netnology
simplifying SDx and Cloud

Cisco AppDynamics, ThousandEyes & Intersight Workload Optimizer Integration Guide

netnology, LLC.
Corporate Headquarters
6203 San Ignacio Avenue, Suite 110
San Jose, CA 95119
http://www.netnology.io
Tel: 415-483-0819

Table of Contents

Confidentiality Statement

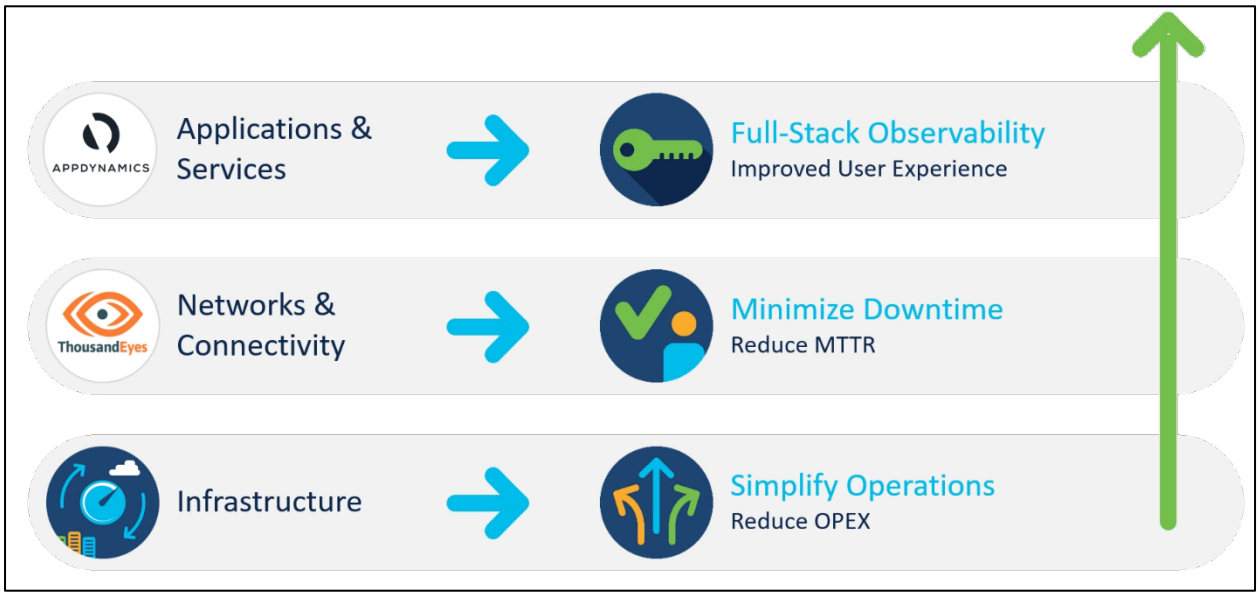
1 - Executive Summary

Cisco is doubling down on integrating cross-domain architectures which address customer business needs. These simple cross-domain integration examples span Cisco SDx and Cloud solutions, while having a laser focus on security, analytics, and automation. In other words, one part of Cisco making another part of Cisco even better.

Cisco AppDynamics, ThousandEyes (TE) and Intersight Workload Optimizer (IWO) integration can demonstrate full-stack observability from end user to the underlying infrastructure of an application. By implementing this integration, customers will enhance organizational alignment and productivity for DevOps, Site Reliability Engineering (SRE), application and infrastructure teams.

2 - AppDynamics, ThousandEyes & Intersight Workload Optimizer Integration Overview

The integration of AppDynamics, ThousandEyes and Intersight Workload Optimizer provides full-stack observability to any organization. Cisco's AppDynamics is used to monitor and display the services of an application. Cisco's ThousandEyes is needed to monitor and display the network connectivity from end users to the applications, irrespective of the end user's location. Cisco's Intersight Workload Optimizer is required to monitor, display and analyze, and proactively scale infrastructure resources based on needs of the application. In this guide we will go through the step-by-step process of integrating these three solutions and demonstrate improved user-experience, reduced MTTR and OPEX.



2.1 - AppDynamics and IWO Prerequisites

- Hypervisor environment to deploy AppDynamics (on prem) or AppDynamics SaaS
- Environment setup with communication to an on prem application
- Known application already setup in AppDynamics or an application to start monitoring
- AppDynamics license
- Intersight setup
- Intersight licenses added for IWO
- IWO Hypervisor target already configured
- All login information for application components (root or administrator rights)
- General requirements listed below:
  - https://docs.appdynamics.com/display/PRO45/Platform+Requirements
  - https://community.appdynamics.com/t5/Knowledge-Base/SaaS-Network-Requirements-IP-Ranges-to-connect-to-AppDynamics/ta-p/28352

2.2 - AppDynamics and TE Prerequisites

- Hypervisor environment to deploy AppDynamics (on prem), or a SaaS AppDynamics Environment setup with communication to on prem application
- Known application already setup in AppDynamics or an application to start monitoring
- AppDynamics license
- TE SaaS license
- TE SaaS initial setup completed
- All login information for application components (root or administrator rights)
- General requirements listed below:
  - https://docs.appdynamics.com/display/PRO45/Platform+Requirements
  - https://community.appdynamics.com/t5/Knowledge-Base/SaaS-Network-Requirements-IP-Ranges-to-connect-to-AppDynamics/ta-p/28352

2.3 - Logical Topology

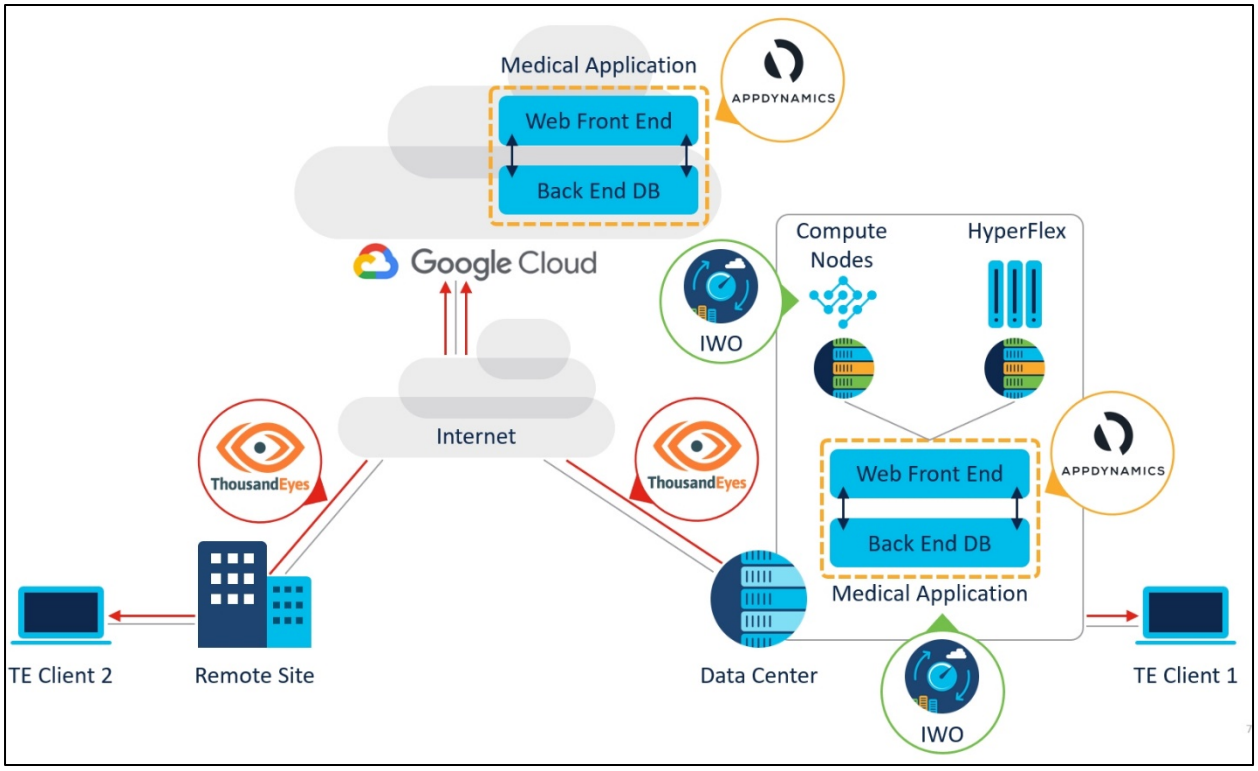
Below is a high-level topology for this demonstration. In this environment, we have a Cisco Hyperflex cluster that is running vCenter 6.7 and ESXI host of 6.5. We also have an on prem instance of AppDynamics deployed.

Our multi-tier Medical Application is deployed on prem and in Google Cloud. All of these are connected to a set of fabric interconnect switches which then uplink to a set of core switches to the rest of the network.

Our Intersight Assist VM is deployed on prem which has communication to IWO in the cloud and can communicate to all other resources deployed on prem.

We have two client machines with TE enterprise agents accessing our Medical Application.

Cisco TE Application is SaaS based so the communication from our on-prem AppDynamics instance is being allowed to talk with the Cisco TE Instance.



2.4 - Software Versions

This integration was developed with the following software version of each component.

Components	Software Version
Cisco Intersight	Latest SaaS Version
AppDynamics Controller	Build 4.5.16.2272
VMware vCenter	6.7
ESXi Host	6.5
ThousandEyes	Latest SaaS Version
Cisco Intersight Virtual Appliance for vSphere	1.0.3.230

3 - AppDynamics and IWO Integration

Following are the key components that need to be configured for this integration.

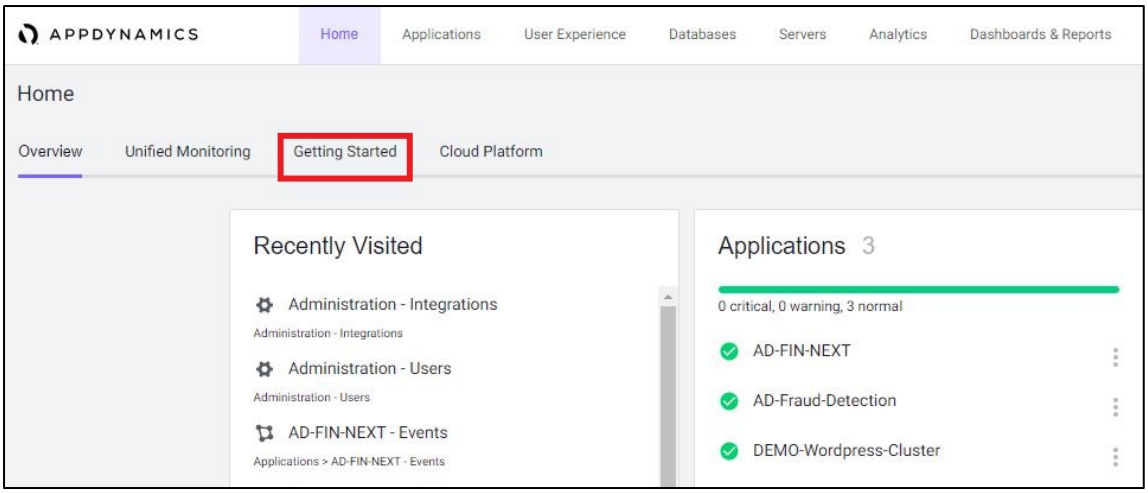
- AppDynamics Agent setup
    - Database Agent
- AppDynamics IWO User Creation
- Intersight Assist VM Install
    - Integrate Assist VM with Intersight
- IWO target configuration for AppDynamics

3.1 - AppDynamics Agent Setup Overview

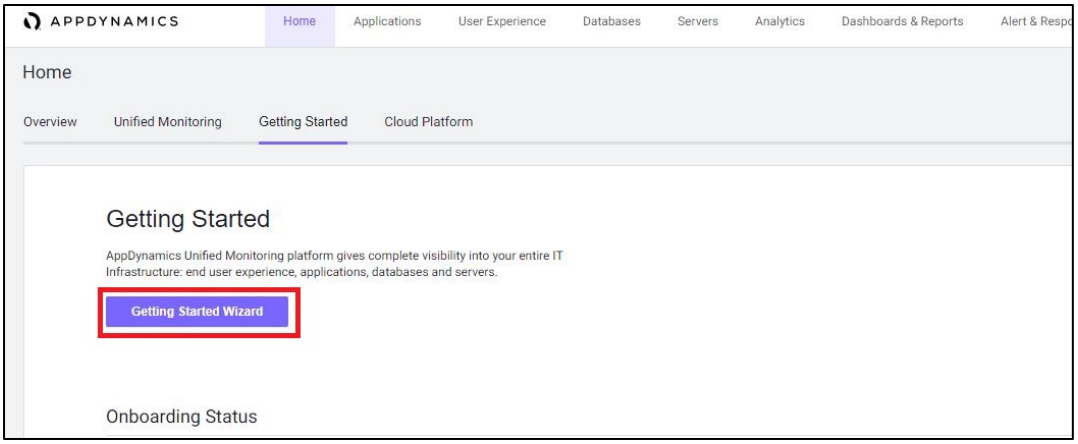
This section describes the process for setting up a Database agent on a MySQL Server and connecting it back to the AppDynamics application.

3.1.1 - Database agent

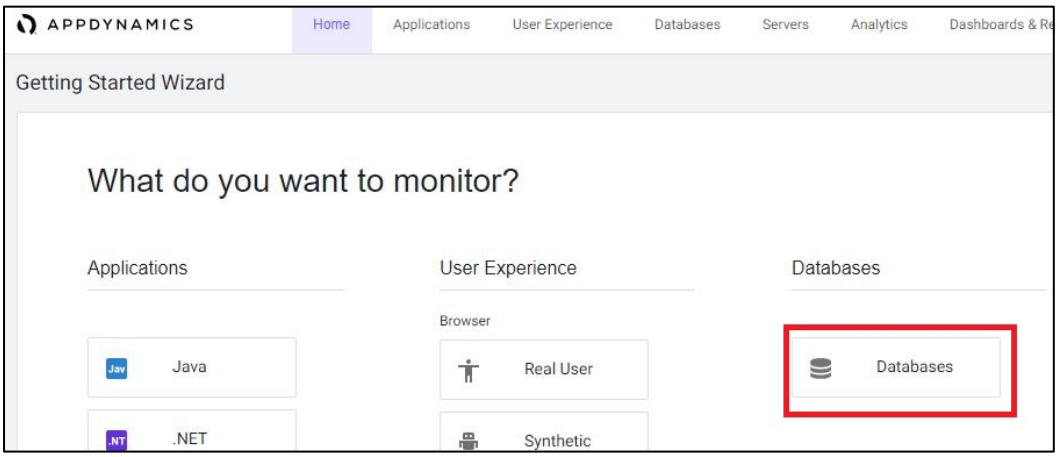
Start from the AppDynamics Controller home page. After logging into the Controller, **click** on **Getting Started**.



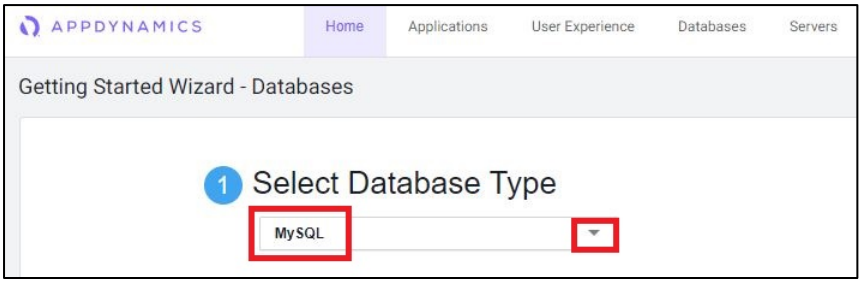
From the ‘Getting Started’ page, **click** on the **Getting Started Wizard** button.



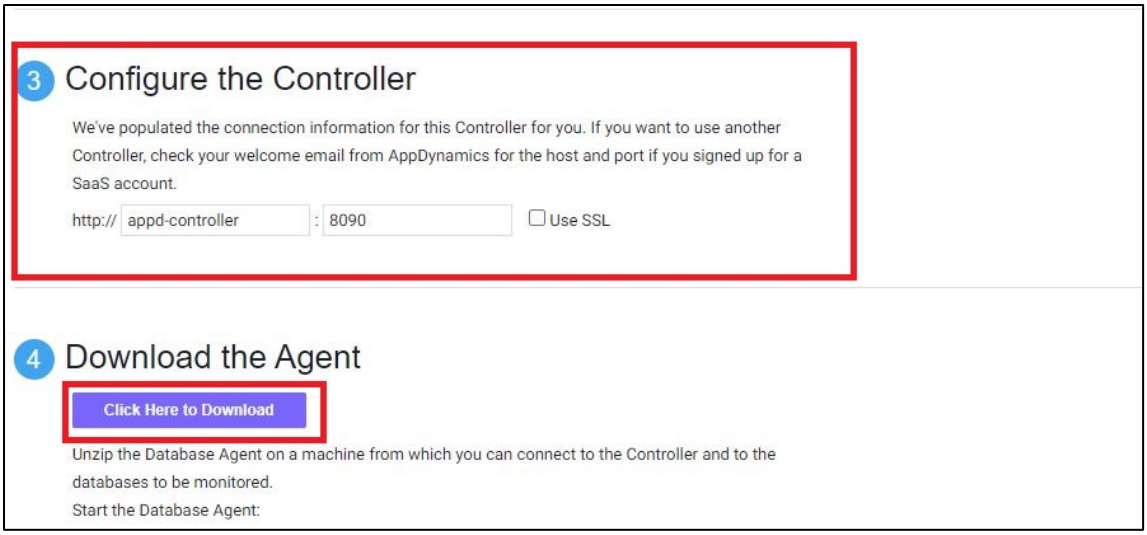
In the wizard menu for "What do you want to monitor?" area, **click** on the **Databases icon** under the Databases section.



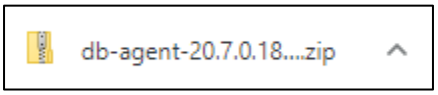
In the next window, in the drop-down box, select the database that needs to be connected. For this integration, **'MySQL'** was selected.



Scroll down the screen to **verify** that the information in '**Configure the Controller'** section is correct. Next, move down to the '**Download the Agent'** section and **click** on '**Click Here to Download'.**



From here, AppDynamics application prepares the **db-agent** zip file that would be downloaded.



After the **db-agent** has been downloaded, move/copy this entire zip folder over to the server that has MySQL server running. For this integration, MySQL was setup on an Ubuntu server where it was running a few database applications.

In the DEMO-MYSQL-03 server, it can be seen that Java and unzip were installed using apt-get. There was also a directory created in the /home/administrator area called **dbagent**. The zip file was copied to this folder. Permissions of this directory were also changed.

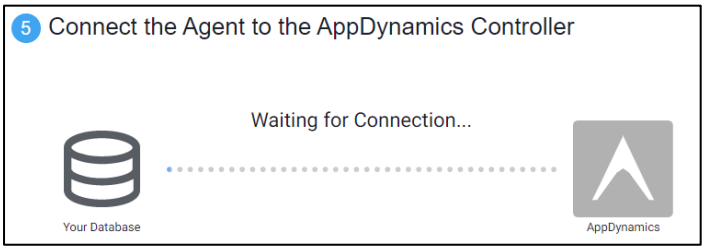
```
root@DEMO-MYSQL-03:/home/administrator# apt-get install -y default-jre
Reading package lists... Done
Building dependency tree
Reading state information... Done
default-jre is already the newest version (2:1.8-56ubuntu2).
0 upgraded, 0 newly installed, 0 to remove and 20 not upgraded.
root@DEMO-MYSQL-03:/home/administrator# apt-get install unzip
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  zip
The following NEW packages will be installed:
  unzip
0 upgraded, 1 newly installed, 0 to remove and 20 not upgraded.
Need to get 158 kB of archives.
After this operation, 530 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 unzip amd64 6.0-20ubuntu1 [158 kB]
Fetched 158 kB in 0s (617 kB/s)
Selecting previously unselected package unzip.
(Reading database ... 89655 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-20ubuntu1_amd64.deb ...
Unpacking unzip (6.0-20ubuntu1) ...
Processing triggers for mime-support (3.59ubuntu1) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up unzip (6.0-20ubuntu1) ...
root@DEMO-MYSQL-03:/home/administrator# mkdir dbagent
root@DEMO-MYSQL-03:/home/administrator# chmod -R 777 dbagent/
root@DEMO-MYSQL-03:/home/administrator# ls
dbagent  percona-release_0.1-4.xenial_all.deb  percona-release_0.1-6.xenial_all.deb  percona-rel
root@DEMO-MYSQL-03:/home/administrator# cd dbagent/
root@DEMO-MYSQL-03:/home/administrator/dbagent# ls
db-agent-20.7.0.1892 (3).zip
root@DEMO-MYSQL-03:/home/administrator/dbagent#
```

Proceed to **unzip** the **db-agent-20.7.0.1892.zip** file while in the current /home/administrator/dbagent directory by running **unzip db-agent-20.7.0.1892.zip** command.

Following are the files obtained after unzipping.

```
root@DEMO-MYSQL-03:/home/administrator/dbagent# ls
auth  conf  db-agent-20.7.0.1892 (3).zip  db-agent.jar  keys  lib  monitors  start-dbagent  start-dbagent.bat  vbs
root@DEMO-MYSQL-03:/home/administrator/dbagent#
```

Be mindful that these steps are done to get the **dbagent** running on the server. However, in the AppDynamics Controller, the wizard is still waiting for a connection, once the **dbagent** start running.



Next, back on DEMO-MYSQL-03 Server, edit the controller-info.xml file with your favorite text editor. To do this, we need to cd to the conf/ directory where this file is located. Now, we can edit this file.

```
root@DEMO-MYSQL-03:/home/administrator/dbagent# ls
auth  conf  db-agent-20.7.0.1892 (3).zip  db-agent.jar  keys  lib  monitors  start-dbagent  start-dbagent.bat  vbs
root@DEMO-MYSQL-03:/home/administrator/dbagent# cd conf/
root@DEMO-MYSQL-03:/home/administrator/dbagent/conf# ls
controller-info.xml  logging
root@DEMO-MYSQL-03:/home/administrator/dbagent/conf# vi controller-info.xml
root@DEMO-MYSQL-03:/home/administrator/dbagent/conf#
```

While in the editor, do verify that all the information in the Tag lines is correct.

```
<controller-info>

    <!--  For a more detailed information on different configurations that an agent can support
    and the respective rules, one should reference following URL:
    https://docs.appdynamics.com/display/PRO42/Java+Agent+Configuration+Properties
    In general configuration properties may be set in different ways. Here is the order
    of precedence how the agent will read and respect specific configurations.
      1. Environment Variables
      2. System Properties
      3. Version specific controller-info.xml. See .../$(Agent-Base-Install-Dir)/ver$(agent-version-no)/conf/controller-info.xml
      4. Default/Root controller-info.xml. See .../$(Agent-Base-Install-Dir)/conf/controller-info.xml  -->

    <!--  This is the host name or the IP address of the AppDynamics Controller e.g. 192.168.1.22 or myhost or myhost.abc.com
    This is the same host that you use to access the AppDynamics browser based User interface.
    This can be overridden with an environment variable 'CONTROLLER_HOST_NAME' or
    the system property '-Dappdynamics.controller.hostName'  -->

    <controller-host>appd-controller</controller-host>

    <!--  This is the http(s) port of the AppDynamics Controller , If 'controller-ssl-enabled' below is set to true, you must
    specify the HTTPS port of the Controller, otherwise specify the HTTP port. The default values are 8090 for HTTP and 8181 for HTTPS
    This is the same port that you use to access the AppDynamics browser based User interface.
    This can be overridden with an environment variable 'APPDYNAMICS_CONTROLLER_PORT' or
    the system property '-Dappdynamics.controller.port'  -->

    <controller-port>8090</controller-port>

    <!--  This specifies if the AppDynamics agent should use SSL(HTTPS) to connect to the Controller. If you set this to true,
    the controller port property above should be set to the HTTPs port of the Controller.
    This can be overridden with an environment variable 'APPDYNAMICS_CONTROLLER_SSL_ENABLED' or
    the system property '-Dappdynamics.controller.ssl.enabled'  -->

    <controller-ssl-enabled>false</controller-ssl-enabled>

    <!--  If this value is set to true, it indicates all the passwords and access-keys must be encrypted.
    If this value is false or undefined, passwords and access-keys will be in plain-text.
    One must reference following URL for the process and guidelines of encrypting passwords:
    https://docs.appdynamics.com/display/PRO42/Java+Agent+Configuration+Properties  -->

    <use-encrypted-credentials></use-encrypted-credentials>

    <!--  Full qualified path name for the SCS-KeyStore file name.  -->

    <credential-store-filename></credential-store-filename>

    <!--  Password for the 'Secure Credential Store' (SCS). This password must be obfuscated.  -->

    <credential-store-password></credential-store-password>

    <!--  If the AppDynamics Controller is running in multi-tenant mode or you are using the AppDynamics SaaS Controller
    you must specify the account name and account access key for this agent to authenticate with the controller.
    If you are running in single-tenant mode (the default) there is no need to configure these values.
    This can be overridden with an environment variable 'APPDYNAMICS_AGENT_ACCOUNT_NAME' or
    the system property '-Dappdynamics.agent.accountName'  -->

    <account-name>customer1</account-name>

    <!--  This can be overridden with an environment variable 'APPDYNAMICS_AGENT_ACCOUNT_ACCESS_KEY' or
    the system property '-Dappdynamics.agent.accountAccessKey'
    This account-access-key must be encrypted if 'use-encrypted-credentials' is set to true  -->
```

If everything is correct, scroll down and add a few lines to let AppDynamics know which application this node belongs to, including the node name and node's tier.

```
    <!--  This can be overridden with an environment variable 'APPDYNAMICS_AGENT_ACCOUNT_ACCESS_KEY' or
    the system property '-Dappdynamics.agent.accountAccessKey'
    This account-access-key must be encrypted if 'use-encrypted-credentials' is set to true  -->

    <account-access-key>5bf4149e-119d-4796-b77c-0edcefb70971</account-access-key>
        <force-agent-registration>true</force-agent-registration>
        <application-name>DEMO-Wordpress-Cluster</application-name>
        <node-name>DEMO-MYSQL-03</node-name>
        <tier-name>Database</tier-name>
```

After adding the few lines, go ahead and exit out of the text editor while saving changes to the file.

Moveout of the /conf directory and back into the dbagent home directory by running **cd ..** command.

```
root@DEMO-MYSQL-03:/home/administrator/dbagent/conf# cd ..
root@DEMO-MYSQL-03:/home/administrator/dbagent# ls
auth  conf  db-agent-20.7.0.1892 (3).zip  db-agent.jar  keys  lib  monitors  start-dbagent  start-dbagent.bat  vbs
root@DEMO-MYSQL-03:/home/administrator/dbagent#
```

Now, the dbagent is ready to be started on the DEMO-MYSQL-03 server. This can be achieved in two different ways.

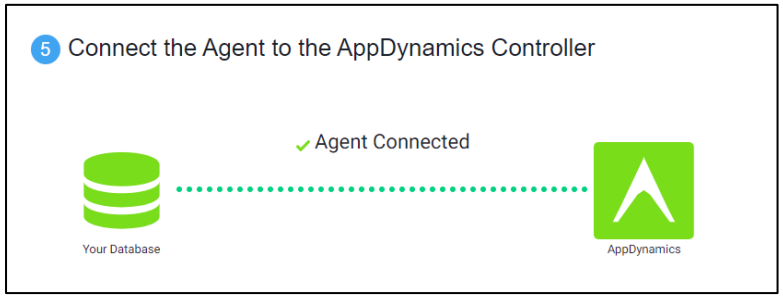
1) Running command java -jar <db-agent-home>/db-agent.jar
2) Running command ./start-dbagent

Once started, it should look like this.

```
20 Aug 2020 14:25:06,084  INFO [DBAgent-1] RegistrationChannel:132 - Auto agent registration SUCCEEDED!
20 Aug 2020 14:25:06,102  INFO [DBAgent-1] AgentRegistrationTask:44 - Registered DB_AGENT with agentID[783]
20 Aug 2020 14:25:06,102  INFO [DBAgent-1] Agent:239 - Starting Default Database Agent....
20 Aug 2020 14:25:06,116  INFO [DBAgent-1] Agent:303 - Skew-Handler is : enabled
20 Aug 2020 14:25:06,139  INFO [DBAgent-1] DBAgentConfigManager:46 - Scheduling configuration refresh at an interval of 10 seconds
20 Aug 2020 14:25:06,141  INFO [DBAgent-1] DBAgentConfigManager:46 - Configuration refresh task interval is 10 seconds
20 Aug 2020 14:25:06,143  INFO [DBAgent-1] Agent:341 - Configuration manager successfully configured
20 Aug 2020 14:25:06,158  INFO [DBAgent-1] EventService:39 - Event Generation Service is : enabled
20 Aug 2020 14:25:06,299  INFO [DBAgent-1] DBAgentMonitor:35 - DBAgentMonitor started
20 Aug 2020 14:25:06,299  INFO [DBAgent-1] Agent:246 - Started Default Database Agent successfully
```

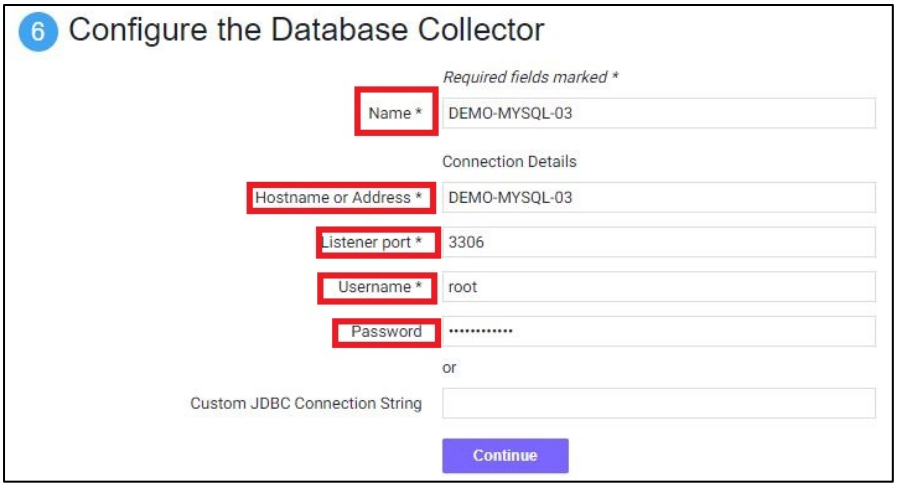
After the agent has started, we go back to the AppDynamics application to move to the next steps of configuring the db collector.

After a few minutes, Section 5 of the wizard should show that your dbagent is now connected with AppDynamics.

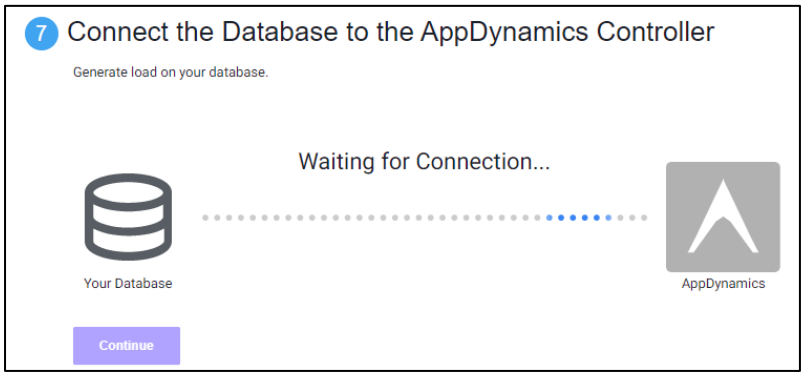


For Section 6 of the wizard in AppDynamics, the following parameters need to be configured:

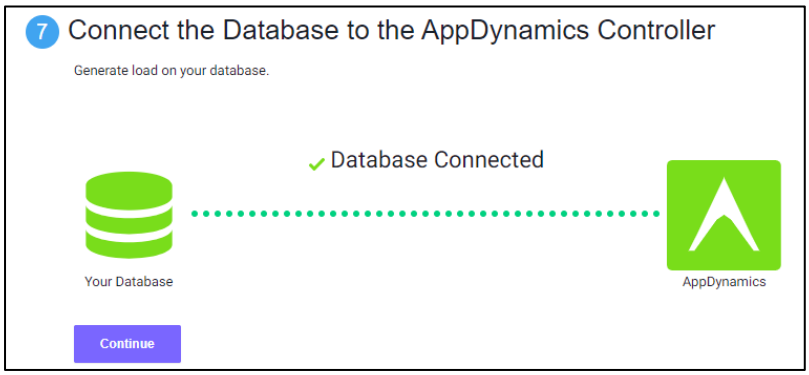
1) Name (name for the collector)
2) Hostname or Address (FQDN or IP Address)
3) Listener port (port that MySQL is listening on)
4) Username (MySQL account with the correct permissions)
5) Password (for configured Username)
6) Or you can use a Custom JDBC Connection String



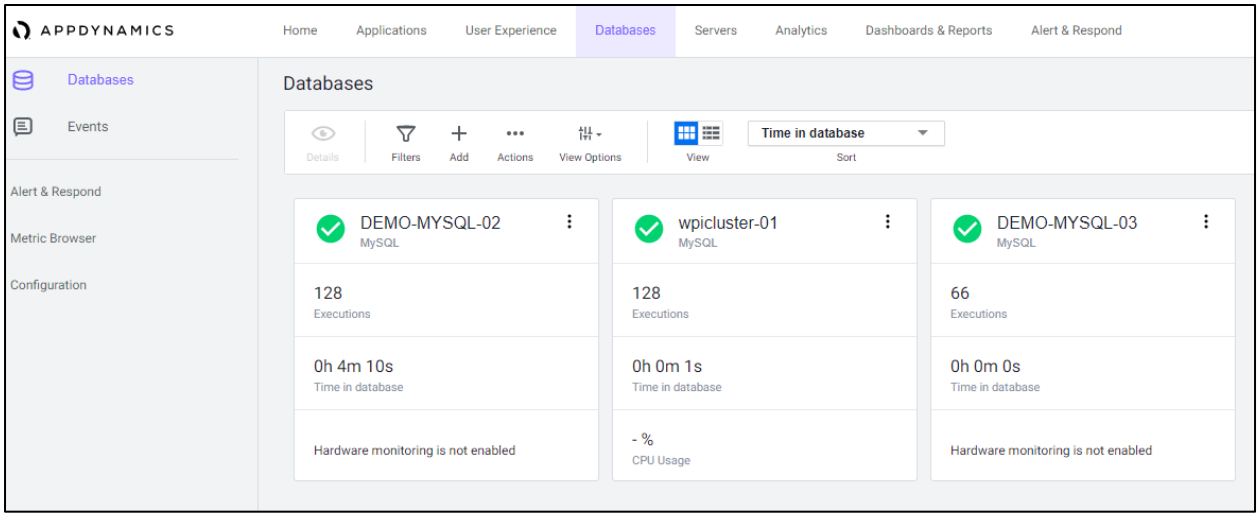
Then **click** on the **Continue** button.



After putting a small amount of load on the DB Server and waiting for a few minutes, see that the connection to AppDynamics is connected.



Now **click** on **Continue** button to exit out of the wizard. From the Database tab, see a collector called DEMO-MYSQL-03, which was just created.



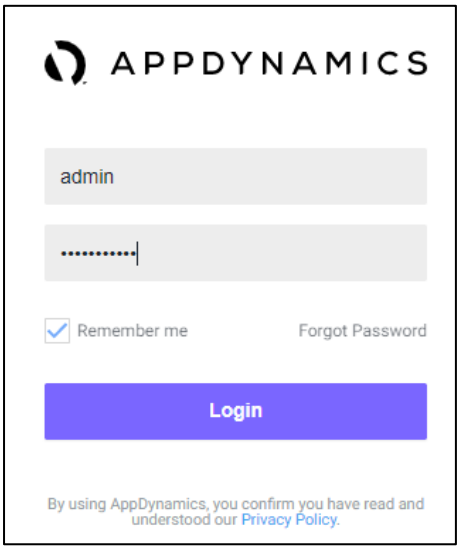
3.2 - AppDynamics IWO User Creation

In this section, we will be going over the creation of a user in AppDynamics application to be used for the integration or target creation in IWO.

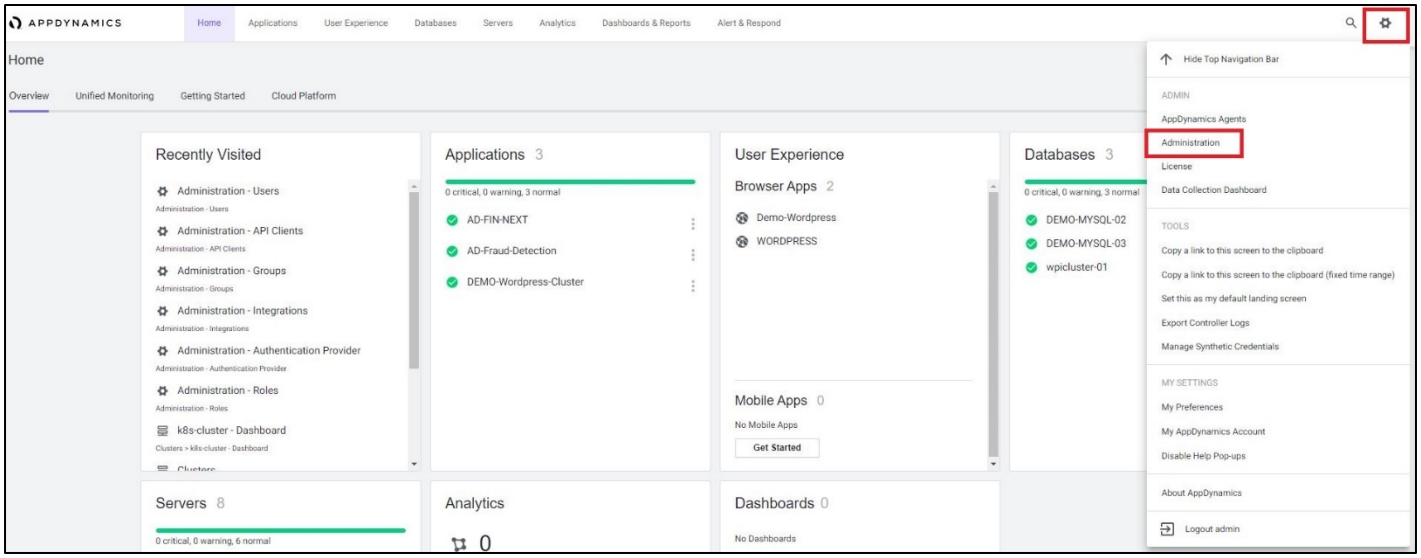
3.2.1 - User Creation

Start off in the AppDynamics application.

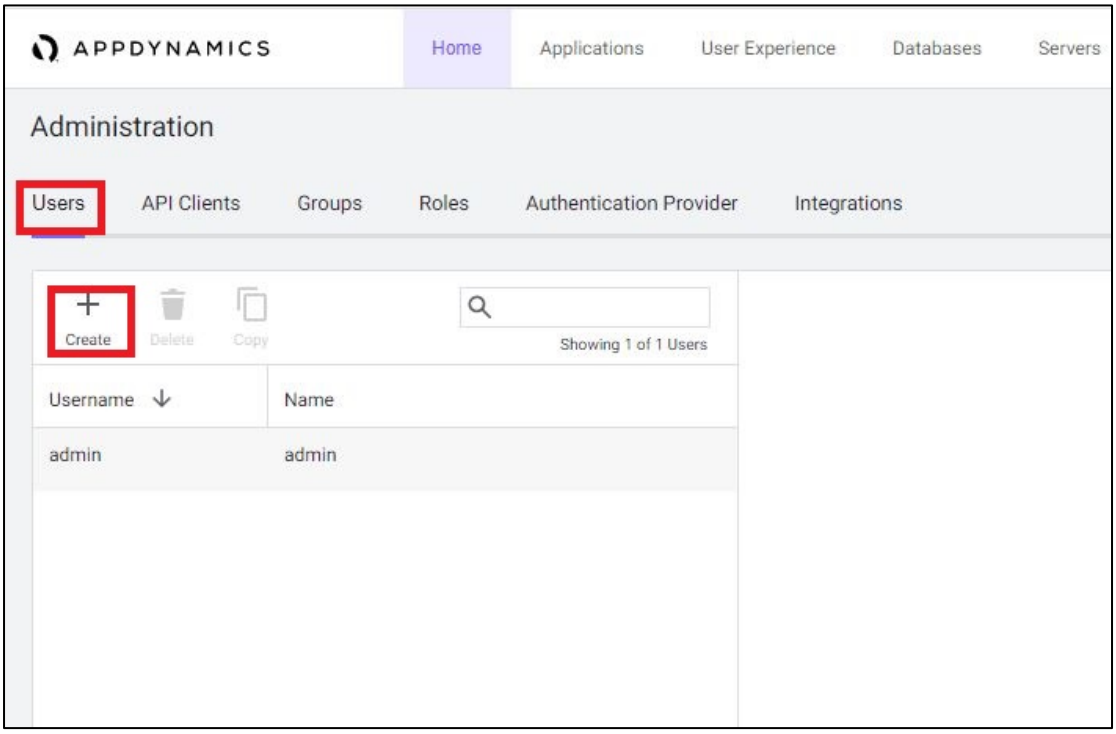
On the login screen, proceed with the root/Administrator account credentials. And **click** on **'Login'** button.



On the AppDynamics home page, in the top right-hand corner, **click** on the '**Cog'**, then **click** on the **'Administration'** button.



From the Administration section, **select** the **User's tab** and **select** the **+ Create** button. (If AppDynamics is integrated with Active Directory, select the drop-down menu, and then select AppDynamics NOT LDAP).

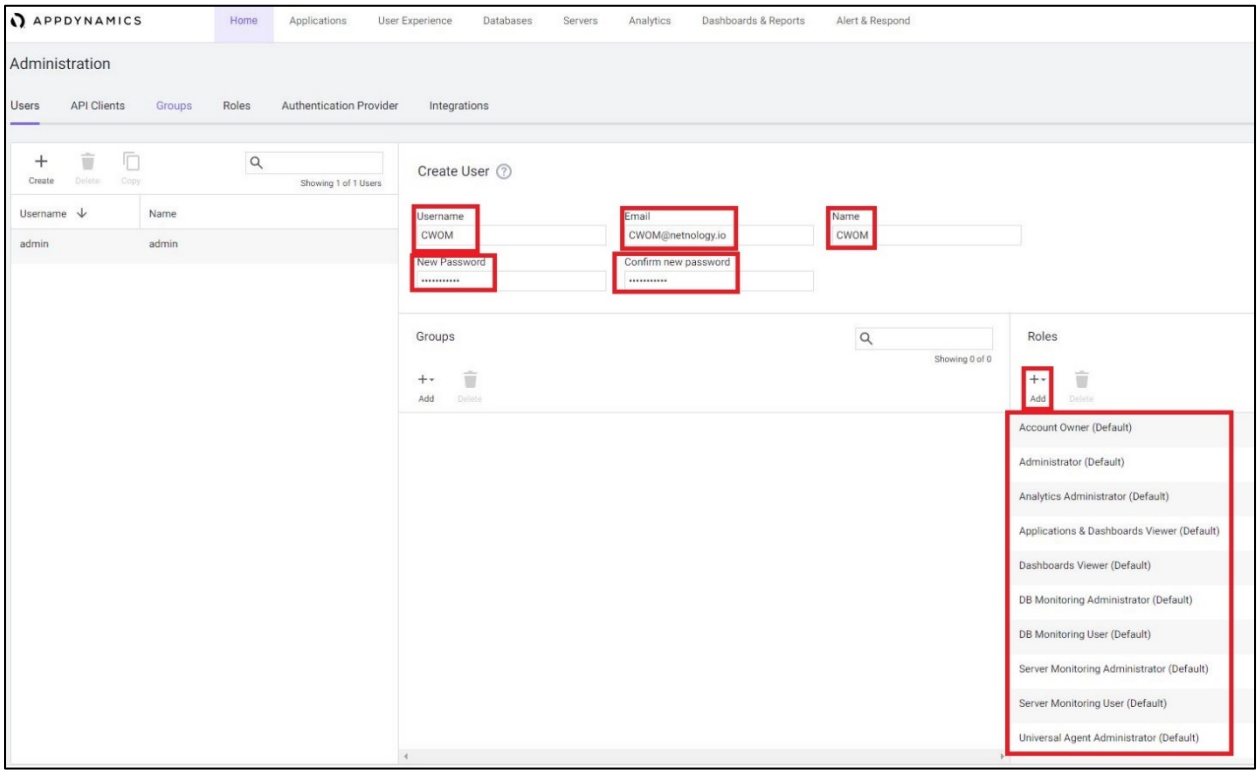


In the **'Create'** User section, fill in the required information for a user account. For this integration, the details are as follows:

1) Username: CWOM
2) Email: CWOM@netnology.io
3) Name: CWOM
4) New Password: SUPERSECRETPASSWORD

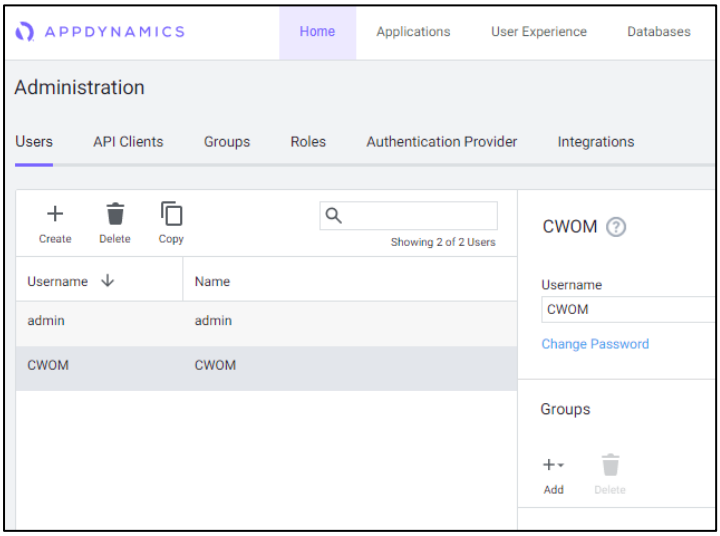
5) Confirm Password: SUPERSECRETPASSWORD

Roles for this account should also be added. For this integration, all roles were added but in production environment, only add the least privileges whenever possible. For example, if we only want IWO to read the information from AppDynamics and never make changes to resources, then Read Only would be needed. However, if we wanted IWO to perform actions then we need to give more than Read Only rights.

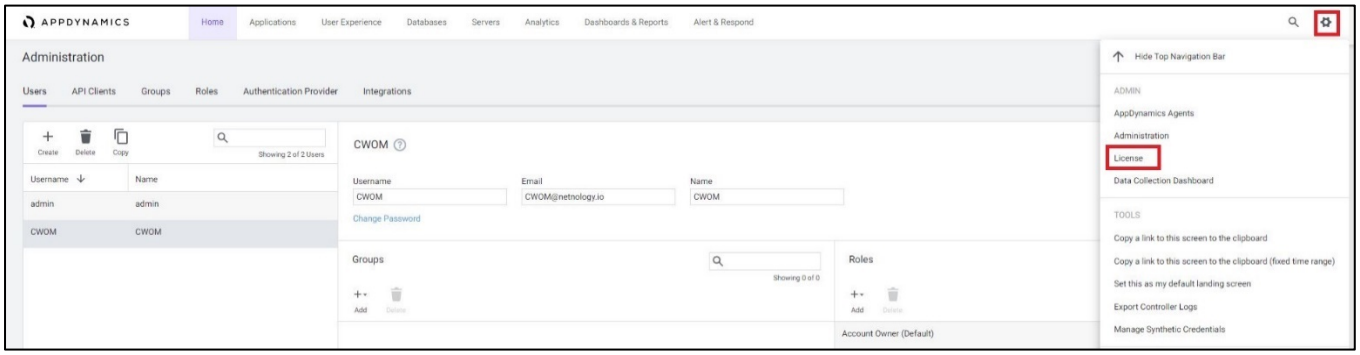


Click the '**Save'** button to create the new user.

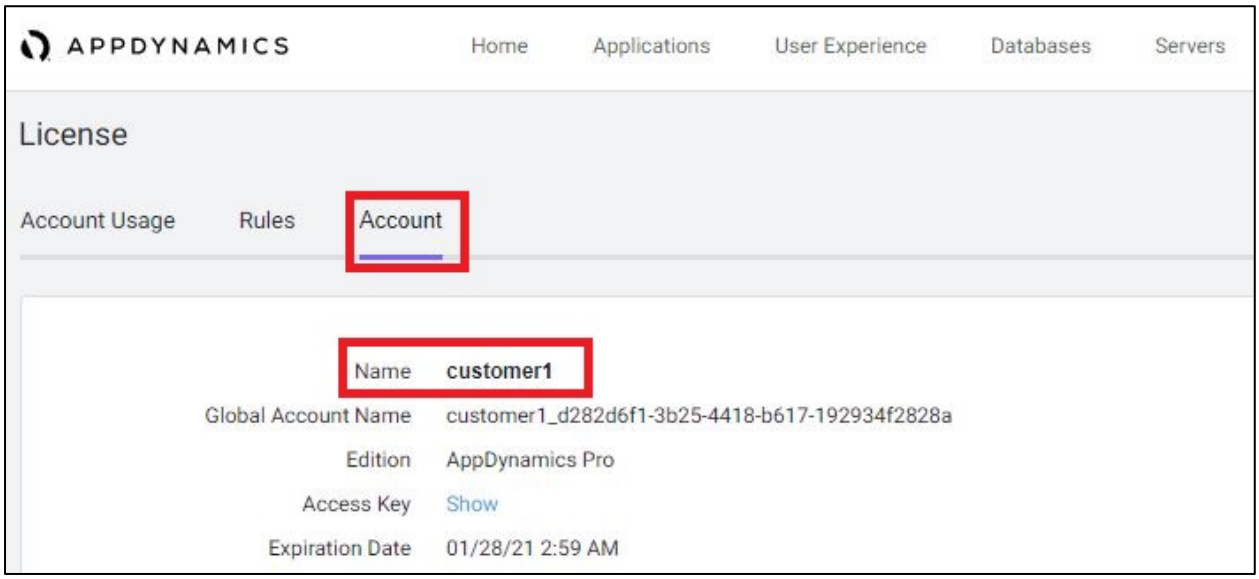
Verify the creation of the new under '**Users'** Tab.



Take note of the account name for your AppDynamics instance. This can be achieved by **clicking** on the '**Cog'** in the top right-hand corner of the application and **clicking** on '**licenses'**.



From the license section, **navigate** to the '**Accounts'** tab and here we can take note of the **Name:** section. For this integration, the Account Name is **customer1.**



NOTE: This will be used when creating the target in IWO in a later step.

3.3 - Intersight Assist VM on prem with Intersight

For this Integration there are a few steps we need to complete before moving forward.

1) Install an Intersight assist VM on prem with minimum size of Small
2) Integrate Intersight with the Intersight Assis VM on prem
3) Integrate AppD with IWO

Requirements:

Table 1: Intersight Assist Resource Requirements

Resource Requirement	System Requirements		
	Tiny	Small	Medium
vCPU	8	16	24
RAM (GiB)	16	32	64
Number of servers		2000	5000
Supported Hypervisors	VMware ESXi 6.5 and higher VMware vSphere Web Client 6.5 and higher		

This following table lists the system requirements to deploy Cisco Intersight Assist for Intersight Workload Optimizer

Table 2: Intersight Assist Resource Requirements for Workload Optimizer

Resource Requirement	System Requirements	
	Small	Medium
vCPU	16	24
RAM (GiB)	32	64
Storage (Disk in GiB)	500	500
Deploy Configuration	Up to 1000 Virtual Machines	Up to 30,000 Virtual Machines
Supported Hypervisors	VMware ESXi 6.5 and higher	

**To deploy up to 100,000 Virtual Machines, increase the vCPU to 32 and RAM to 96 GB or more.

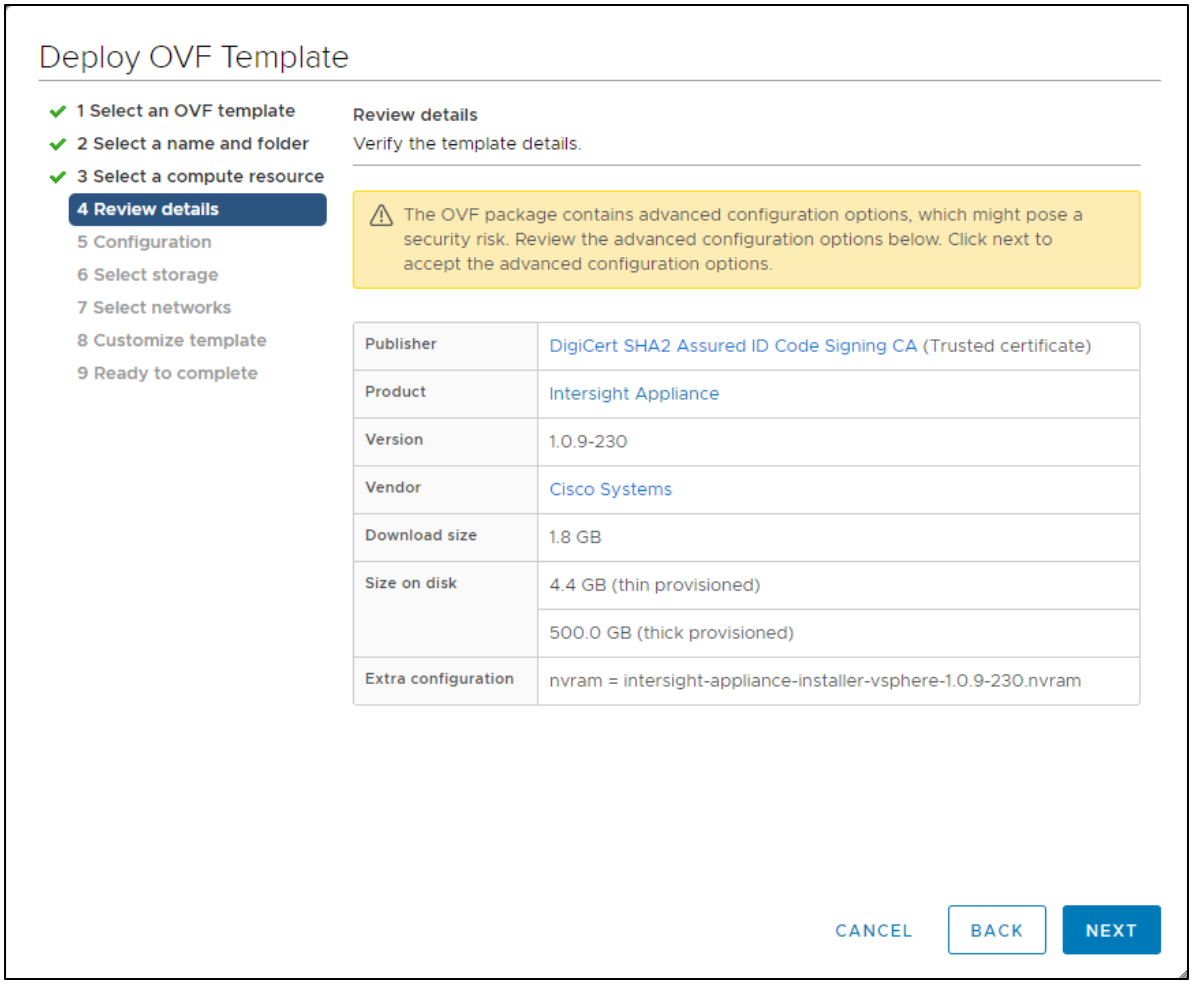
Assumptions:

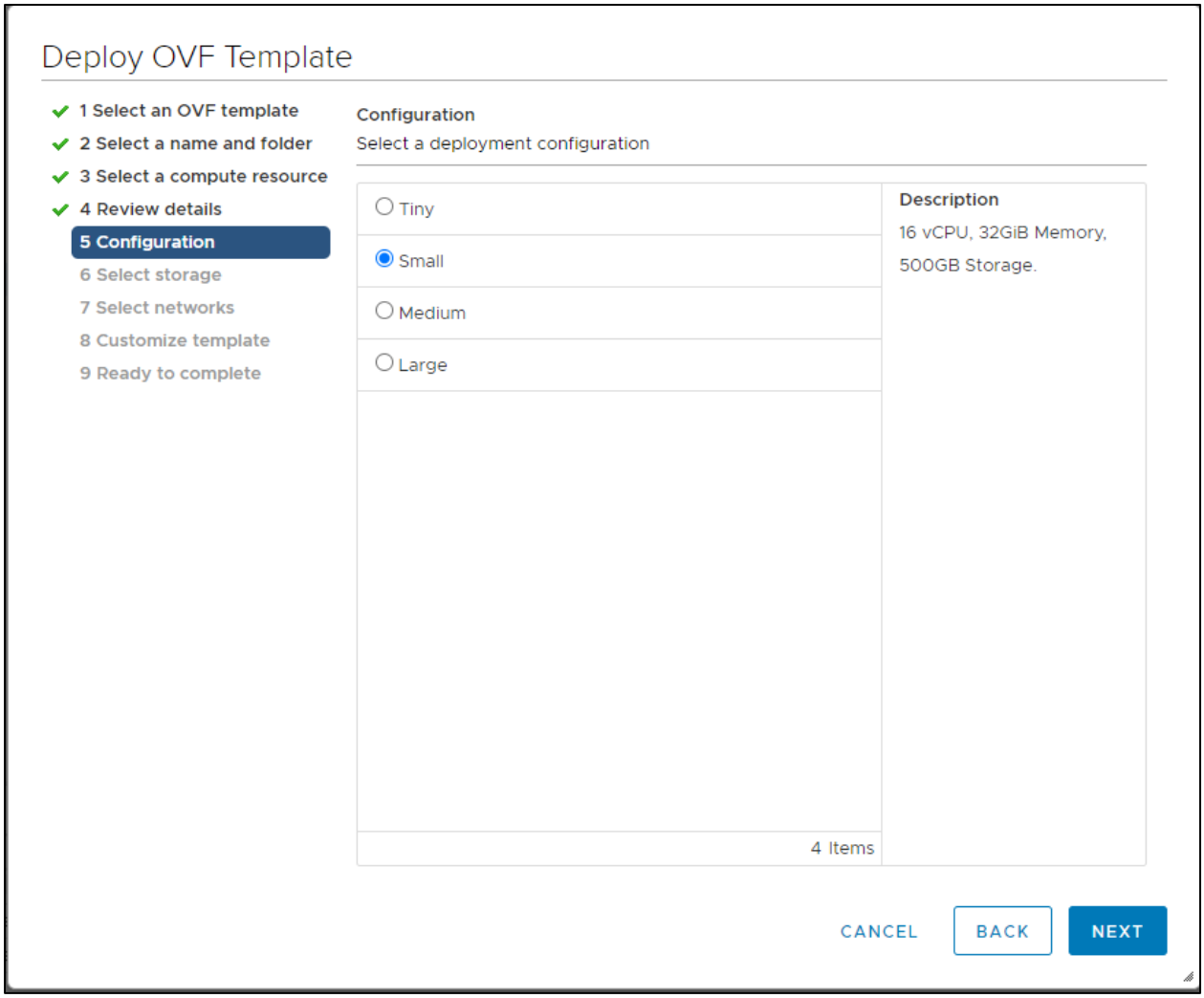
1) Already have Cisco Intersight assist OVA file, and uploaded to vCenter
2) DNS entries created for Intersight Assist and Pointer records (REQUIRED)
3) Already have an Intersight Instance running (cloud)
4) Have the proper license for Intersight Workload Optimizer (cloud)

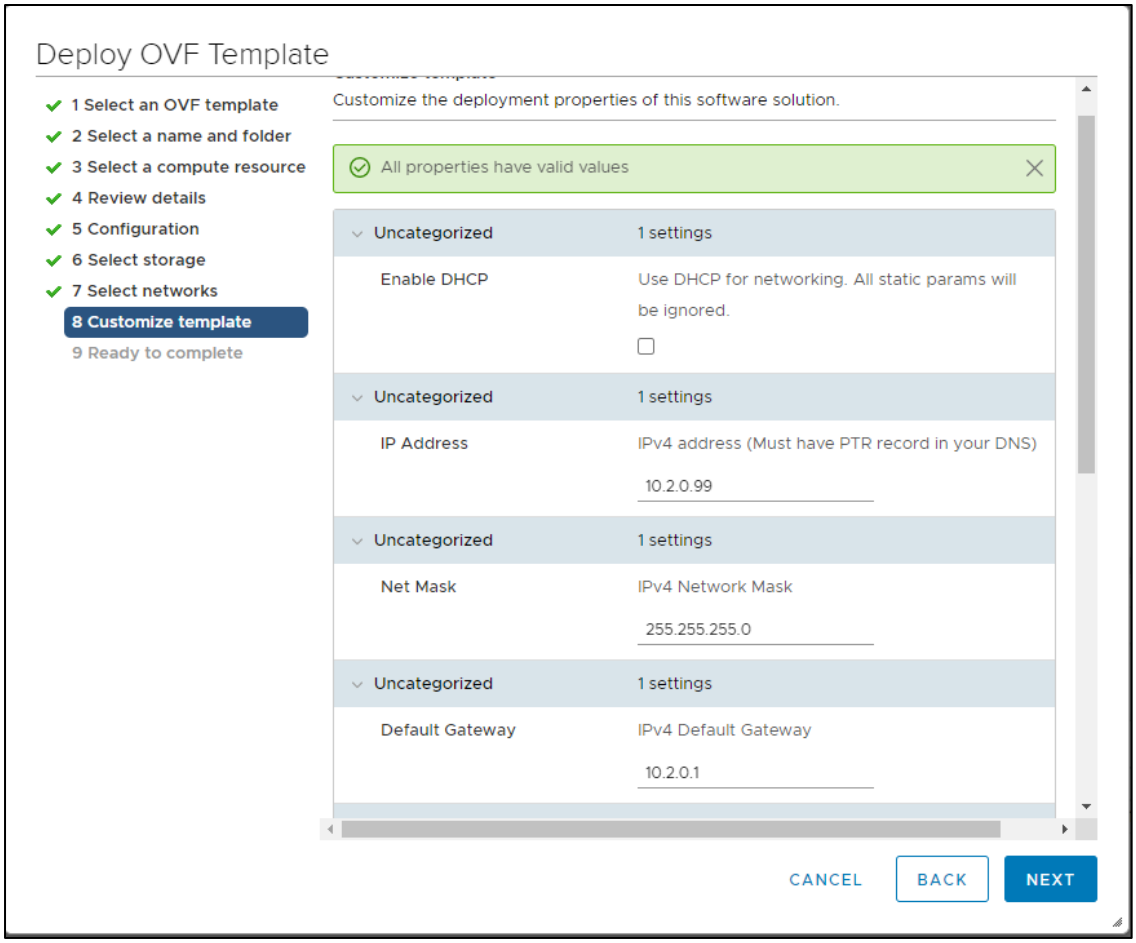
3.3.1 - Install/Integrate Intersight Assist VM with Intersight

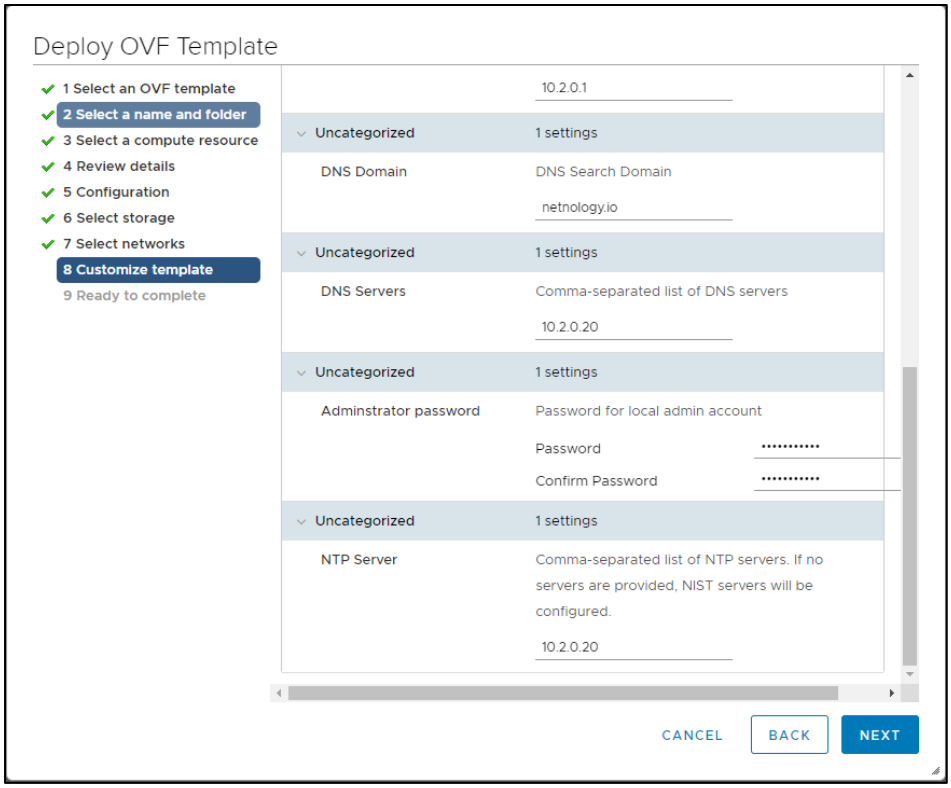
During the OVA upload process, we will have to select some configuration details for the appliance. In this demonstration for the Configuration size, we are just going to deploy the Small size. Then we will fill out the customize template options.

Screenshots are below of this OVA upload process and customization.


Deploy OVF Template
1 Select an OVF template
2 Select a name and folder
3 Select a compute resource
4 Review details
5 Configuration
6 Select storage
7 Select networks
8 Customize template
9 Ready to complete
Review details
Verify the template details.
The OVF package contains advanced configuration options, which might pose a security risk. Review the advanced configuration options below. Click next to accept the advanced configuration options.
Publisher
DigiCert SHA2 Assured ID Code Signing CA (Trusted certificate)
Product
Intersight Appliance
Version
1.0.9-230
Vendor
Cisco Systems
Download size
1.8 GB
Size on disk
4.4 GB (thin provisioned)
500.0 GB (thick provisioned)
Extra configuration
nvram = intersight-appliance-installer-vsphere-1.0.9-230.nvram
CANCEL
BACK
NEXT


Deploy OVF Template
1 Select an OVF template
2 Select a name and folder
3 Select a compute resource
4 Review details
5 Configuration
6 Select storage
7 Select networks
8 Customize template
9 Ready to complete
Configuration
Select a deployment configuration
Tiny
Small
Medium
Large
4 Items
Description
16 vCPU, 32GiB Memory, 500GB Storage.
CANCEL
BACK
NEXT


Deploy OVF Template
1 Select an OVF template
2 Select a name and folder
3 Select a compute resource
4 Review details
5 Configuration
6 Select storage
7 Select networks
8 Customize template
9 Ready to complete
Customize the deployment properties of this software solution.
All properties have valid values
Uncategorized
1 settings
Enable DHCP
Use DHCP for networking. All static params will be ignored.
Uncategorized
1 settings
IP Address
IPv4 address (Must have PTR record in your DNS)
10.2.0.99
Uncategorized
1 settings
Net Mask
IPv4 Network Mask
255.255.255.0
Uncategorized
1 settings
Default Gateway
IPv4 Default Gateway
10.2.0.1
CANCEL
BACK
NEXT



Once the OVA is finished being deployed and all of the pre-requisites are fulfilled then we can proceed to power on the Intersight Assist VM on running through the first-time setup process.

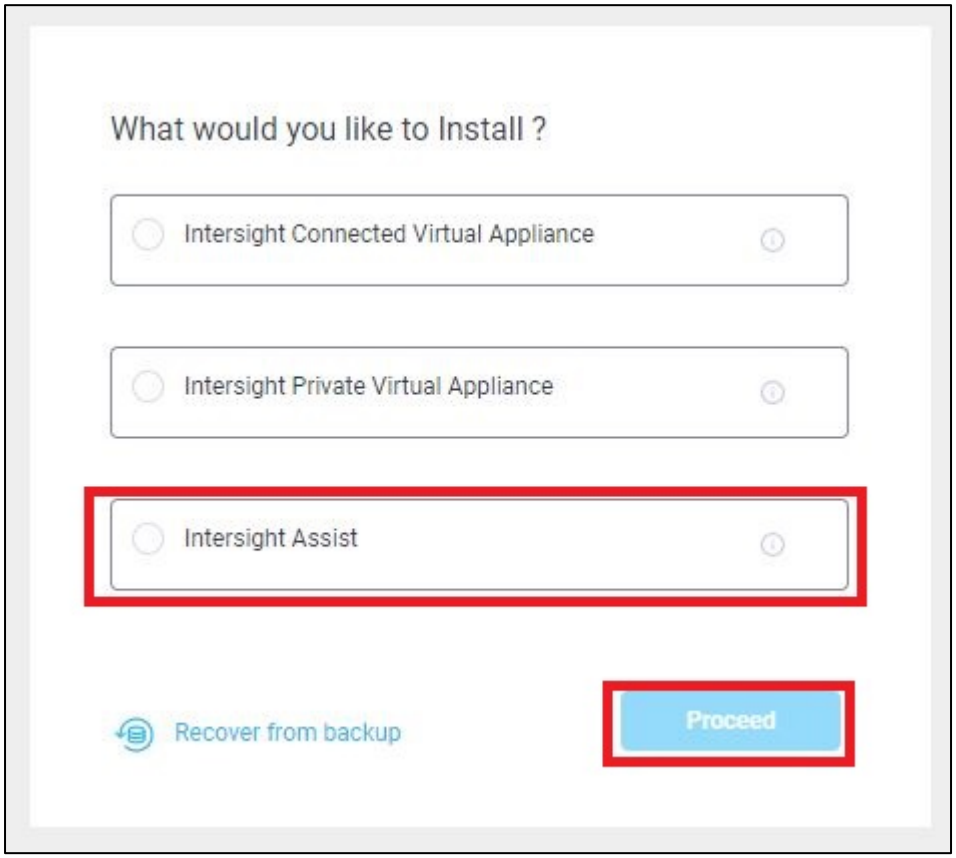
Now on the Console as the VM is booting up we can see the following.

```
################################## WARNING!!! ##########################################
################### READ THIS BEFORE ATTEMPTING TO LOGON ###############################
#                                                                                      #
#   This System is for the use of authorized users only.  Individuals                  #
#   using this computer without authority, or in excess of their                       #
#   authority, are subject to having all of their activities on this                   #
#   system monitored and recorded by system personnel.  In the course                  #
#   of monitoring individuals improperly using this system, or in the                  #
#   course of system maintenance, the activities of authorized users                   #
#   may also be monitored.  Anyone using this system expressly                         #
#   consents to such monitoring and is advised that if such                            #
#   monitoring reveals possible criminal activity, system personnel                    #
#   may provide the evidence of such monitoring to law enforcement                     #
#   officials.  You cannot copy, disclose, display or otherwise                        #
#   communicate the contents of this server except to other Cisco                      #
#   employees who have been authorized to access this server.                          #
#                                                                                      #
########################## Cisco Confidential Information ##############################
intersight2 login:
```

On the CLI once we get to the Login prompt, we are good to go.

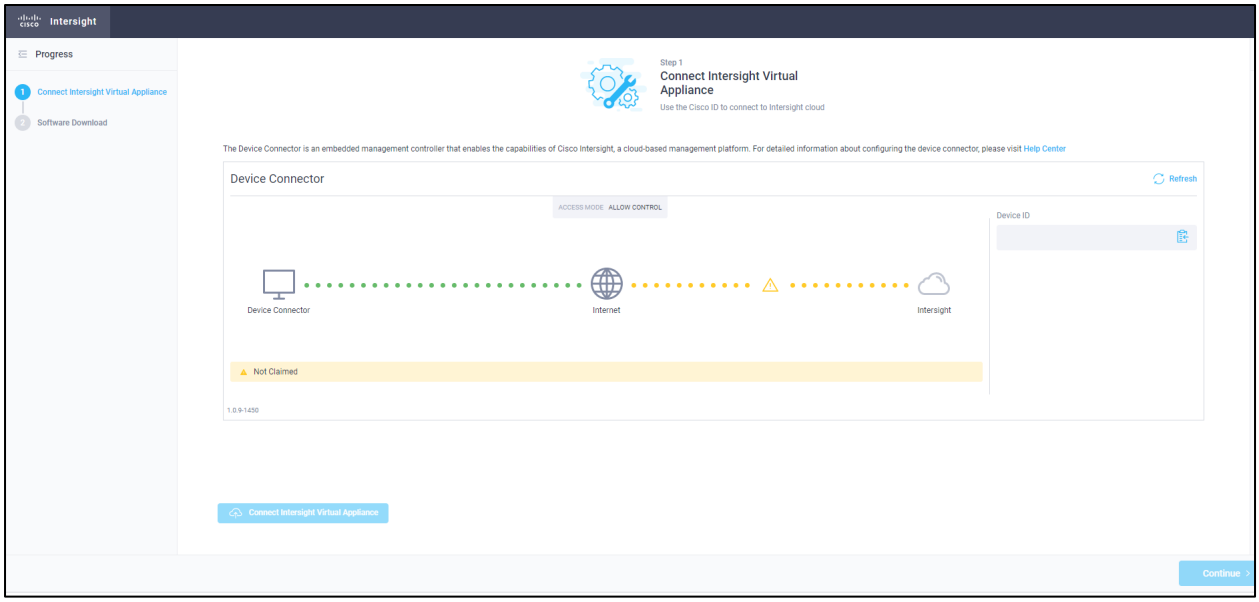
So, we need to browse to our DNS record name. in this demonstration we will browse to http://intersight2/

When we browse to the address above, we instantly see the following screen.



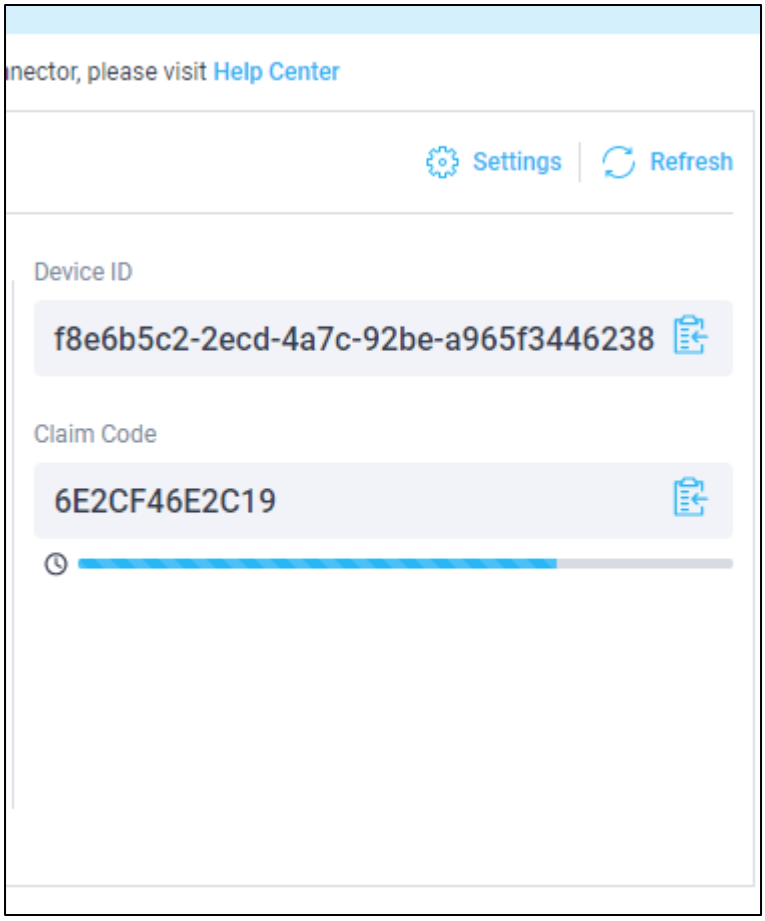
For this we need to select the **Intersight Assist** option then hit proceed.

Next, we will come directly to the Connect Intersight Virtual Appliance screen



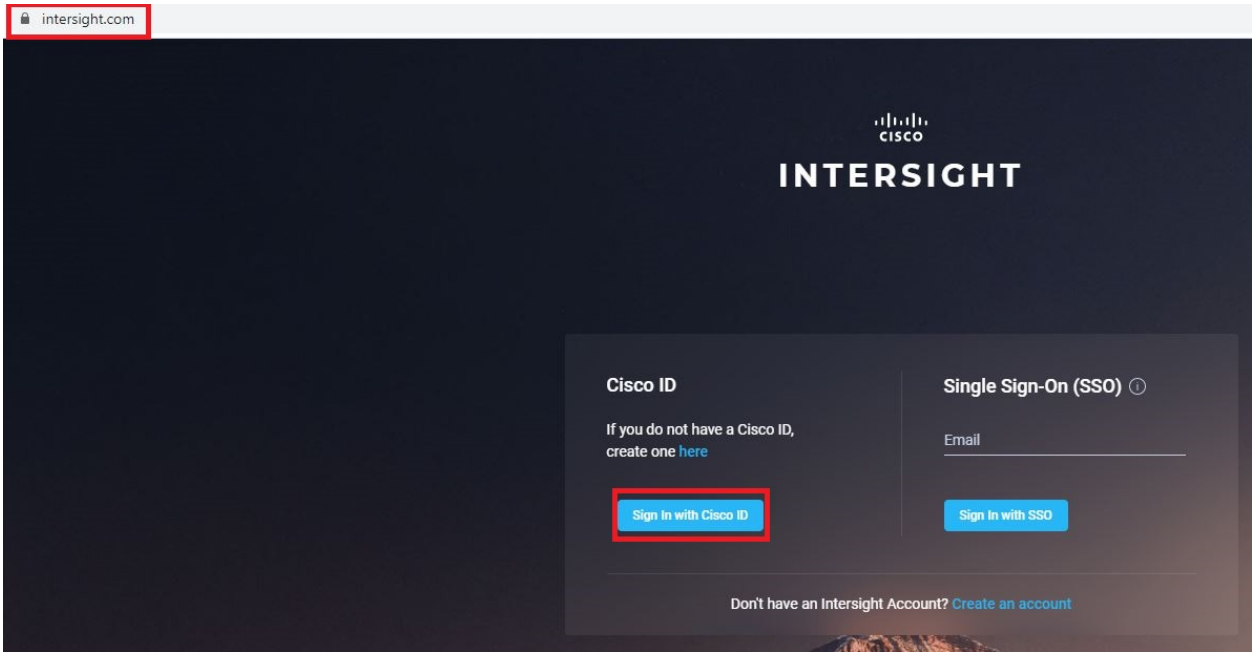
From this screen we need to wait just a couple minutes until the Device IS and Claim Code display on the right-hand side of the screen.

*Note: this claim code has a time out and will change every couple of minutes.

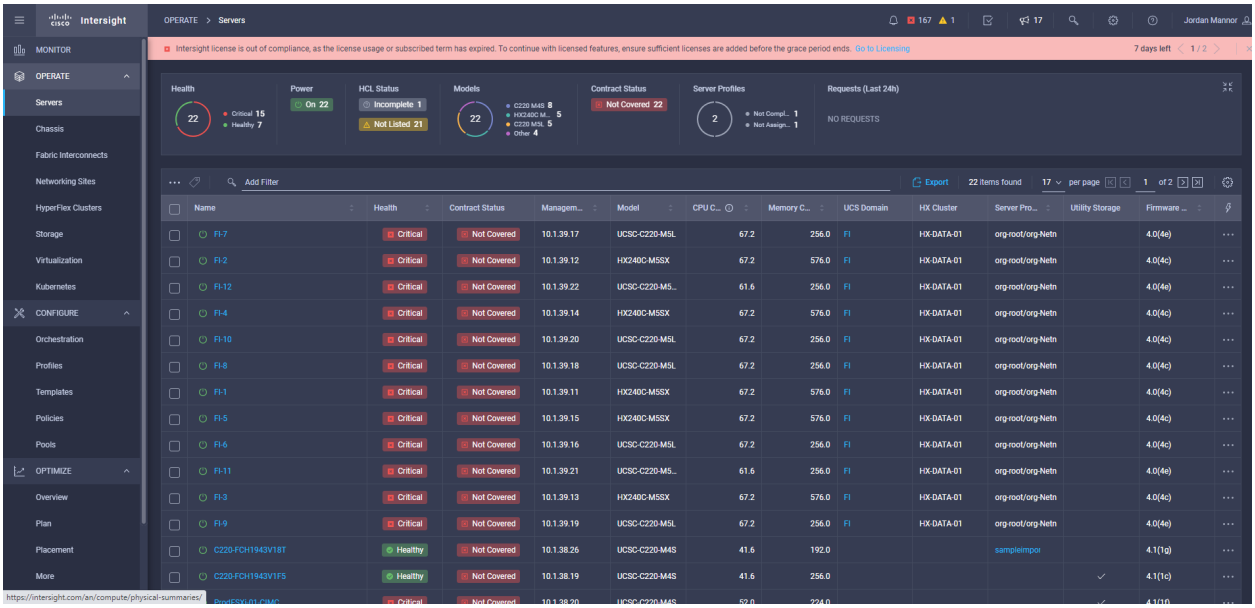


With the information collected from the Intersight Assist VM, we now need to jump into our Intersight appliance.

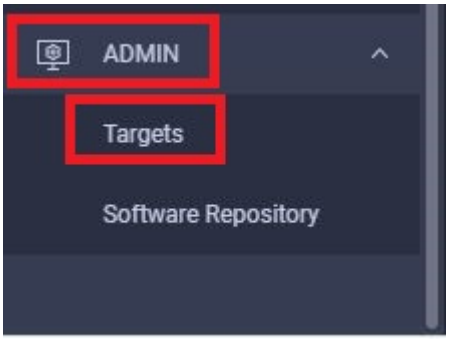
We can simply browse to www.intersight.com and click on Sign in with Cisco ID.



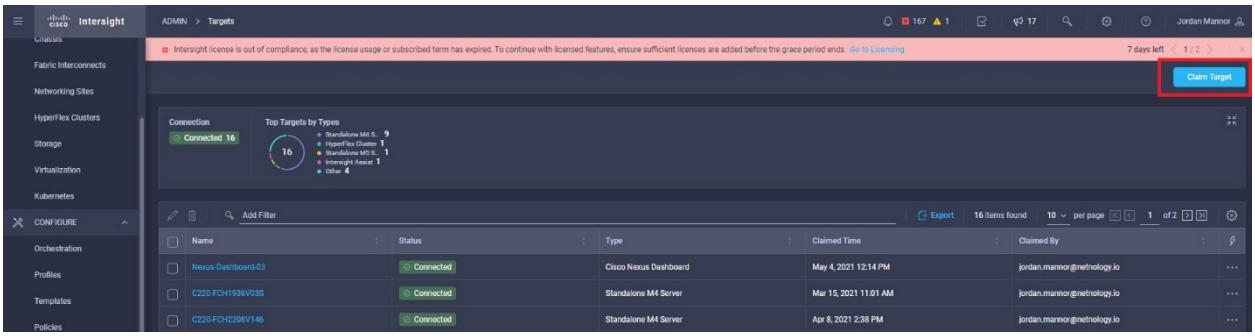
Once logged in, we are taken to the main dashboard of Intersight. For this demonstration, the dashboard should look like the screen below.



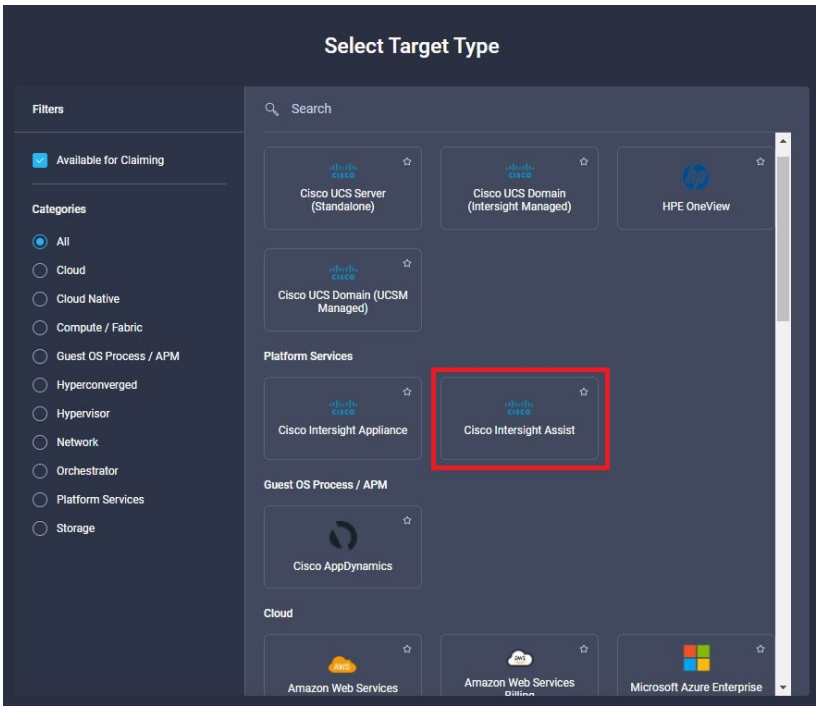
From the main dashboard we need to navigate to the bottom of the left-hand pane to the ADMIN section and click on **Targets**.



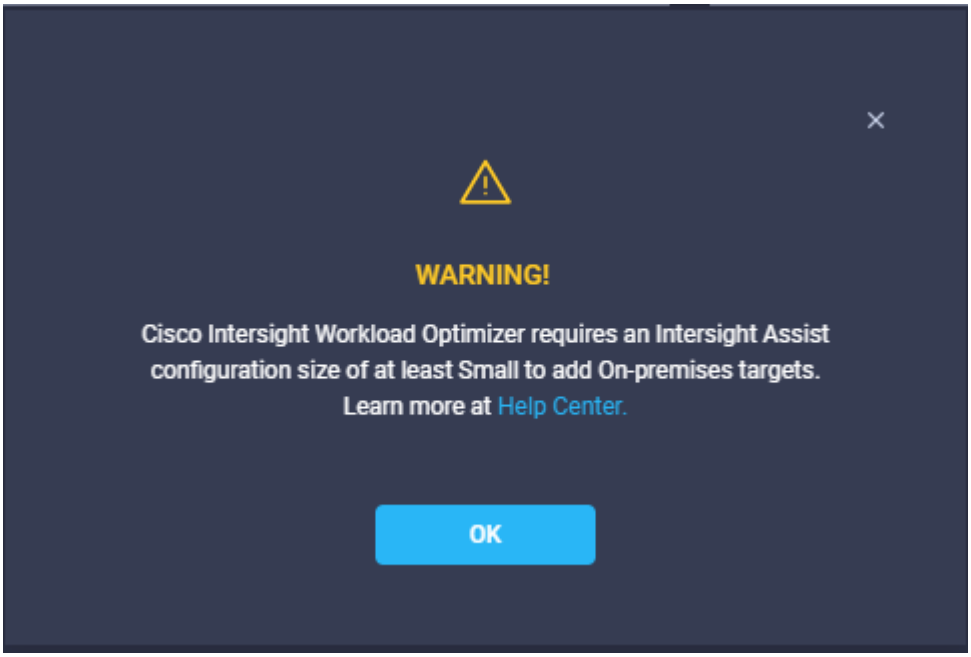
From the Targets screen, because we already have targets configured, we need to click on the **Claim Target** button in the upper right-hand corner of the window. If this is your first target from Intersight, this option will be displayed in the center of the screen.



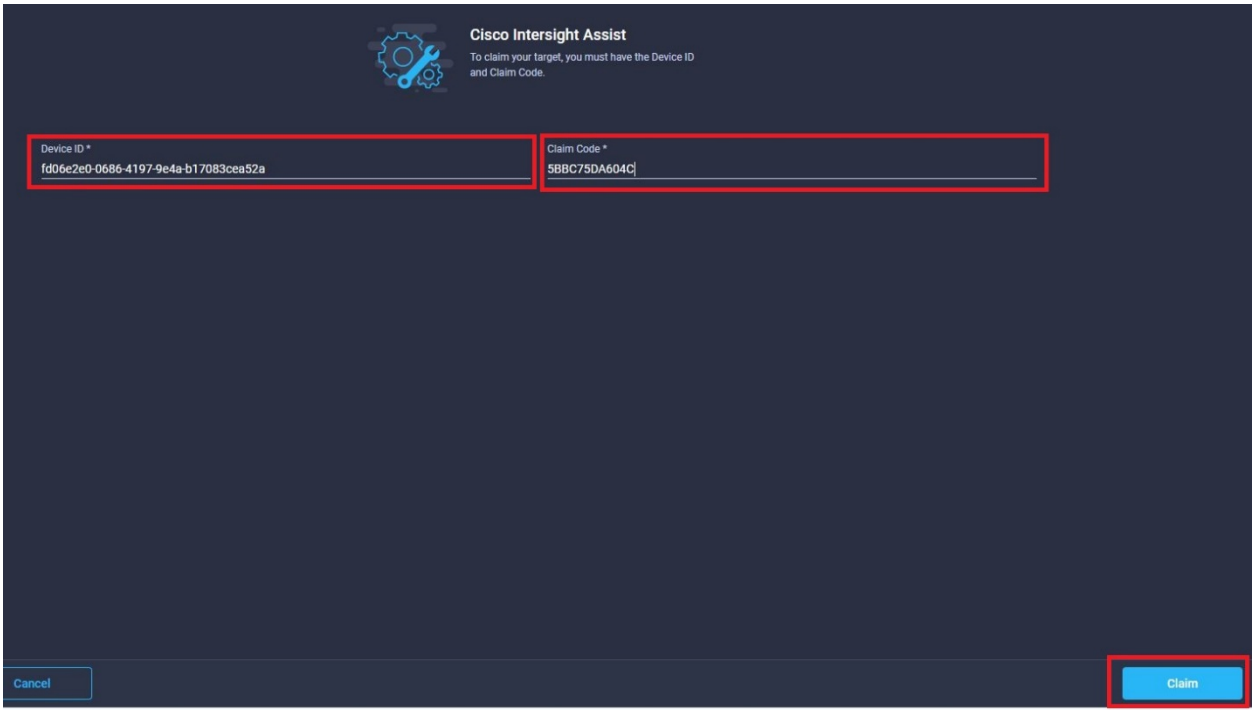
From the **Select Target Type** screen we can then click on **Cisco Intersight Assist** button then hit start in the lower right-hand corner of the window.



From here a warning screen will pop up saying that requirement size to use IWO requires Small setup. But we can go ahead and hit ok on this screen to proceed.

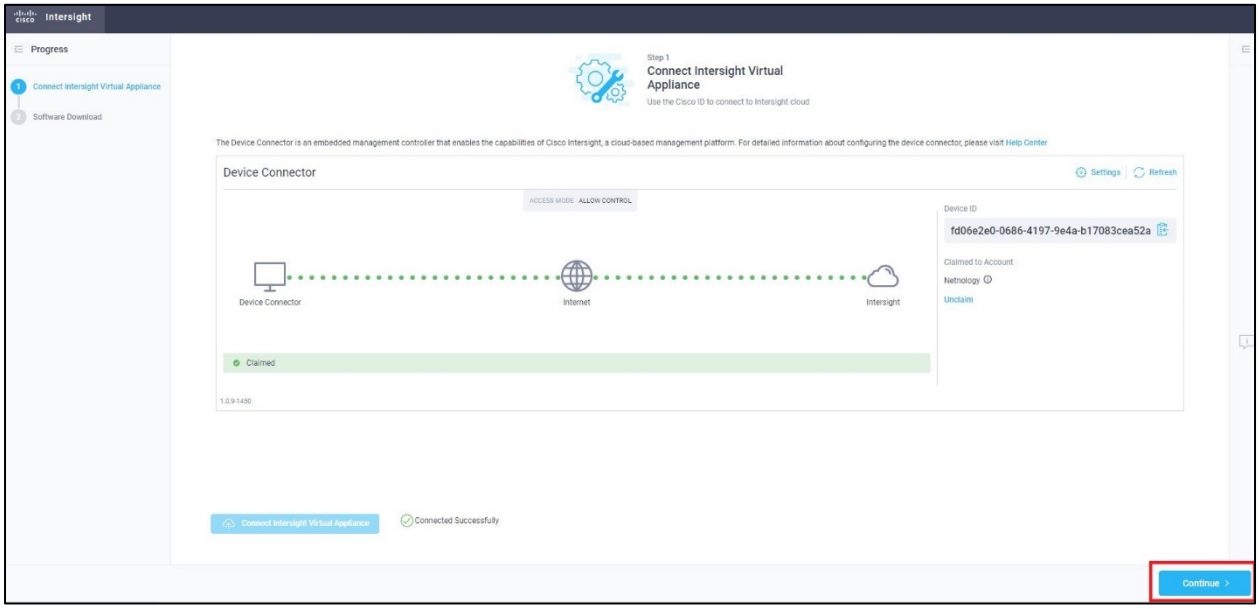


Next, we can fill in the **Device ID** and **Claim Code** from the Cisco Intersight Assist VM screen, then hit **Claim**.

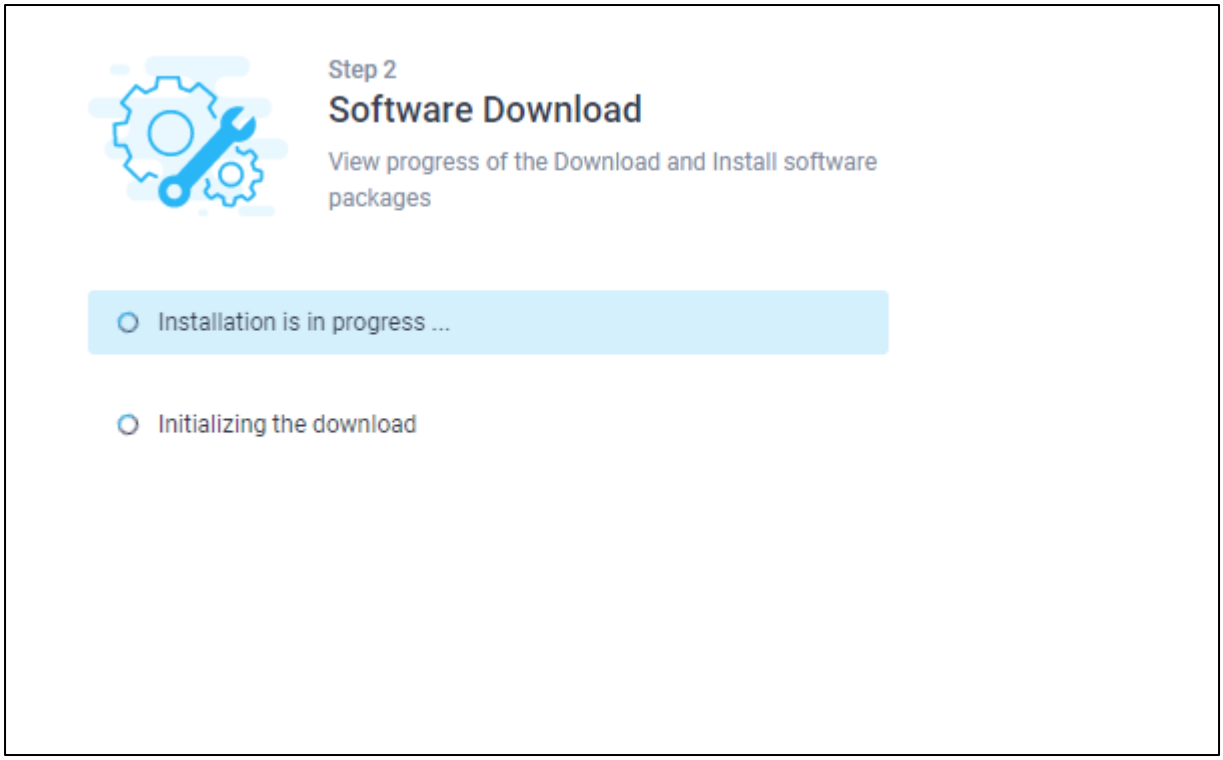


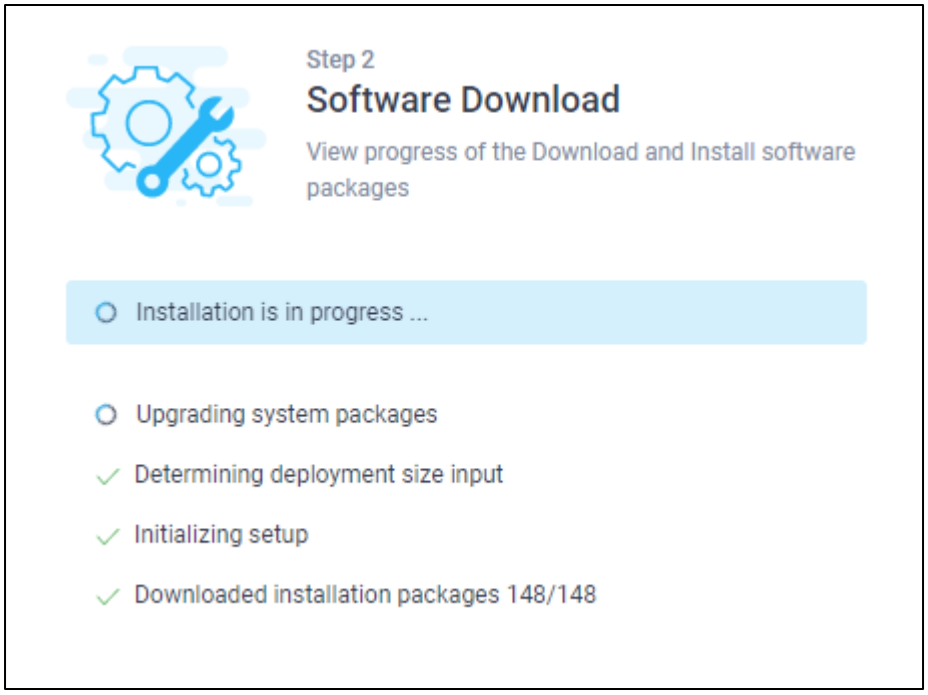
After hitting **Claim**, we can monitor both Intersight and the Intersight Assist VM.

We can then move to the Intersight Assist VM and click continue.

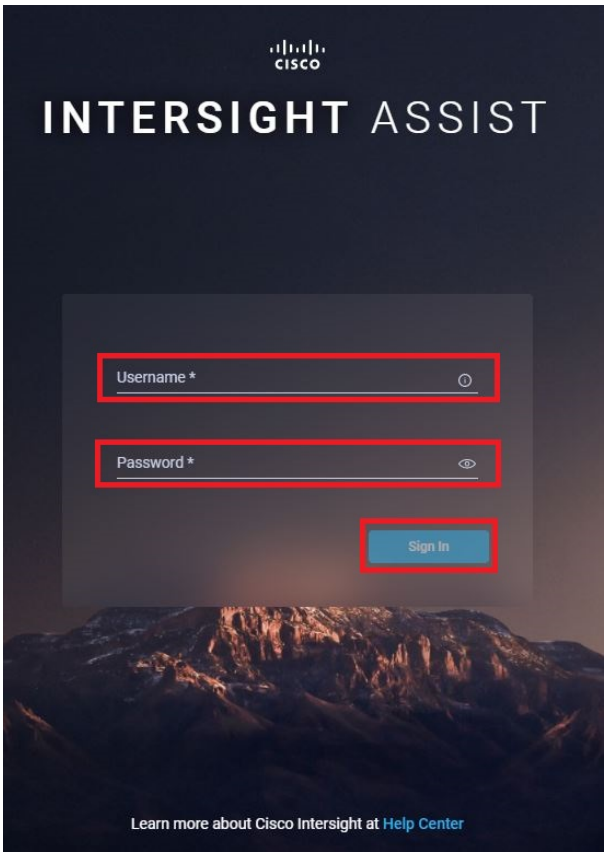


After hitting continue, the Intersight Assist VM will begin the installation and download the required packages.

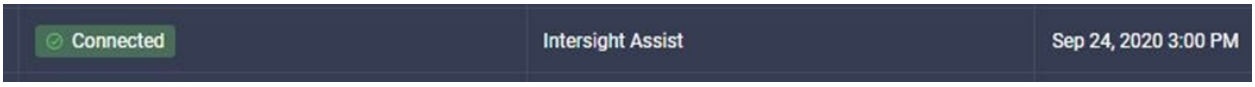




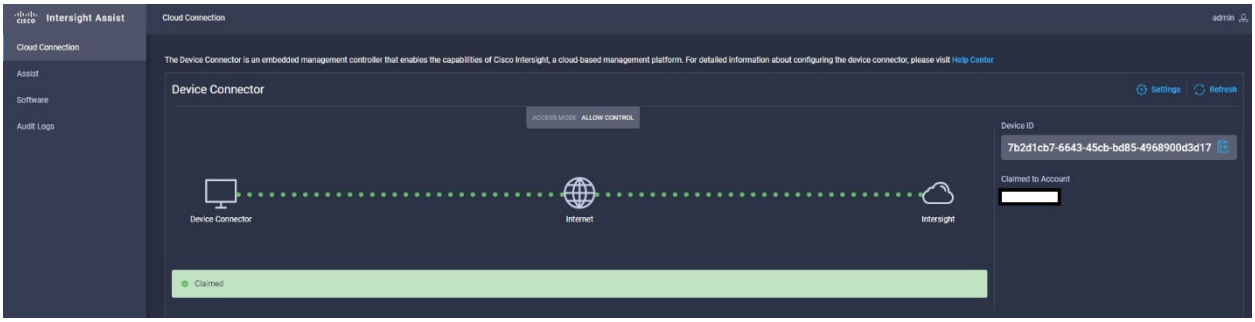
After some time, the Intersight Assist VM setup will finish and take you to the login screen.



We can also notice from Intersight that a device is connected.



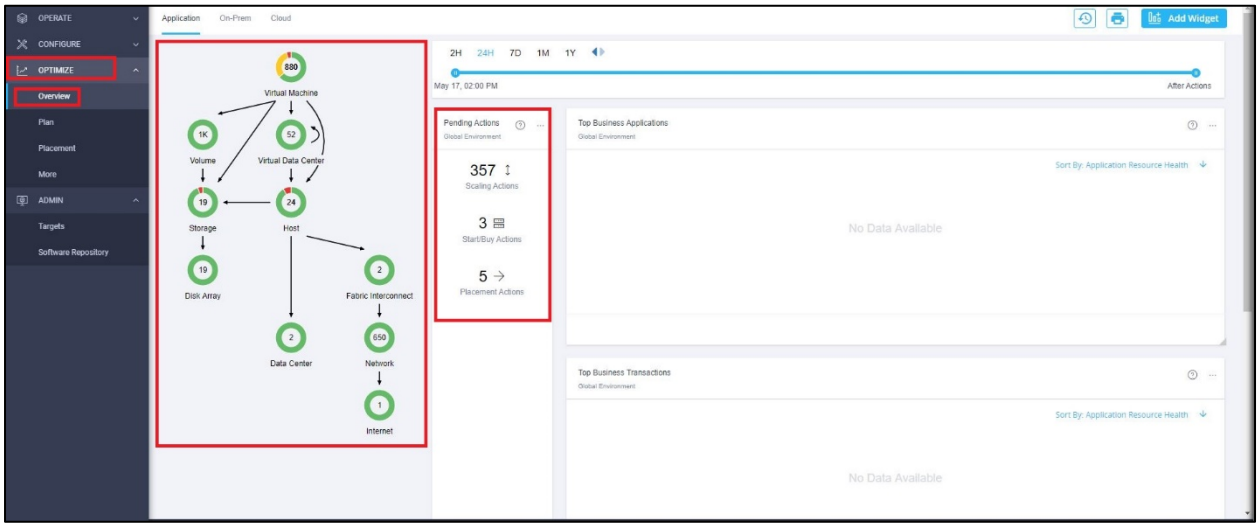
From the Intersight Assist VM, after logging in, we can see the device has been claimed by Intersight.



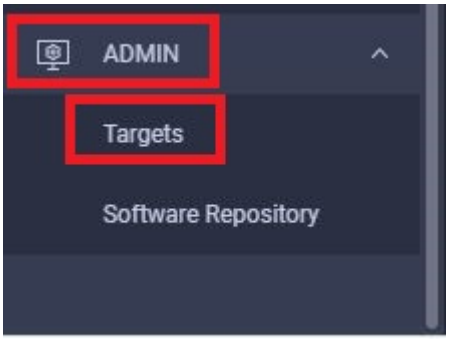
This is all we need to complete to start using IWO with Targets just as CWOM.

3.4 - IWO Target Configuration for AppDynamics

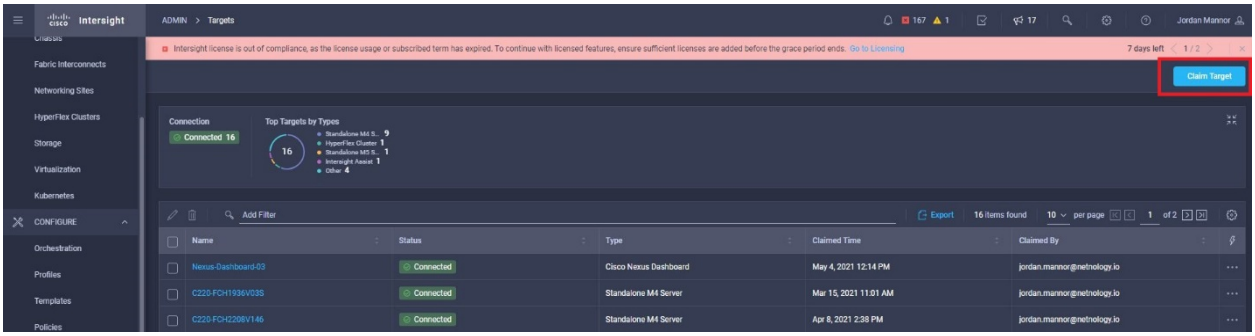
We can start from Intersight and navigate to the **Optimize** section and click on the **Overview** section. In this demonstration we already have a few targets configured. But we can take note of the IWO overview tab.



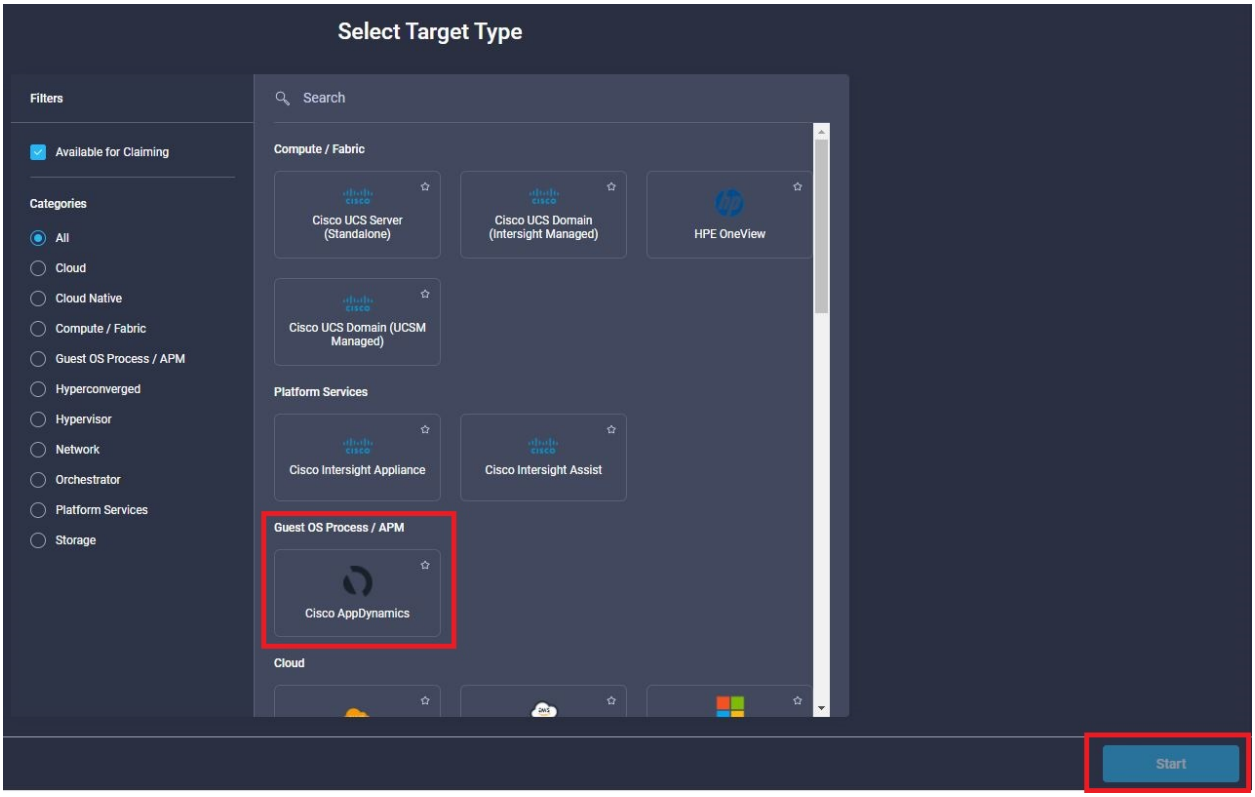
Now we can continue to the target addition of AppDynamics. So, we need to navigate to the ADMIN section and click on Targets again.



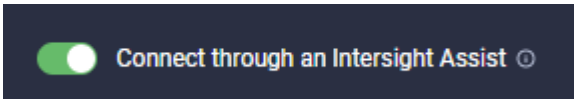
From here we are going to click on **Claim Target** again in the upper right-hand corner of the window.



On the Select Target type window we are going to select Cisco AppDynamics this time and then click **Start**.

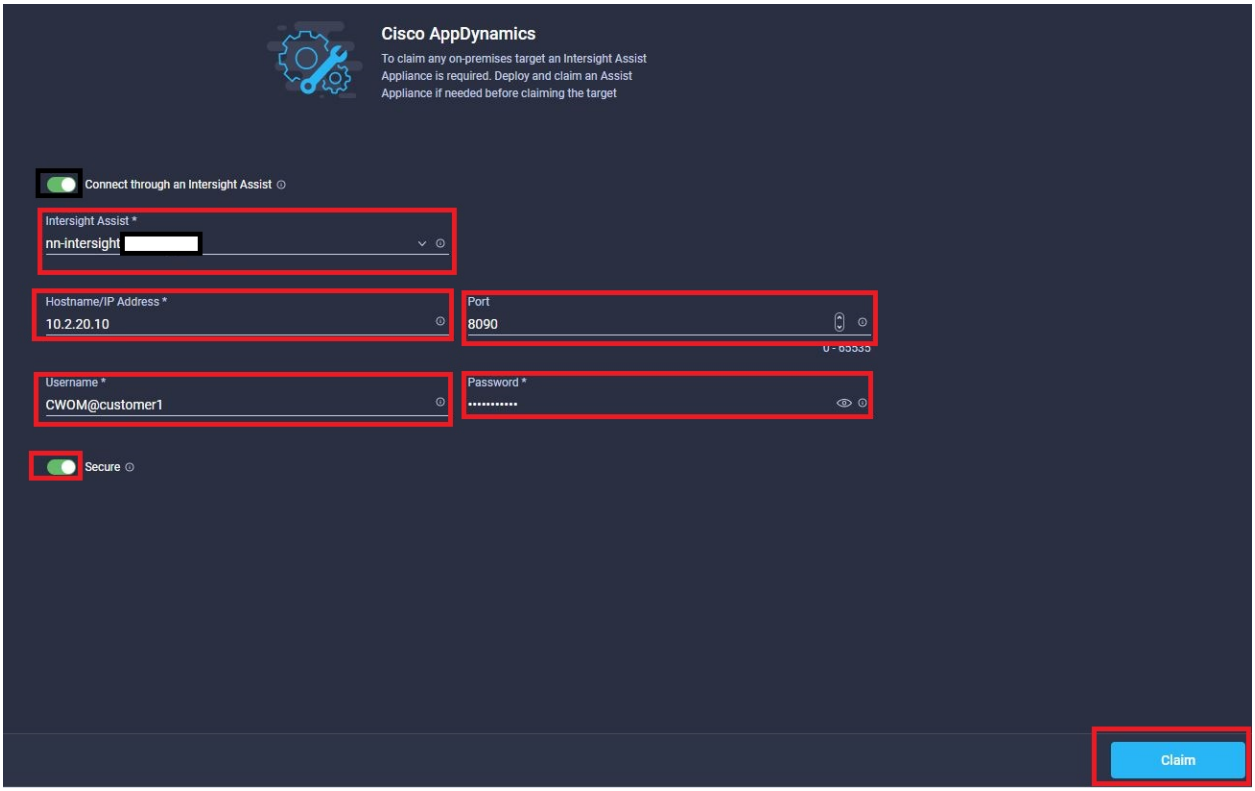


From the next screen we have 2 options, because AppDynamics can be on prem or a SaaS instance. For this demonstration we are going to connect to AppDynamics using a Intersight Assist VM because this AppDynamics instance is deployed on prem.



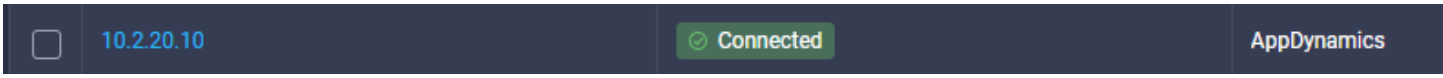
We also need to fill out the required information like Hostname/IP Address, Port, Username/Password. Then hit **Claim**.

*For this demonstration, we are going to use the same account CWOM that was created to connect to AppD from IWO.



After hitting **Claim**, we can instantly see that the claim of AppDynamics is in progress.

After a few mins we can see that the AppDynamics target is now claimed from the Targets screen on Intersight.



4 - AppDynamics and TE Integration

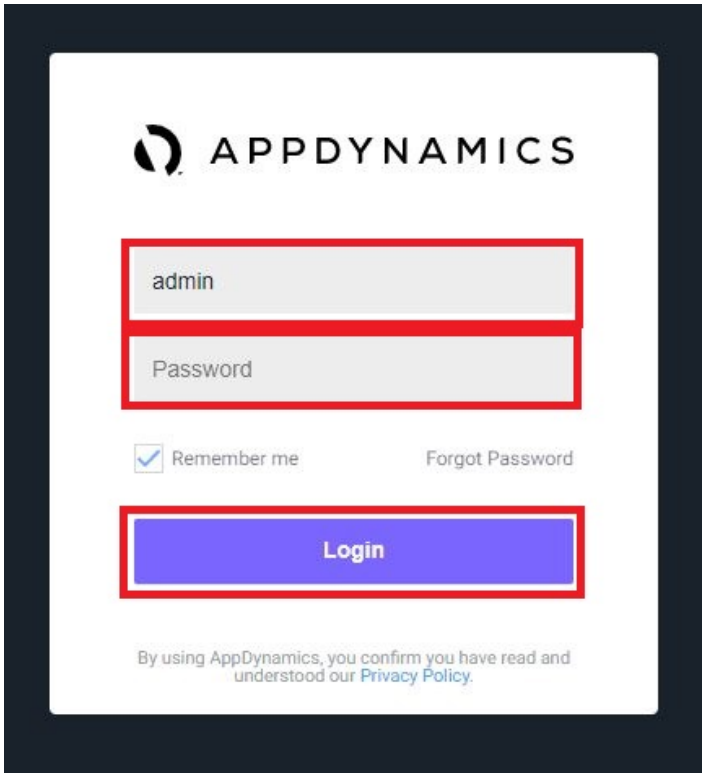
There are a few key components which are needed for this integration:

- AppDynamics Instance
- AppDynamics Application Name
- AppDynamics Username and Password
- AppDynamics Customer Name

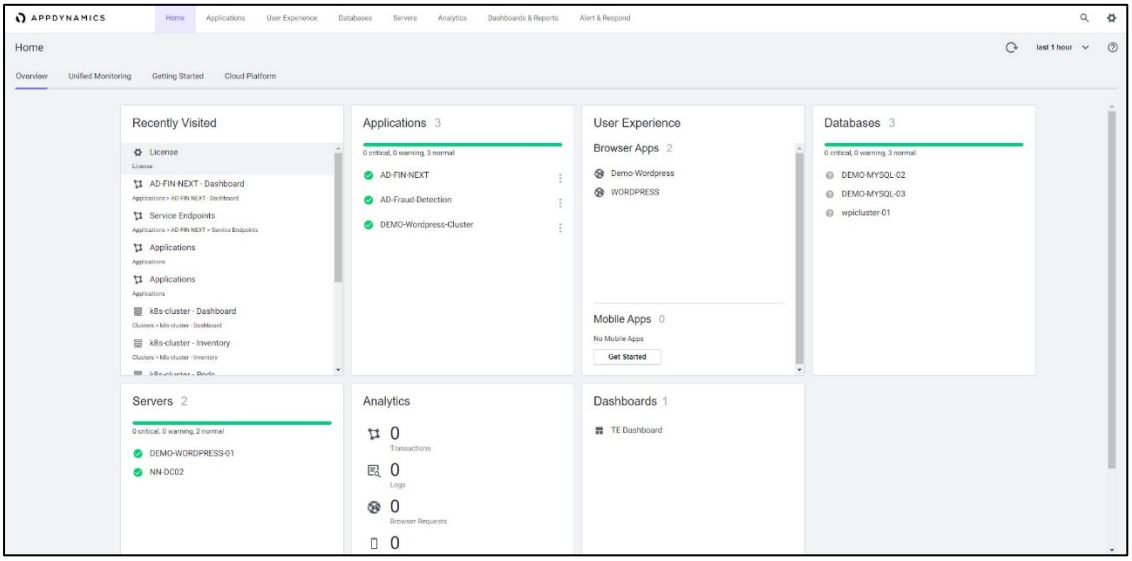
4.1 - AppDynamics Instance

This section describes the information required to be gathered on the AppDynamics dashboard in order to integrate AppDynamics with TE.

Browse to the on-prem AppDynamics instance and **Login**.



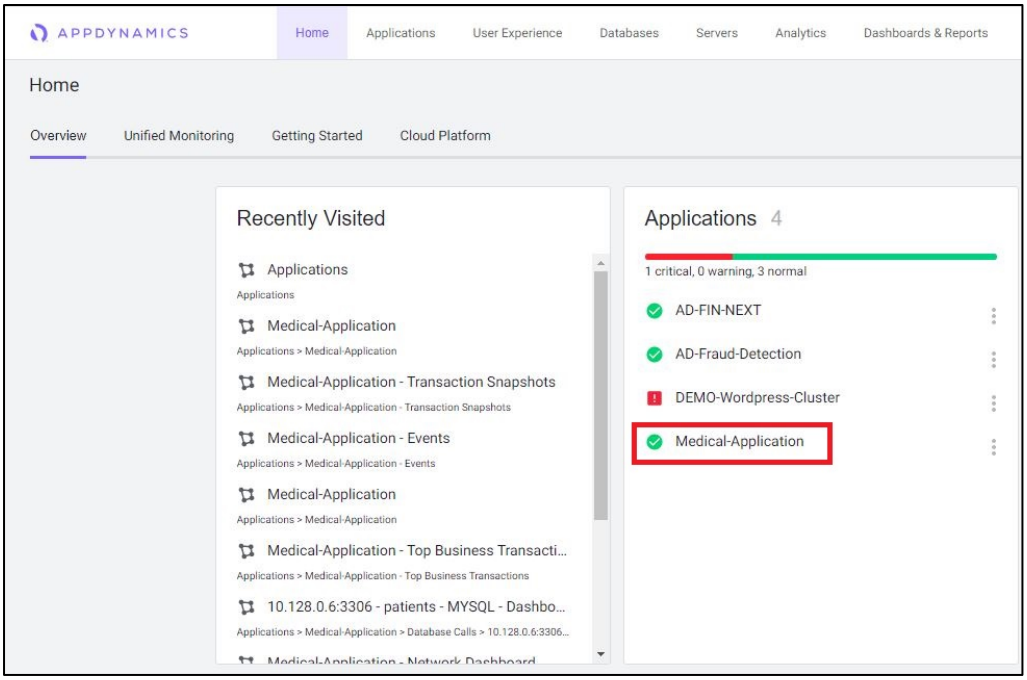
Once logged in, the main AppDynamics Dashboard can be viewed.



At this step, simply gather the information from the URL. This information will be used to point to the AppDynamics instance from the TE dashboard. Since this is an on-prem AppDynamics instance, firewall is configured to translate the private IP address to a public IP address, however the URL for the port 8090 is needed (AppDynamics default).

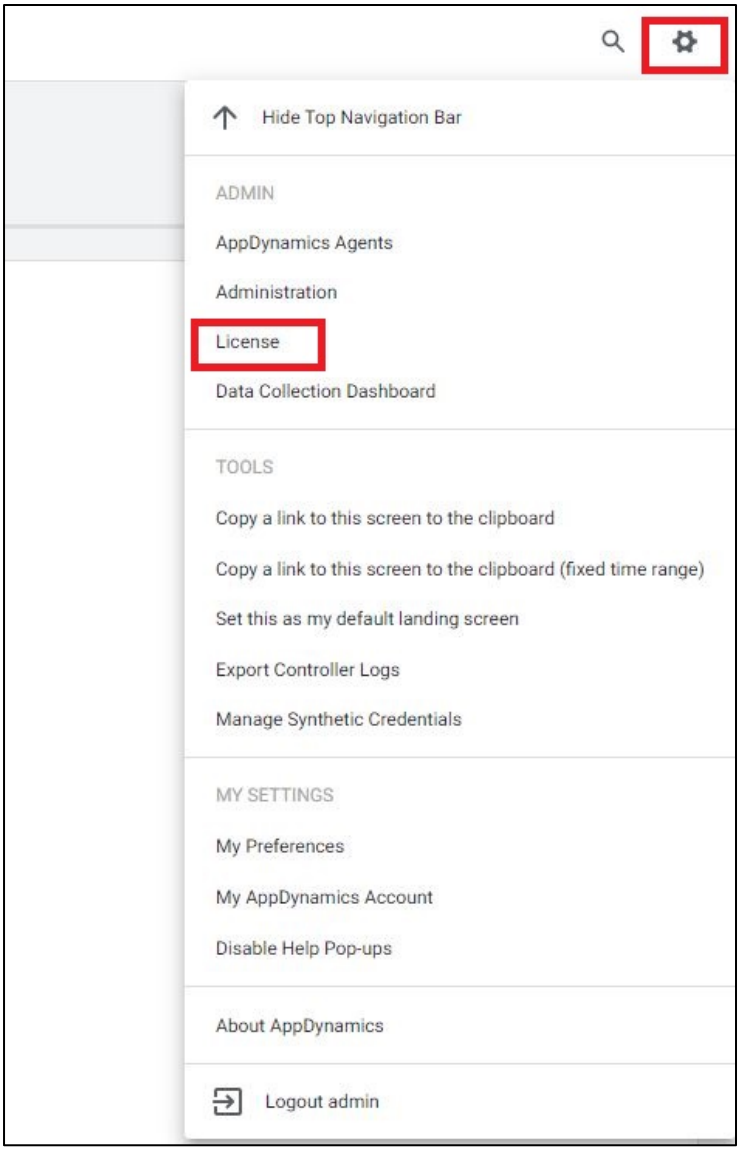
```
appd-controller:8090/controller/#/l
```

Next, application name **(Medical-Application)** is added which is being monitored by AppDynamics. This information is found on the main AppDynamics Dashboard.

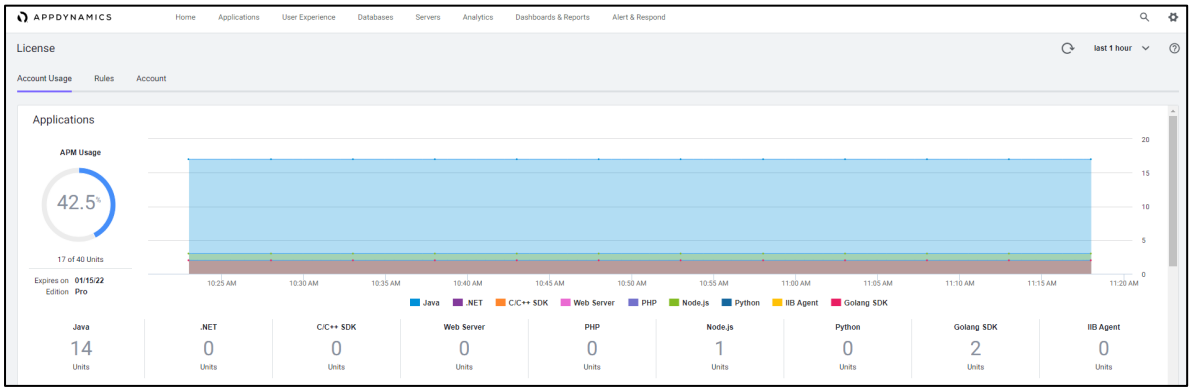


Next, the Account name is needed. To get this information, browse to the **License** settings.

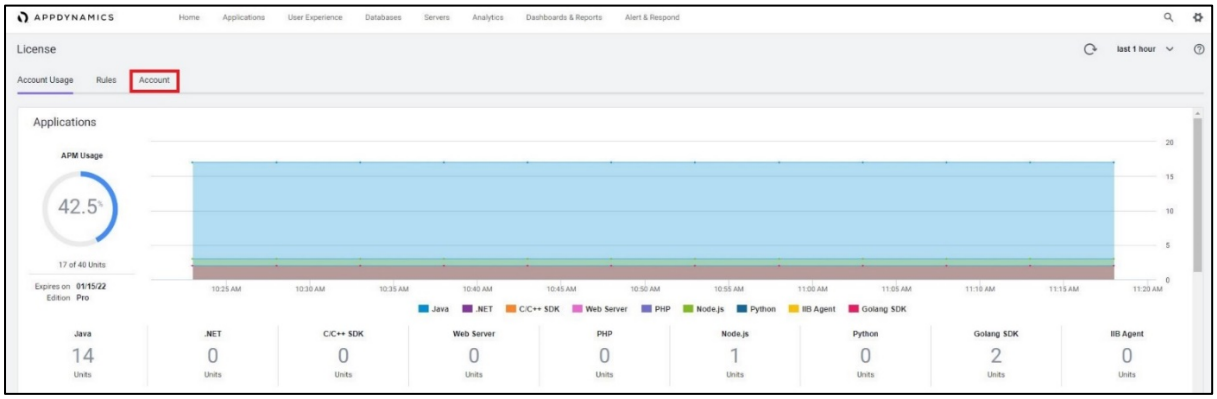
To do this from the main AppDynamics Dashboard, go to the upper right-hand corner of the dashboard and click on the **Cog** icon. Then, scroll down and click on the **License** section.



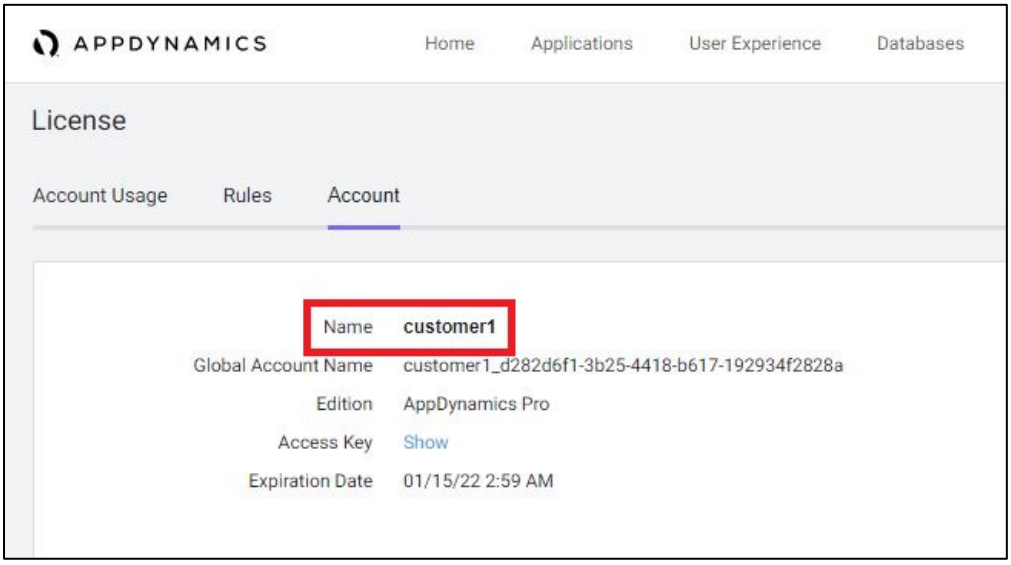
After clicking on the **License** section, the License **Account Usage** page will appear.



From here, click on the **Account** tab to view the customer information.



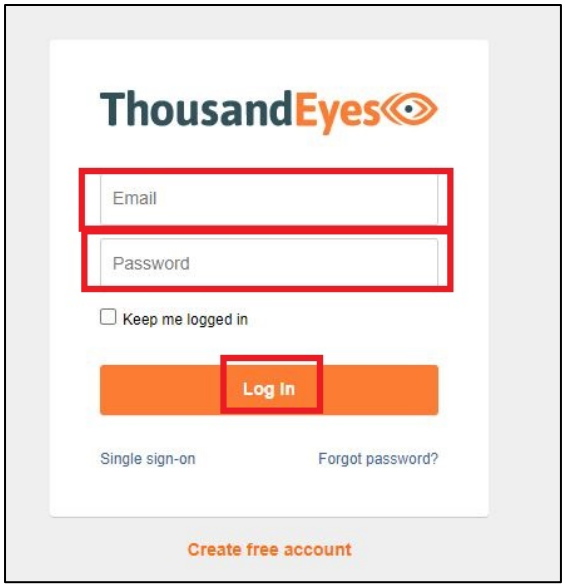
From the License Account page, the **Name customer1** can be seen. This is the last piece of information needed for the AppDynamics and TE integration.



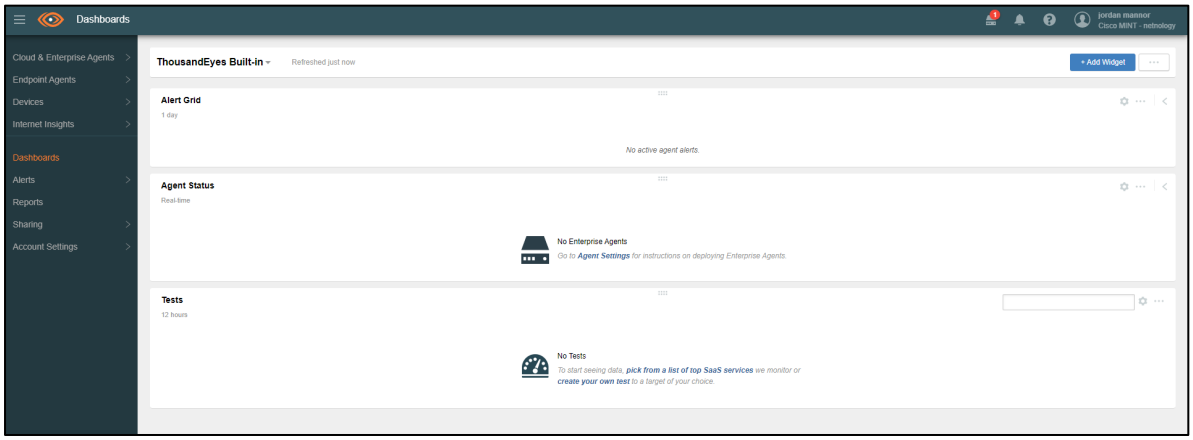
4.2 - Integration

This section describes the AppDynamics and TE integration from the TE Dashboard. This integration is done based on the information collected in the last section.

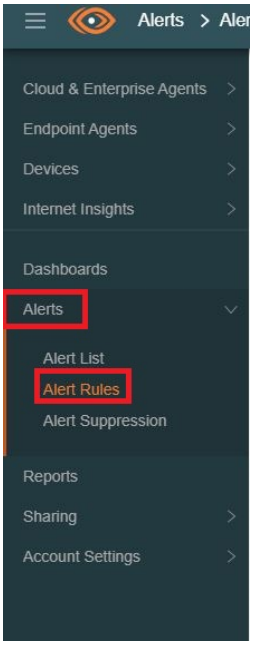
To get started, browse to the TE website and **Login**.



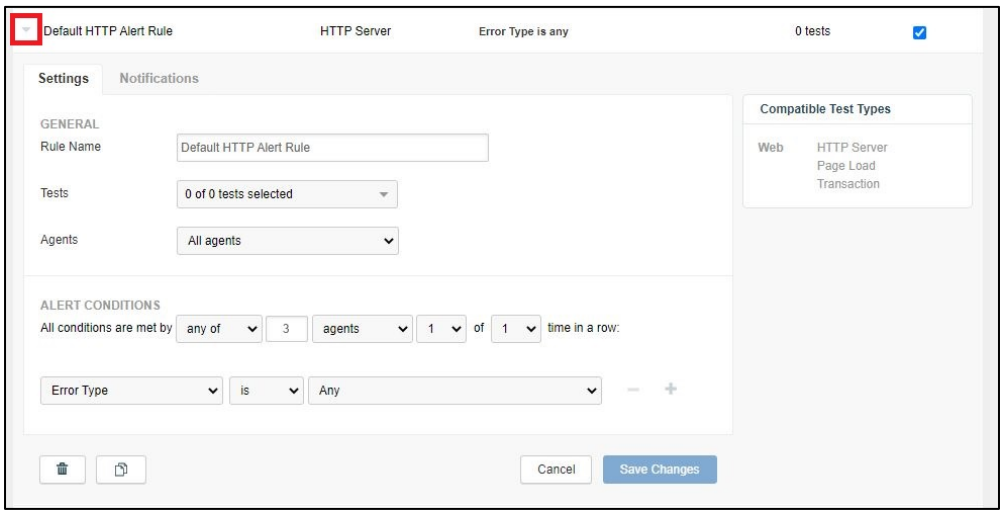
After logging in, the main TE Dashboard is displayed.



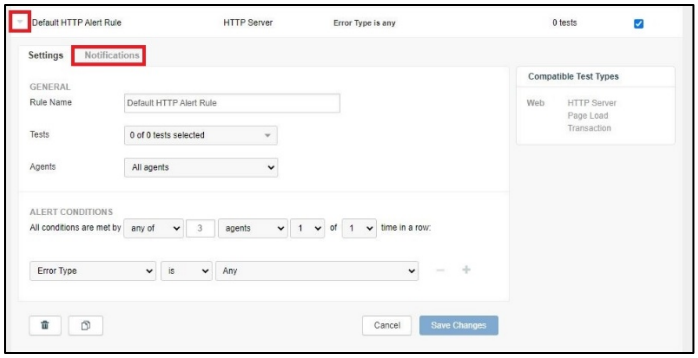
From here, browse to the **Alerts** section on the left-hand side of the dashboard and click on **Alert Rules**.



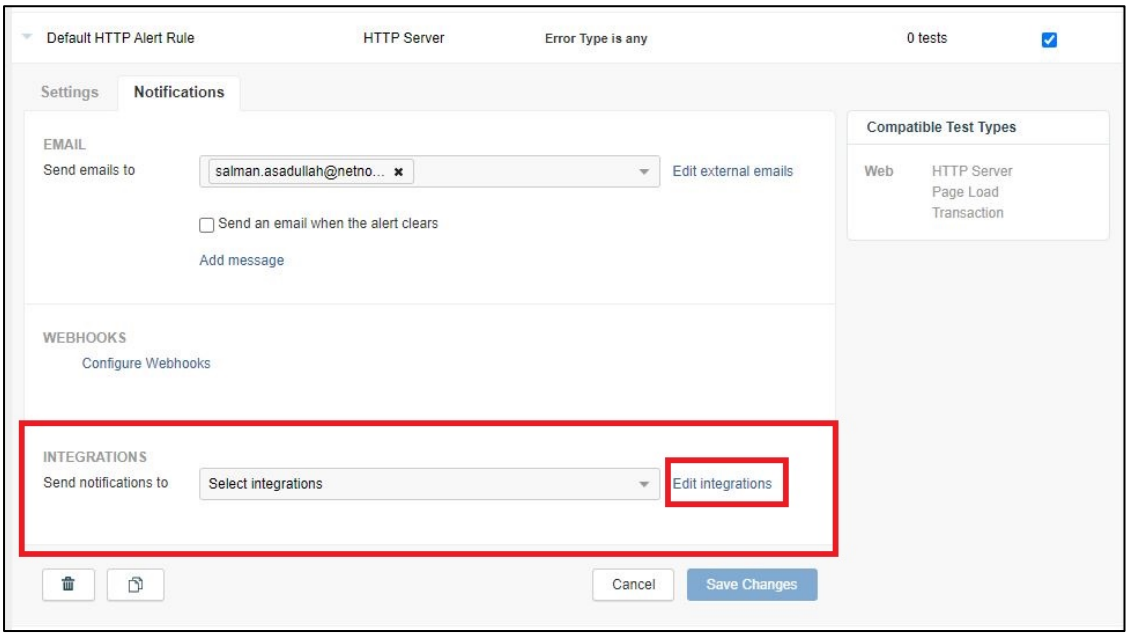
From the Alerts Rules page, navigate to 1 of the Rule Types and open it. For this integration, use-case **Default HTTP Alert Rule** is used.



From here, click on the **Notifications** tab to get to the integration area.

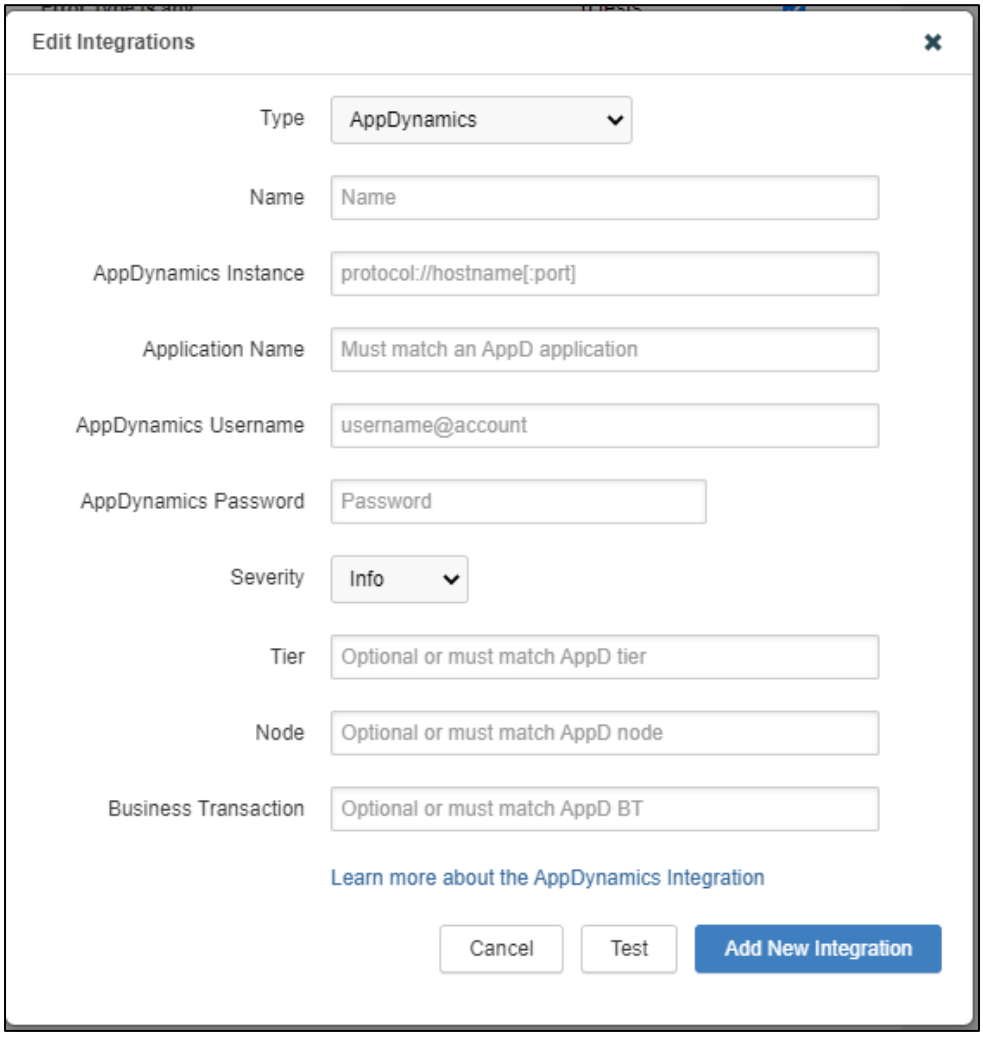


From the **Notifications** section of the **HTTP Alert Rule**, see in the lower section of the rule in the Integrations section. For this, **click** on the **Edit Integrations** blue icon to the right of the drop-down menu.



This will bring up another window that will show any current integrations. However, to add a new integration, click on the **Add New Integration**.

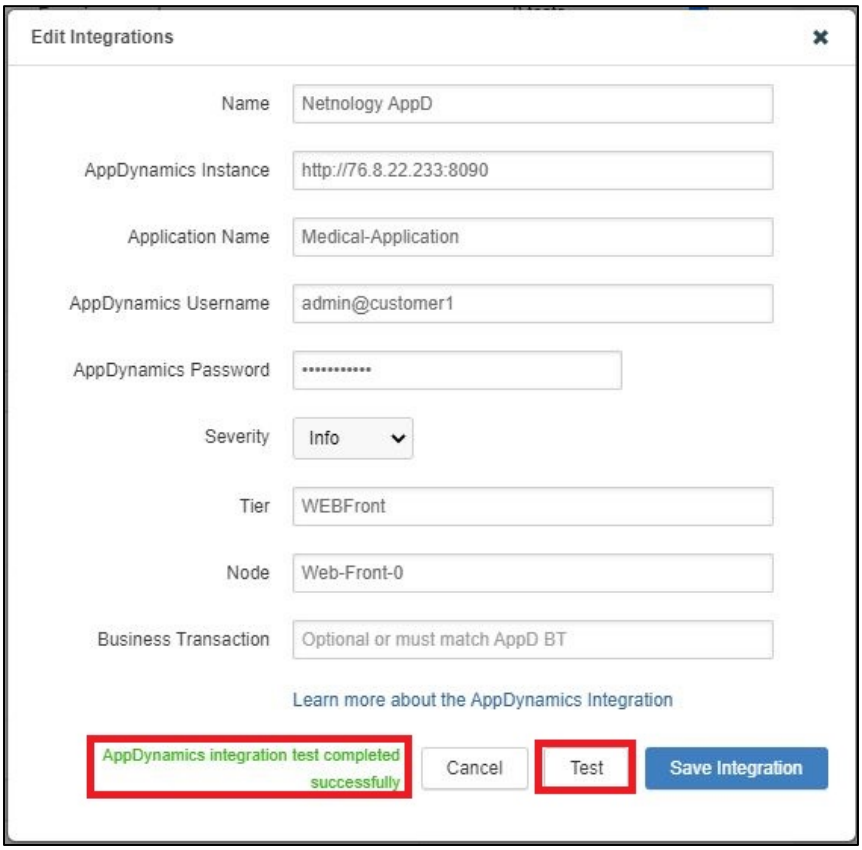
After clicking on the **Add New Integration** button, select the different integration from the drop down menu on the pop-up window. For this integration, use-case **AppDynamics** is used.



Now, complete all the required information which was already gathered from our earlier steps.

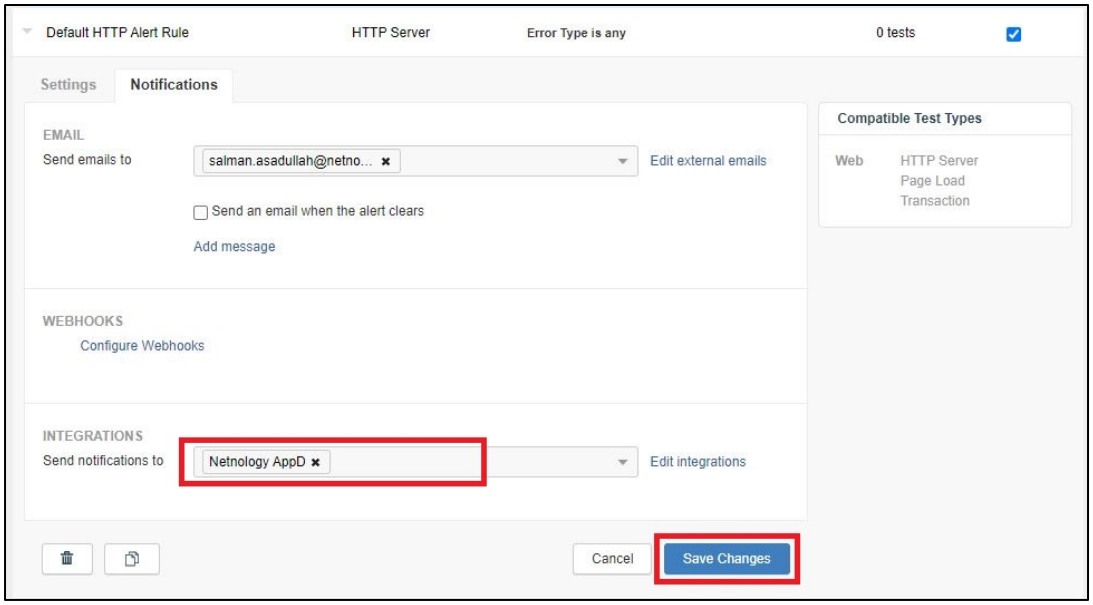
1) Name:
2) AppDynamics Instance:
3) Application Name:
4) AppDynamics Username:
5) AppDynamics Password:
6) Severity:

Once all the required fields are completed, **click** on the **Test** button to check if communication is working. As shown below, **"AppDynamics integration test completed successfully"** text in green should appear if TE is successful in communicating with the AppDynamics instance.



Based on successful test result, go ahead and click the **Save Integration** button and exit out of the **Edit Integration** window.

On the **Notifications** tab of the **Default HTTP Alert Rule**, select to **Send notifications to** the AppDynamics integration just completed and click on **Save Changes**.

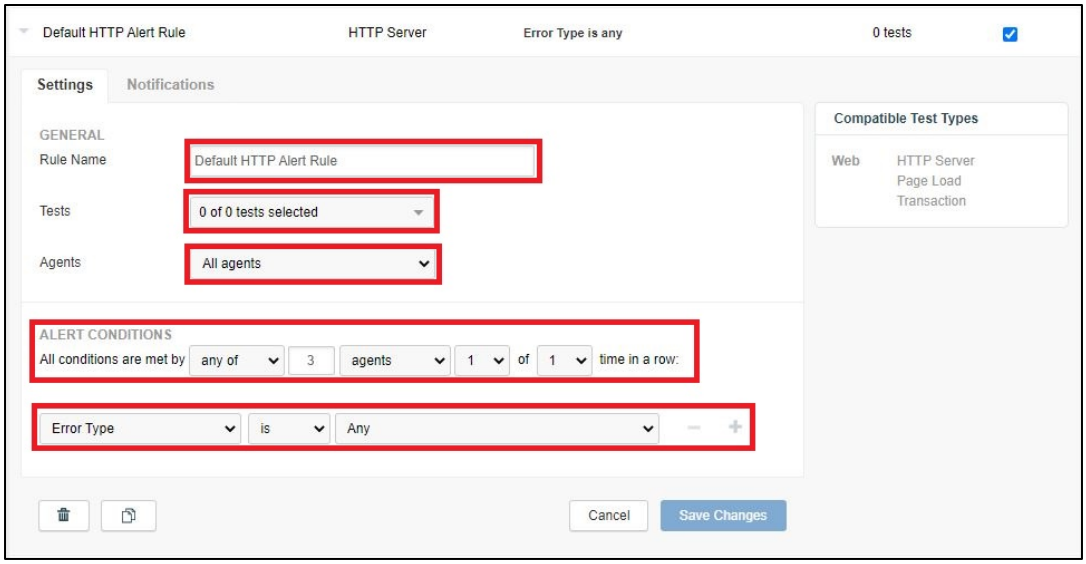


Next, under the **Settings** tab of the **Default HTTP Alert Rule** there are multiple selection options available such as:

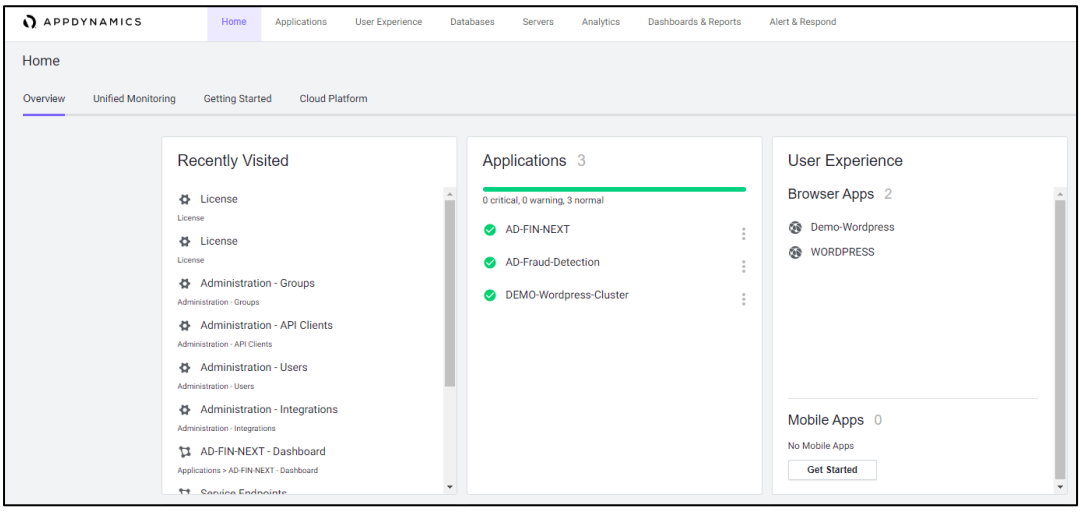
- Select Rule Name
- Select to link this alert to any test which is already set up in TE
- Select specific agent type or select all agents
- Select Alert conditions triggered due to an error or error type
- Select wait time of greater than 10 milliseconds

After making all the selections, click on **Save Changes** button.

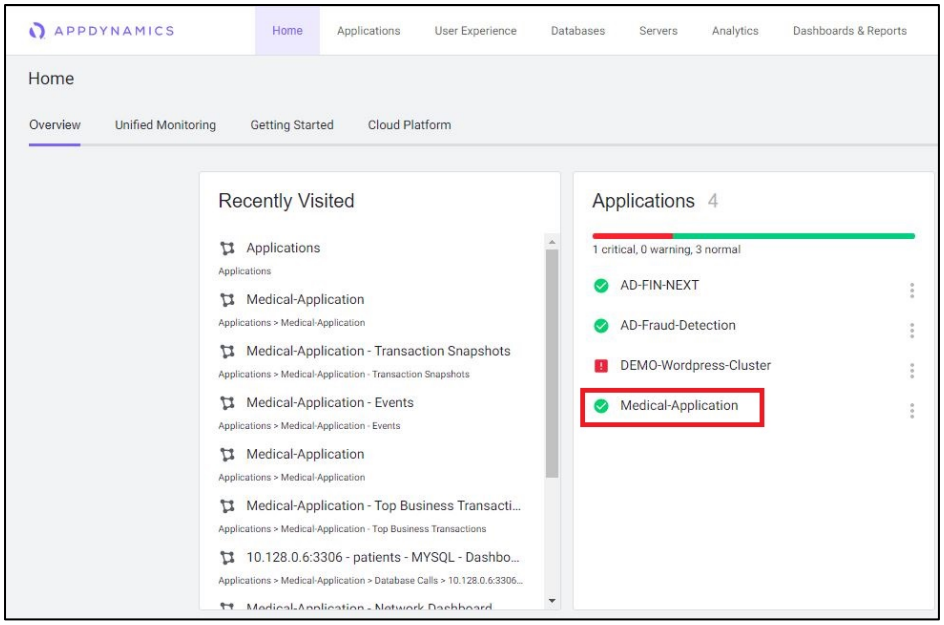
Any of the selected conditions would trigger an alert to be sent to AppDynamics explaining that some agents on a remote workers computer took longer than 10 milliseconds to get to the Application this alert is connected to. In this example, it is the **Medical-Application** application.



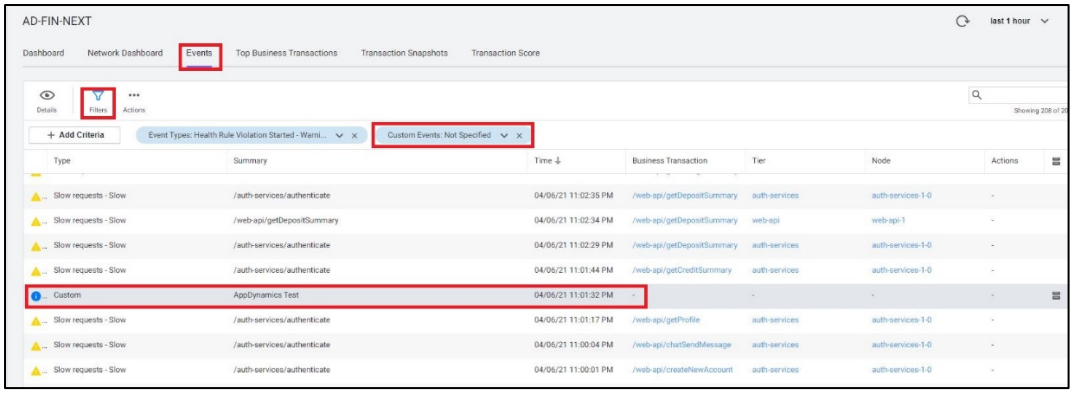
To make sure that this AppDynamics and TE integration is working, go back to the AppDynamics on-prem instance and login.



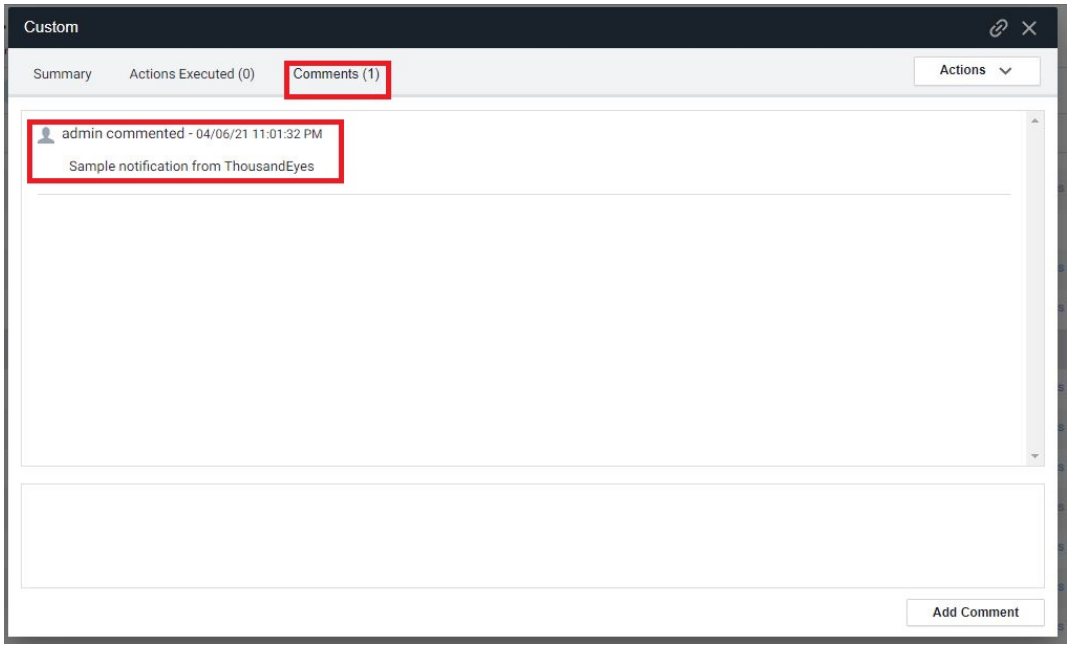
From the AppDynamics Dashboard, simply click on the **Medical-Application** application on the **Applications** window.



From the **Medical-Application** Applications Dashboard navigate to the **Events** tab and **Filter** on **Custom** alerts. Here, its seen that **Custom** alerts from TE are being sent to AppDynamics and **Medical-Application** is successfully testing the connection and alerts from TE.



Just to confirm that this information came from TE, double click on the **Custom** alerts, and navigate to the **Comments** tab. Message is seen with description "**Sample notification from ThousandEyes**".



Now any of the default TE **Alert Rules** can be set to integrate with AppDynamics and linked to an application if there is a need to track DNS Errors, FTP applications being monitored by AppDynamics, HTTP traffic errors, or SSL certificate expiry within 30 days rules for an AppDynamics application.

5 - Conclusion

Cisco AppDynamics, TE and IWO integration can provide any organization full-stack observability into the resources of an application all the way to the end user browsing experience.

Following key tasks were executed in this integration example:

- Install Database agents on server for AppDynamics.
- Create User in AppDynamics.
- Integrate AppDynamics with IWO.
- Integrate AppDynamics with TE.
- Install Intersight Assist VM and add to IWO.
- Configure a Target in IWO.
- Manually increase and decrease application resources in IWO.