EMS Rule Base – RDC Only

**3.1 Process – EMS Health Check**

Test: Process Count of command *java -Dlog4j.configuration=file:log4j.xml -jar start.jar OPTIONS=jndi jetty.port=8091* = 0

Actions If True:

Create High Alert: Secondary EMS Healthcheck: ‘Process Count’ instances of ‘Process Name’ running on ‘Hostname’ since ‘Timestamp’

Write to log file: *‘Date , Timestamp’* Hawk: A required instance of EMS Health Checker process is not running. Currently Running: *‘Process Count’*

Actions when cleared:

Write to log file: *‘Date , Timestamp’* Hawk: The required number of EMS Health Checker processes are running. Currently Running: *‘Process Count’*

**3.2 Microagent – Custom Script – Secondary EMS Health Check Control Script**

Test1 : if the Secondary EMS Health Checker Control Script ‘status’ method does not return the string *“EmsHealthCheck is running…”*

Actions If True:

Create High Alert: Secondary EMS Health Check: Secondary EMS HealthCheck Status script on *‘Hostname’* is returning *‘Return String’* since ‘*Timestamp’*

Test2: if the EMS Control Script ‘status’ method returns the string *“EmsHealthCheck is not running…””*

Actions If True:

Create Notification: Attempting to start Secondary EMS Healthcheck on *‘Hostname’ at* ‘*Timestamp’*

Attempt to start the Secondary EMS Health Check: Execute: *‘EMS Control Script’* start (note this will be attempted 3 times, not sooner than after 30seconds)

## 4.0 TBSB Core Rule Base

**4.1 System – CPU Utilization**

Test: getCpuInfo() method returns % Time Idle <=15 %

Actions If True:

Create Low Alert: CPU on ‘*Hostname’* running at >= 85% since ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’* Hawk: CPU utilization at >= 85%

Actions when cleared:

Write to log file: *‘Date , Timestamp’* Hawk: CPU utilization < 85%

**4.2 FileSystem – Mount Point free space**

Test: getByMountPoint method returns % Free <= 15% for any mount point

Actions If True:

Create Low Alert: Disk space on *‘Hostname’ ‘MountPoint’ ‘%Free’* free since ‘*Timestamp’*

Write to log file: *‘Date , Timestamp’* Hawk: %Free space on mount point *‘mountpoint’ ‘%free’*

Actions when cleared:

Write to log file: *‘Date , Timestamp’* Hawk: %Free space on mount point *‘mountpoint’ ‘%free’*

## 5.0 TBSB Policy Agent Rule Base

**5.1 Process – Policy Agent**

Test: Process Count of process startProxyAgent != 2

Actions If True:

Create High Alert: Policy Agent: *‘Process Count’* instances of *‘Process Name’* running on *‘Hostname’* since *‘Timestamp’*

Write to log file: *‘Date , Timestamp’* Hawk: A required instance of Policy Agent processes is not running. Currently Running: *‘Process Count’*

Actions when cleared:

Write to log file: *‘Date , Timestamp’* Hawk: The required number of Policy Agent processes are running. Currently Running: *‘Process Count’*

## 6.0 TIBCO Administrator Rule Base

**6.1 Process – TIBCO Administrator**

Test: Process Count of process startProxyAgent != 2

Actions If True:

Create Low Alert: Administrator: *‘Process Count’* instances of *‘Process Name’* running on *‘Hostname’* since *‘Timestamp’*

Write to log file: *‘Date , Timestamp’* Hawk: The TIBCO Administrator process is not running

Actions when cleared:

Write to log file: *‘Date , Timestamp’* Hawk: The TIBCO Administrator process is running