

Configure your Observer properties and actions in ServiceNow

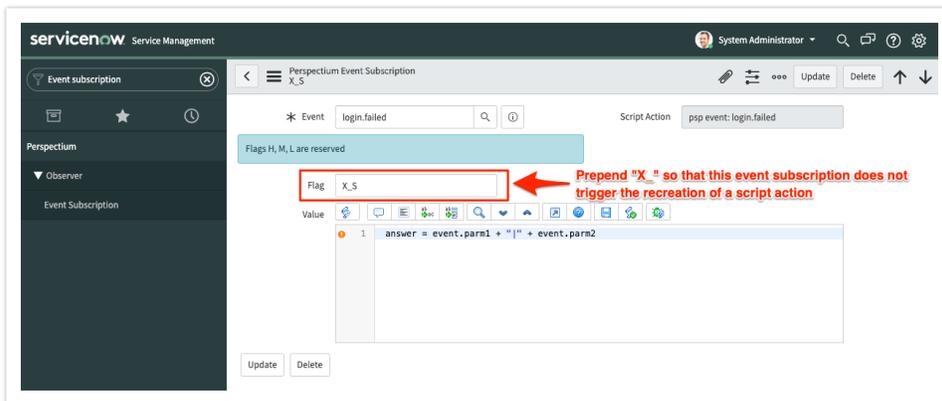
This page outlines instructions to configure your Observer properties, as well as actions, in ServiceNow. Observer actions are a way to track any number of metrics on your ServiceNow instance. These metrics are packed into messages and sent through MBS that you can access through your Observer portal.

Configure Observer properties

1. Log into your ServiceNow instance and navigate to **Perspectium > Observer > Properties**.
2. Select **Enable all script actions when running Start All Jobs** in order to enable all script actions when running **Start All Jobs** for Observer. If not selected, then any script actions that are disabled will not be automatically restarted when Start All Jobs is ran.
3. Enable/disable other properties as desired. Here's a description of available properties:

Property	Description
Enable a time period daily when Observer will not send data	Only allows data to be sent over to your Observer at a specified time.
Enable limiting the maximum number of Observer messages in the Outbound Messages	Limit the number of Observer messages in your outbound message with a specified number.
Include usernames/email addresses in top ten health check results	Allow user names and email addresses to be included in data sent to the Integration Mesh for displaying in the Observer Health Check. Since the Health Check queries for metrics from your instance like top ten SQL queries ran, these queries can include your username or email address in there and will be sent to the Mesh if you select this feature. Otherwise these metrics will be sent with a generic metric name such as Slow SQL query 3 if this feature is not selected.

4. If you want certain script actions to not be recreated when running **Start All Jobs**, you can prepend "X_" to the tag of the corresponding Event Subscription



Modify Observer actions

To make changes to any Observer actions, follow these steps:

1. Log into your ServiceNow instance and navigate to **Perspectium > Observer > Actions**.

- Click the name of the Observer action you want to modify.
- Actions will only execute if their active flag is set to true. You may de-activate the metrics which you do not care about without impacting any of the other metrics. Select or deselect the Active checkbox to make this change.

NOTE: [Start All Jobs](#) will re-activate the scheduled jobs responsible for processing these actions, but it will NOT re-activate any of your de-activated actions. The same logic applies for the [Stop All Jobs](#).

Find below a breakdown of all available Observer actions:

Tracking Tasks

Action	Description	Purpose	Interval
Active Requests with Closed Items	Will poll the request table for those which have requested items which are no longer active.	Keep track of requests which are no longer valid/necessary.	Hourly
Caller is deactivated on incidents	Will poll the incident table for active records whose caller is no longer active	To track the count of incidents where the caller is no longer valid	Hourly
Approval on Closed Tasks	Will poll the Approval table for approval requests on an inactive task	Keep track of approvals which were not completed in time or properly.	Hourly
Approver is deactivated	Will poll the Approval table for approval requests on where the assigned approver is inactive	Keep track of approvals which were not assigned properly.	Hourly
Open Tasks with Assigned to Deactivated	Will poll the Task table for those which are assigned to an inactive user	To track when tasks are improperly assigned. Can also be easily modified to track the count of unassigned tasks.	Hourly

Users and Access

Action	Description	Purpose	Interval
Local User Accounts	Will poll the User table (sys_user) for a count of local users, by keying off the "source" column.	Track the growth of local users on an instance.	Daily
External User Accounts	Will poll the User table (sys_user) for a count of external users, by keying off the "source" column.	Track the growth of external users on an instance.	Daily
Inactive Users	Will poll the User table (sys_user) for a count of user's who have not logged in the last 60 days	Track user access/retention on an instance.	Daily
Aggregate Active User Roles	Will aggregate the data for the count of user's who have been granted each role (sys_user_has_role)	To track how much access has been granted to certain areas of the instance.	Daily
Application Access Count	Will aggregate the data for the count of user's who have accessed certain applications, per application. Done through polling the app usage table (ua_app_usage)	To track the traffic of each application	On Interval

Activity and Response Metrics

Action	Description	Purpose	Interval
Import Set Queue	Over the last 60 seconds how many import sets (sys_import_set) are in each state (processed/loading/loaded)	Track how "backed up" the instance is in processing its import sets.	On Interval
Import Set Run Queue	Over the last 60 seconds track the counts of all the states of the current Import Set Runs (sys_import_set_runs)	Track the current processing of the Import Sets.	On Interval
Customer Updates	Over the last 60 seconds how many customer updates were made (those which are tracked in sys_update_xml)	Track how much work is being performed on the instance.	On Interval
Email Queue	Over the last 60 seconds track how many emails (sys_email) are in each state are in the instance.	Track how many emails you are generating compared to how quickly you are sending them out.	On Interval
Observe Nodes	Track the computing/sql metrics of each node (as well as averaged) on the instance	Track the CPU usage, response metrics, SQL response metrics, available semaphores, worker threads, etc. Type " stats.do " in the filter navigator for an example.	On Interval

Other

Action	Description	Purpose
--------	-------------	---------

Run on Login Failed	Ran when the event "login.failure" is fired	Track login failures on an instance as well as the common users which see this.
Daily Health Check	Ran nightly to capture a common set of data.	You can read more about the metrics here .

Next, you may want to customize the default settings in Observer.