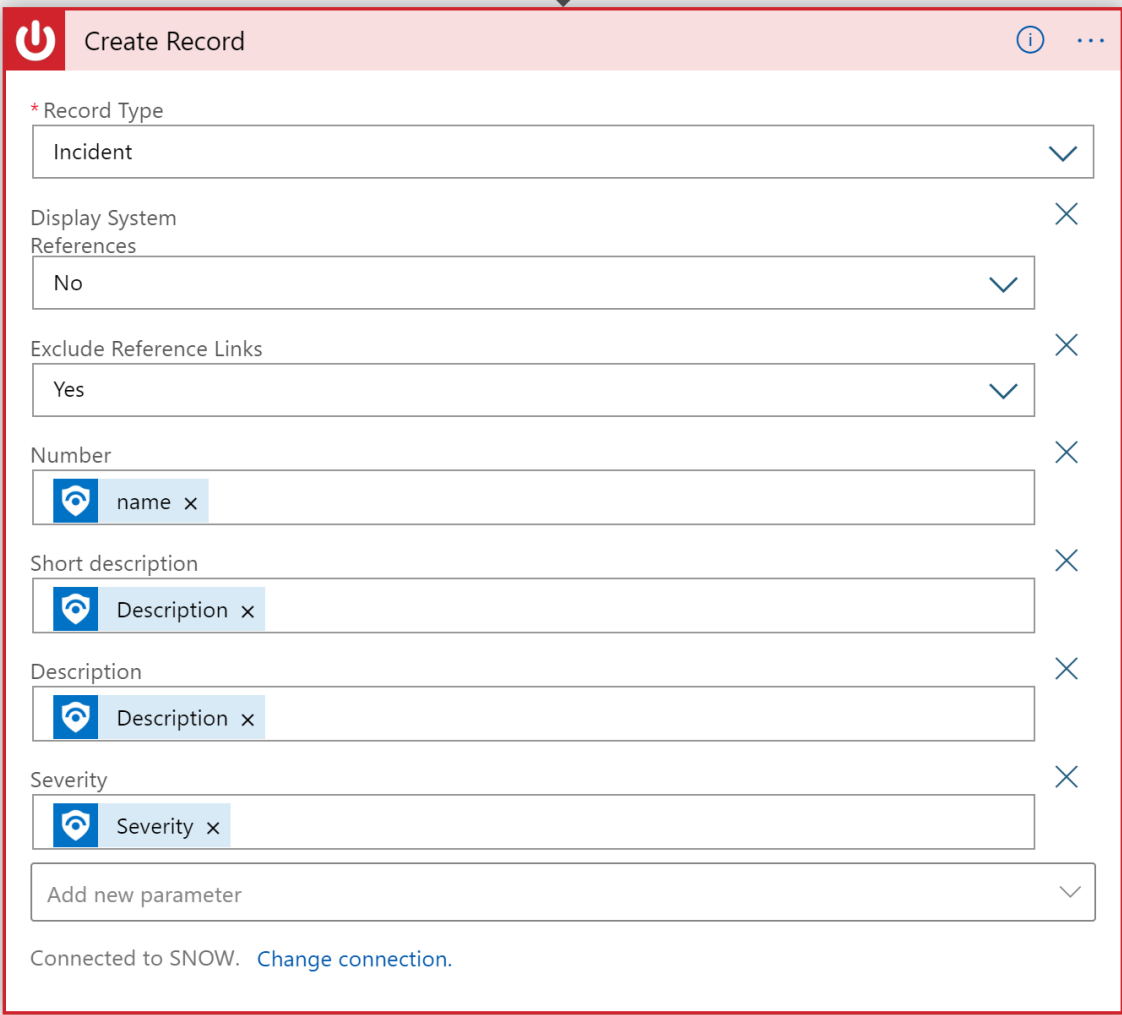


Sentinel to SNOW bi-directional sync how to:

1. Send Sentinel Incident to Service now.

Please keep this properties mapping when sending the incident data into SNOW



The screenshot shows the 'Create Record' form in ServiceNow. The form is titled 'Create Record' and has a red header bar with a power icon on the left and an information icon on the right. The form contains several fields with dropdown menus and a 'name' field. The fields are:

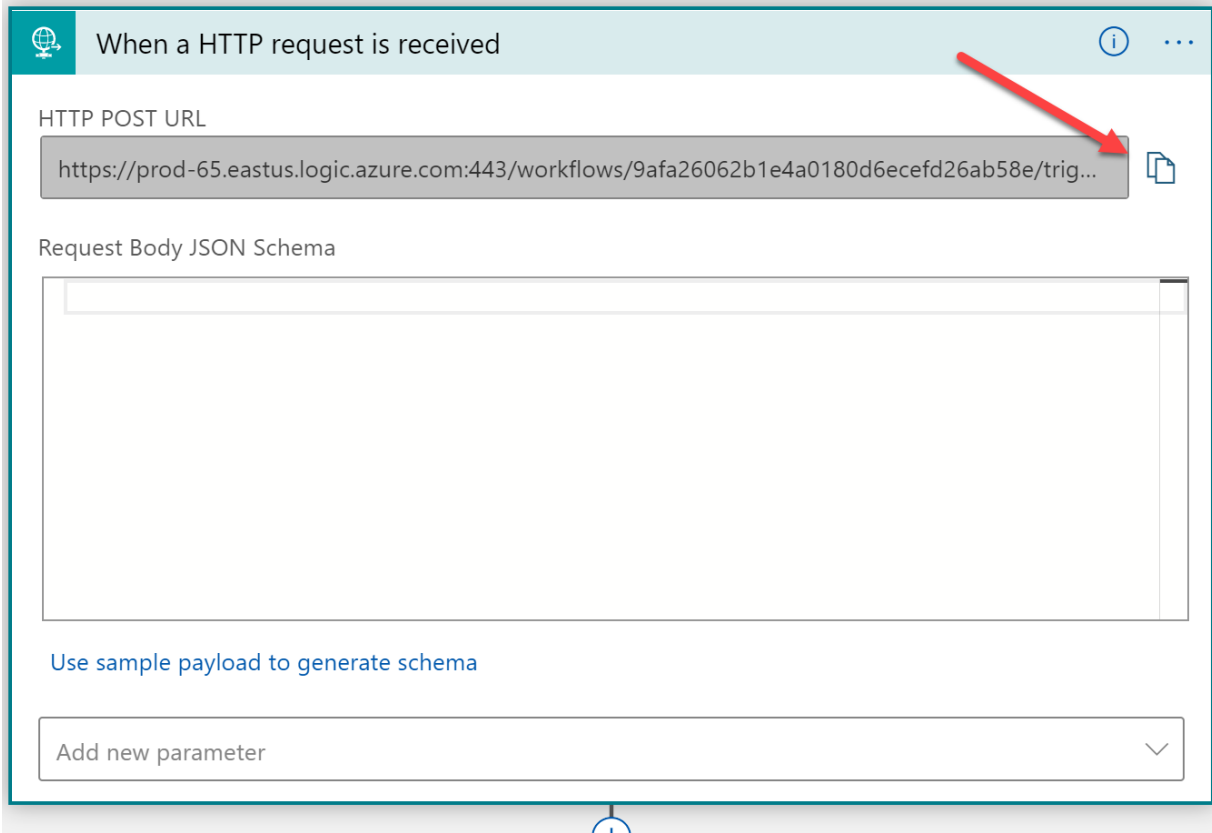
- *Record Type**: A dropdown menu with 'Incident' selected.
- Display System References**: A dropdown menu with 'No' selected.
- Exclude Reference Links**: A dropdown menu with 'Yes' selected.
- Number**: A field with a blue icon and the text 'name x'.
- Short description**: A field with a blue icon and the text 'Description x'.
- Description**: A field with a blue icon and the text 'Description x'.
- Severity**: A field with a blue icon and the text 'Severity x'.
- Add new parameter**: A dropdown menu with 'Add new parameter' selected.

At the bottom of the form, there is a status bar that says 'Connected to SNOW. [Change connection.](#)'

2. Deploy the logic app that act as a webhook.

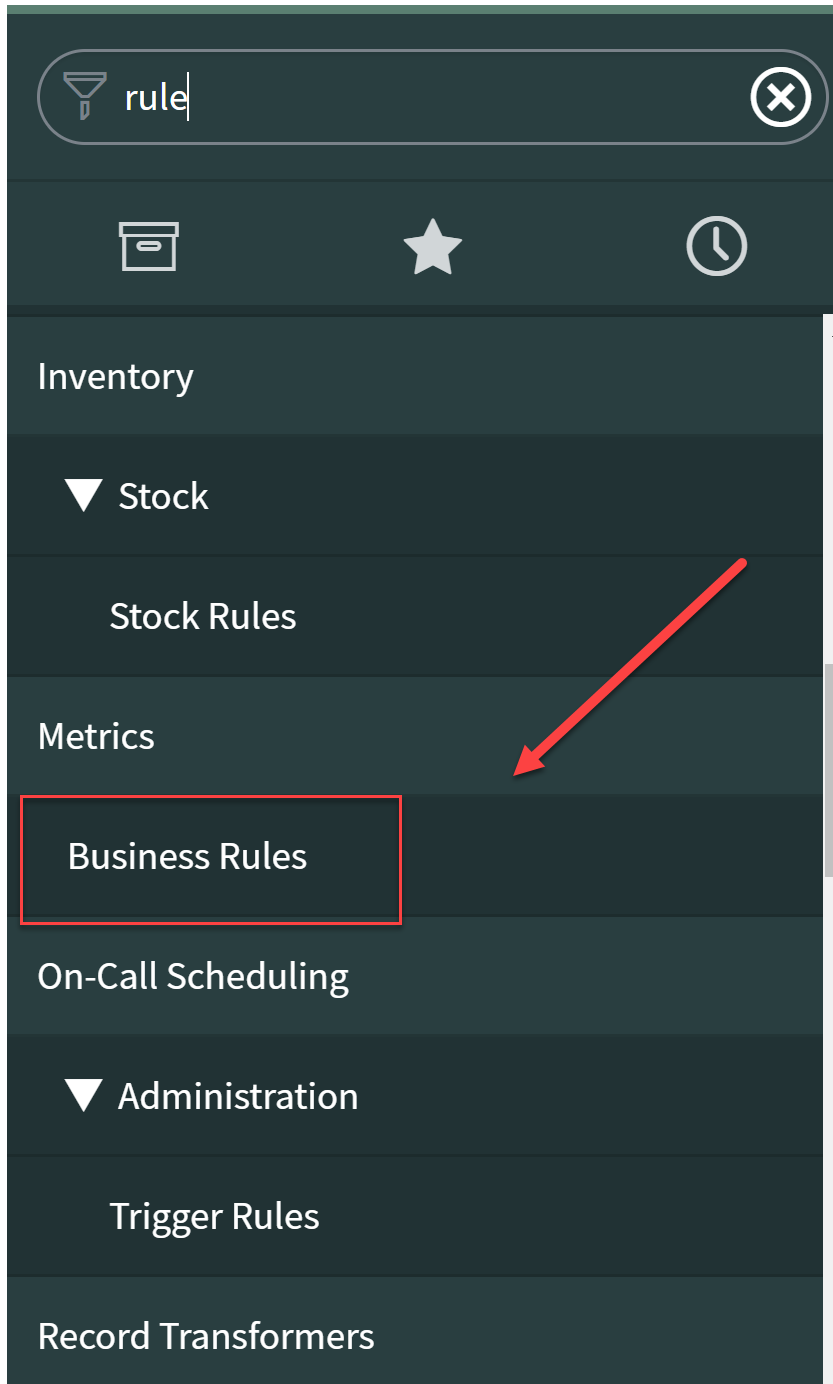
Use this link to [deploy](#) the logic app (complete the pre-requisites)

3. Open the logic app and copy the http endpoint, we will use the value in the business rule code.



Create business Rule on Service Now:

1. On serviceNow open business rule



2. Create new business rule.

The results should look like this:


```
{  
    var state = String(current.incident_state);  
    var id = String(current.u_eventid).replace(/["']/g, "");  
    var ClosedUser = String(current.caller_id);  
    var Description =  
    current.short_description.replace(/(\\r\\n|\\n|\\r|["'])/gm, " ");  
    var number = String(current.number);  
    var group = current.assignment_group;  
    var request = new sn_ws.RESTMessageV2();  
    var requestBody = {"Description": Description , "number": number ,  
    "state": state , "ClosedBy":ClosedUser };  
    request.setRequestBody(JSON.stringify(requestBody));  
    request.setEndpoint('https://prod-  
65.eastus.logic.azure.com:443/workflows/9afa26062b1e4a0180d6ecef  
d26ab58e/triggers/manual/paths/invoke?api-version=2016-10-  
01&sp=%2Ftriggers%2Fmanual%2Frun&sv=1.0&sig=gv1HMcDt8DanJm  
Oe3UvG22uyU_nere4rTQF8XnInYog');  
    request.setHttpMethod('POST');  
    request.setRequestHeader("Accept","application/json");  
    request.setRequestHeader('Content-Type','application/json');  
    var response = request.execute();  
    var responseBody = response.getBody();  
    var httpStatus = response.getStatusCode();
```

```
// this var stores the response from Jira included the unique ID of  
the issue we created.
```

```
var parsedData = JSON.parse(responseBody);  
//var response = request.execute();  
gs.log(response.getBody());  
}
```

4. Save the business rule.