Health Tracker Compare Application

# Health Tracker Compare Application Summary

The Health Tracker Compare Application is to be used in conjunction with the Health Tracker (<https://developer.servicenow.com/app.do#!/share/contents/4953204_health_tracker?t=PRODUCT_DETAILS>). The Health Tracker Compare Application allows organizations to run a Sprint Scan on a higher environment and take the returned findings that are aligned to an update set and compare them with a lower environment. When comparing them to a lower environment results will be returned identifying if developers have run Custom Scans on their update sets. The Health Tracker Compare Application will enable organizations the ability to monitor developers and understand if they are following the procedure set forth by management that is requiring them to run Custom Scans on all update sets and resolve all findings identified by the Custom Scans or ignoring the Findings due to business requirements.

Table of Contents

[Health Tracker Compare Application Summary 1](#_Toc19397369)

[Implementation Process 2](#_Toc19397370)

[Configuring Lower Environments 2](#_Toc19397371)

[Configuring Higher Environment 2](#_Toc19397372)

[Baselining the Higher Environment 3](#_Toc19397373)

[Running Health Tracker Compare Application in Higher Enviroment 4](#_Toc19397374)

[Design Details 6](#_Toc19397375)

# Implementation Process

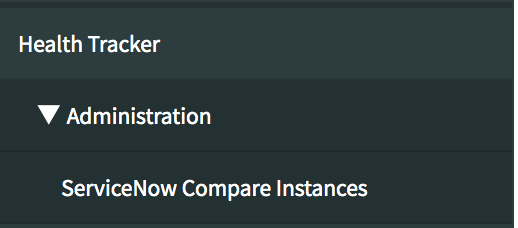
1. Apply Health Tracker (<https://developer.servicenow.com/app.do#!/share/contents/4953204_health_tracker?t=PRODUCT_DETAILS>) to lower environments and higher environment.
2. Apply update set ‘HealthTrackerCompare\_CoreChanges’ to lower environments. **\*Note\*** When apply update set you will receive message regarding deleting certain fields. There is no issue with this. Those fields are not needed.
   1. These are the environments where the developers will be running the Custom Scans. Most likely Dev1, Dev2, Dev3 and etc.
3. Apply update set ‘HealthTrackerCompare\_CoreChanges’ to higher environment. **\*Note\*** When apply update set you will receive message regarding deleting certain fields. There is no issue with this. Those fields are not needed.
   1. This is the environment where code will move. Most likely Test.

# Configuring Lower Environments

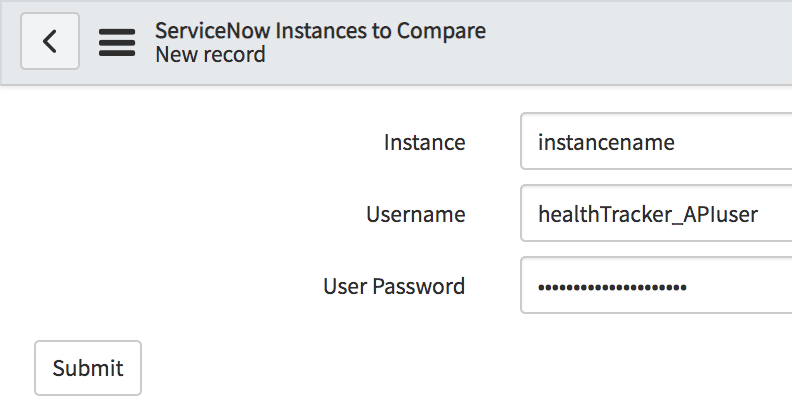
1. Ensure Health Tracker (<https://developer.servicenow.com/app.do#!/share/contents/4953204_health_tracker?t=PRODUCT_DETAILS>) and update set ‘HealthTrackerCompare\_CoreChanges’ has been applied.
2. Each lower environment will need an integration user configured for it. So, the below steps could be repeated more than once.
3. To do this navigate to User Administration > Users
4. Select ‘New’
5. Create a Service account User with necessary information and assign it a Password. Set ‘Web service access only’ and ‘Internal Integration User’ as true.
   1. This is specifically to ensure that this user has only integration user access and no access to the environment.
6. Select ‘Submit’
7. Navigate back to that service account user.
8. Via the related list ‘Roles’ select edit
9. Give the following roles to the user:
   1. rest\_service
   2. x\_snc\_ht.user

# Configuring Higher Environment

1. Ensure Health Tracker (<https://developer.servicenow.com/app.do#!/share/contents/4953204_health_tracker?t=PRODUCT_DETAILS>) and Update set ‘HealthTrackerCompare\_CoreChanges’ has been applied.
2. Navigate to Health Tracker > Administration > ServiceNow Compare Instances



1. Select ‘New’
2. Populate the follow fields:
   1. Instance – This should be the instance name. DO NOT include anything other than the instance name
   2. Username – This should be the aligned username of the integration user that was created in lower environment.
   3. Password – provide the password given to the user.



1. Select ‘Submit’

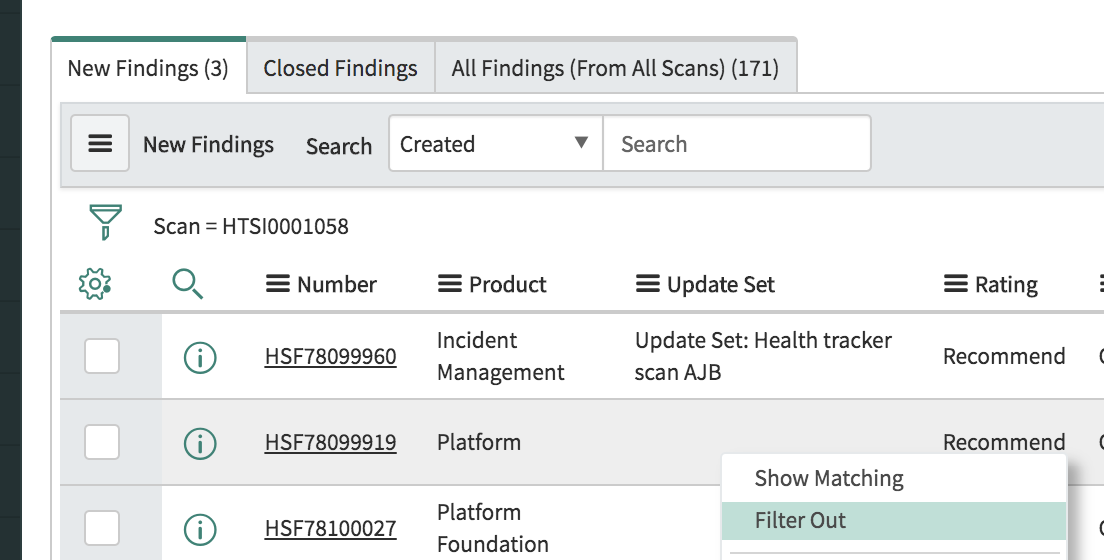
# Baselining the Higher Environment

1. To start this process the higher environment must be Baselined. This is the concept of running a Sprint Scan via the Health tracker.
2. To do this Navigate to Health Tracker > New Scan
3. Select the Access Token (this should already be configured based upon the Health Tracker instructions)
4. Select the Scan type as ‘Sprint Scan’
5. Select ‘Save’
6. Then select ‘Start Scan’
7. This will run for longer than 30 mins when complete the findings will be returned and the record with be in State = Complete

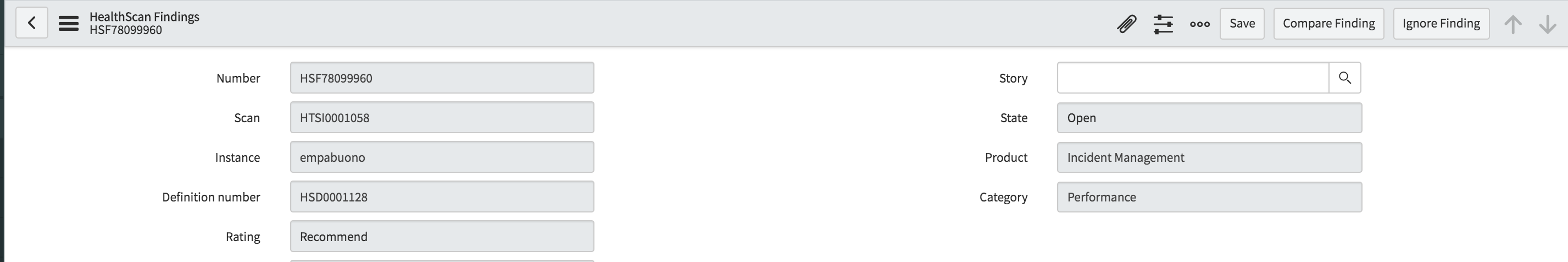
Once this is complete you have now baselined the instance. Meaning all findings of the higher environment have been identified and any new findings that are identified will be net-new brought over from developers update sets.

# Running Health Tracker Compare Application in Higher Environment

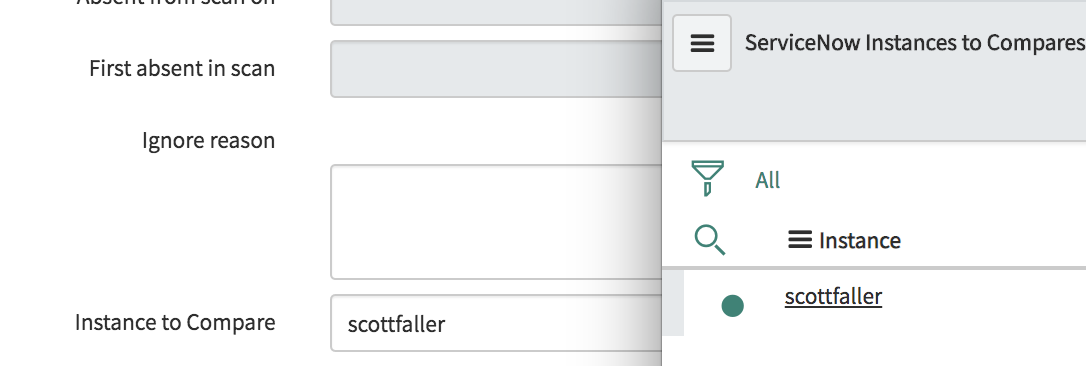
1. Based upon your organizations regular cadence to move development work to the higher environment once this is complete you will want to execute Sprint Scan via the Health tracker.
2. To do this Navigate to Health Tracker > New Scan
3. Select the Access Token (this should already be configured based upon the Health Tracker instructions)
4. Select the Scan type as ‘Sprint Scan’
5. Select ‘Save’
6. Then select ‘Start Scan’
7. This will run for longer than 30 mins when complete the findings will be returned and the record with be in State = Complete
8. Once complete via the ‘New findings’ related list filter out any findings that do not have an aligned update set.



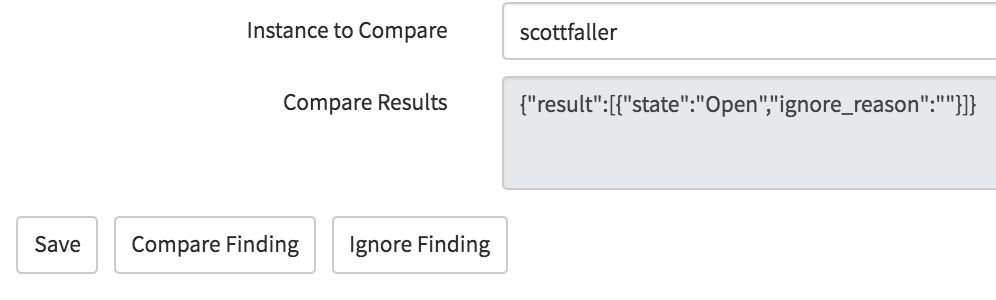
1. The remaining findings are findings that were deployed via the current release cycle.
2. Click into one of the findings.
3. Once clicking into one of the findings you are trying to identify 1 of the following things.
   1. A Custom Scan was never ran on the update set. This means Code is being deployed into a high environment without following the procedure of scanning the update set and resolving all findings identified.
   2. A Custom Scan was run on the update set but the findings was not Closed. This means the developer did not take the proper steps to Close the finding.
   3. A Custom Scan was run on the update set and the findings was marked as Ignore due to business requirements. This means the developer properly marked the finding as Ignore and explained via the Ignore Reason why the finding was ignored.



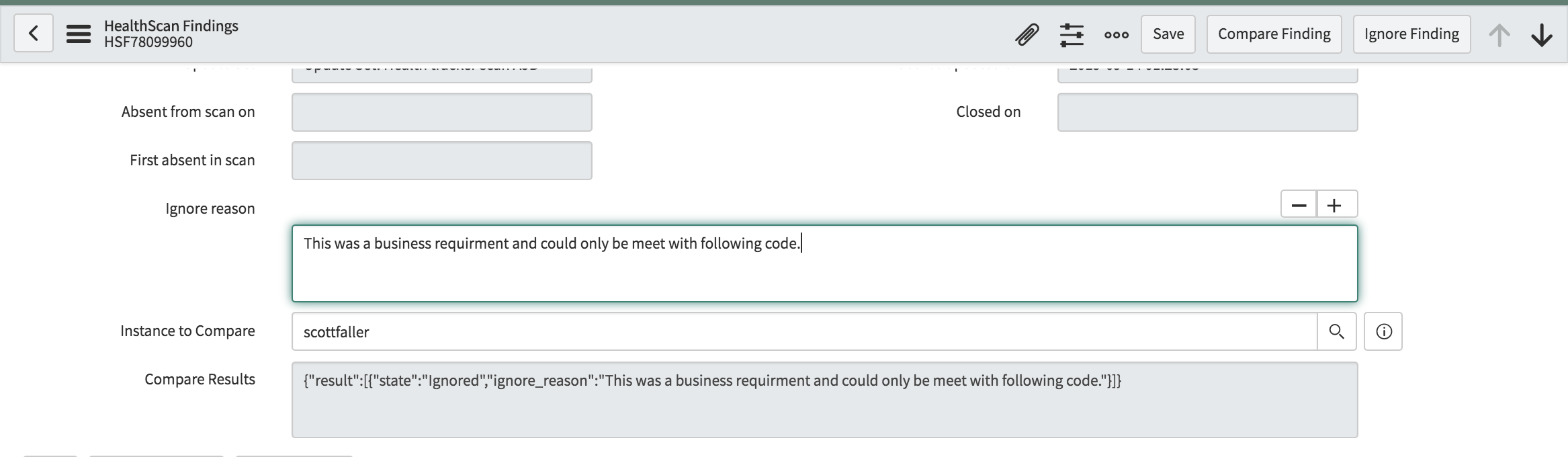
1. Via the field ‘Instance to Compare’ select the Instance where the update set came from.



1. Select ‘Compare Finding’.
2. The results will be returned in the field ‘Compare Results’. Eg. pwcnetworkdev1, pwcnetworkdev2, etc.



1. The different results you can expect are as followed:
   1. Finding was not resolved.
      1. {"result":[{"state":"Open","ignore\_reason":""}]}
   2. Custom Scan has not been ran on the update set.
      1. {"result":[{"error":"NO Scan has been Run on offending Update set"}]}
   3. the update set was not found. So most likely you selected the wrong ‘Instance to Compare’
      1. {"result":[{"error":"Update Set NOT FOUND"}]}
   4. The developer set the finding to ignore and provided an appropriate Ignore Reason.
      1. {"result":[{"state":"Ignored","ignore\_reason":" Business requirement by Org."}]}
         1. After reviewing the Ignore Reason and agreeing with it. Input the ignore reason into the field ‘Ignore Reason’ and select ‘Ignore Finding’.



# Design Details

1. Scripted REST API: HealthTrackerCompare
   * 1. Resource: HealthTrackerCompareUpdateSet
     2. HTTP Method: GET
     3. Resource Path: /api/ x\_snc\_ht/healthtrackercompare/HealthTrackerCompareUpdateSet/{name}/{finding}
2. UI Action: Compare Finding