# Open-AudIT 3.3.0

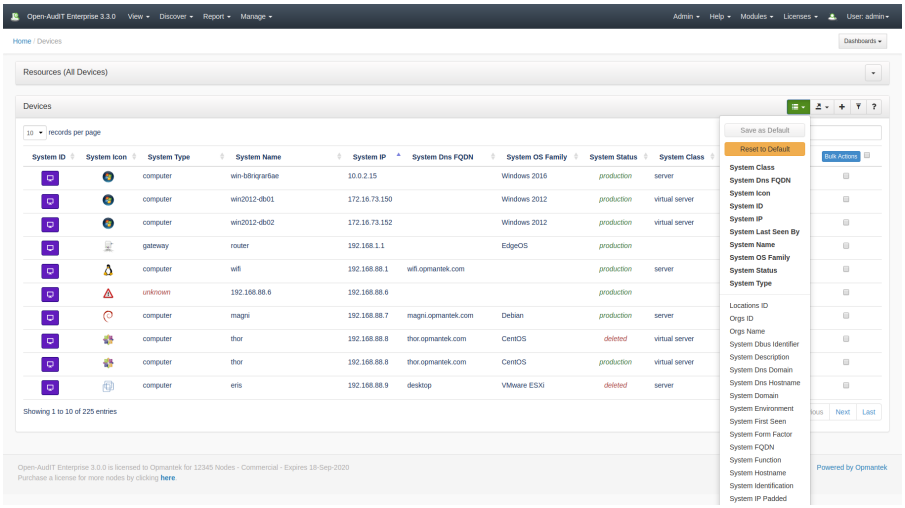


Hi All,

Release 3.3.0 of Open-AudIT has some amazing new features, read on for the details. The release notes are available as usual, here - Release Notes for Open-AudIT v3.3.0.

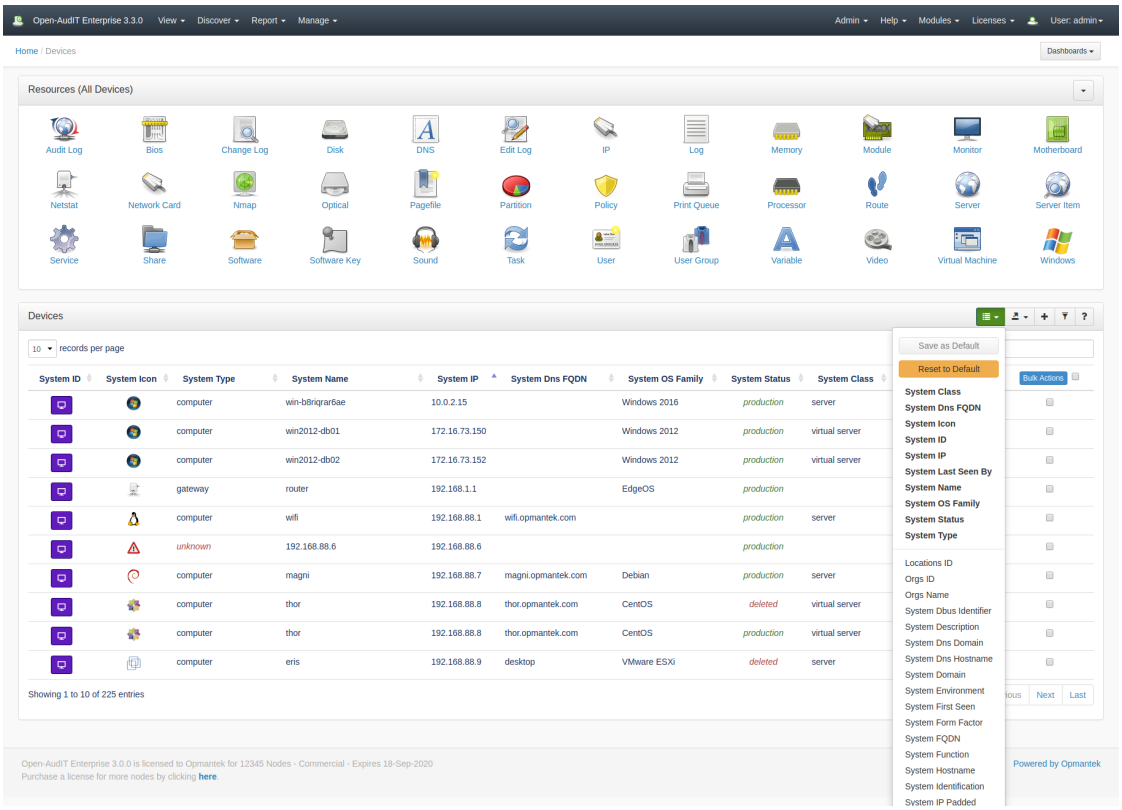
## Configurable Device Columns

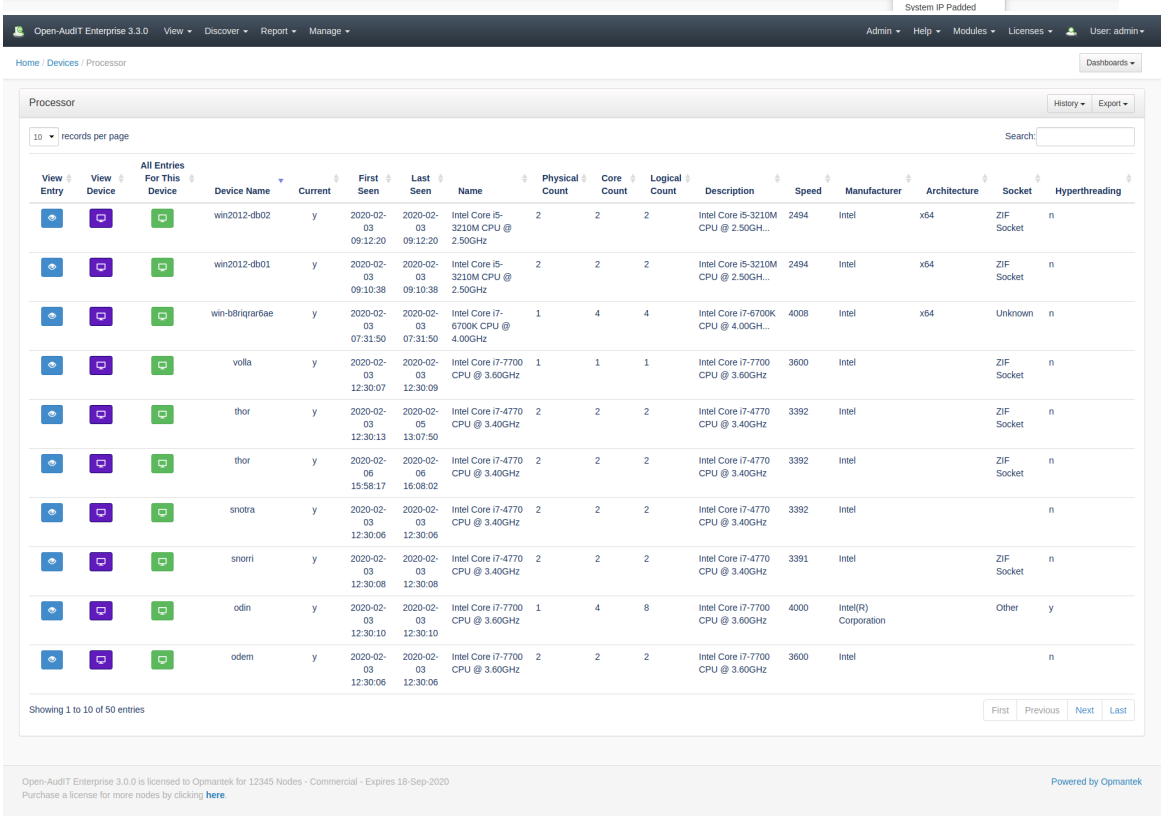
From 3.3.0 onward, when you view the list of devices (Manage  Devices  List Devices), you'll notice a small additional control on the upper right. Click it and you'll see a list of available columns you can display. Click a column name and it will appear. Click a bold column name and it will disappear. If you want that set of columns as your default, click "Save as Default" and every time you view the device list, those will be your default columns. You can also click "Reset to Default" (if your columns are different) to reset them. The default list of columns is in the configuration under the name devices_default_display_columns. If you are seeing a n unacceptable slow down viewin gthe page, you might wish to limit the retrieved (but not displayed) columns. This is also in the configuration under the name devices_default_retrieve_columns. See the screenshot below.




## Device Components

Also on the Devices list page, you'll notice a bar at the very top with a drop down  arrow on the right. Click the arrow and you'll see a list of component types. Click one to see a list of all those items. Be aware this list may be very large so we restrict it to the first "database_show_row_limit" (configuration item) entries. Increase that number to see more. At this stage we have not implemented a GUI for paging, but it is available using the API (or adding to the URL), by specifying limit and offset. So a valid URL might be (for instance) ***http://SERVER/omk/open-audit/devices?sub_resource=software&limit=200&offset=100***. See the API documentation for more information - The Open-AudIT API. The following pages allow you to click links to see this specific entries details, all those entries on a device or the device details itself.




# Comparing Your Database Schema

There is a new entry under menu  Admin  Database called Schema Compare. Running that will show you the schema as it is in your running database and compare it to the schema as shipped with Open-AudIT. If there are any differences, just post them to Questions and we can help you out. For supported customers, just log a support request and we'll assist ASAP.

# Change Log Improvements

Time has been spent to minimize false positive Change Logs being generated. As well as that, we have added two buttons on the Device Details screen (under the left side Actions menu) to remove Change Logs and remove Audit Logs. Using these may help improve database performance where these records are not required. Don't forget you can always clear the entire tables using menu  Admin  Database  List Tables, clicking either table and hitting the Delete button. And don't forget about our new configuration items for keeping non-current items and creating change logs. More information on these can be found here - Device SubSection Data Retention Options.

# Deleting Devices

There is now a configuration item named device_auto_delete. If set to 'y' (it is set to 'n' by default) when you change a device's status (either individually or using Bulk Edit) you will get a regular warning "Are you sure?" and if you answer yes, the device and all it's details will be completely removed from the database. Not just flagged with a status of deleted.

# New Discovery Process and Improvements

With the coming release of Open-AudIT 3.3.0 we have implemented a new discovery process that scales even better than previously. Even faster discovery times!

## The Discovery Queue

With 3.3.0 we have changed to using the discovery queue, not on a per discovery basis as previously, but on a per IP basis. From 3.3.0 onwards, when you click "Execute Discovery", the following happens behind the scenes:

- The server starts a script that calls /util/queue and instantly returns to the web user (or the API user). It starts the shell script and does not wait for a response before returning.
- The user then continues on using the web/API as per normal.
- The shell script calls util/queue - this endpoint only accepts requests from localhost. The resulting function does the following:
    1. Check the config for the queue limit. If this has been reached, exit. If it has not been reached, continue.
    2. Pop an item from the queue (locking the queue table as it does so). The item is read from the database, then deleted. If no queue items exist, exit.
    3. Spawn another script to request util/queue.
    4. Execute the item - which on the first time is always "run discovery on subnet".
    5. When finished, return to #1.

There are (currently) two types of queue entries. The overall discovery entry, and an entry for each IP to be scanned. The second entry is created by the first. So we run the initial discovery, and for each IP we need to scan (that responds, if that option is chosen), we create another entry to scan that device.

We no longer use the discover_subnet.sh or discover_subnet.vbs scripts at all. We now call Nmap directly from within the Open-AudIT code, which frees us up to have one and only one routine (versus a bash and vbscript). It also makes it easier to code - PHP has much easier to use text parsing than bash and vbscript (in my opinion).

Because of the above, we have created a new configuration item called "discovery_limit" and set it to 20 by default. This means when a discovery is run, it will spawn up to 20 processing instances in parallel. Because of this parallel processing of target IPs, discovery is $discovery_limit times faster. Well, not quite, but you get what I mean. The old way ran a discovery and for each target, sequentially, started a scan. Several scans were run at once, but it was still waiting for an Nmap scan, before handing off to PHP to complete the rest. The new way completes the initial scan and loads all resulting devices into the queue to be processed in parallel. At the end of the day, it's just so much faster.

## Sudo SSH Key and Password

Previously we only had support for an SSH Key that used a password, but where that password was also used for sudo. That is obviously not optimal, so as at 3.3.0 you can add a specific ssh key password and a sudo password.

## Additional Nmap Option

We have also added a new option to discovery scan options - open|filtered. Previously we used the "filtered" column to check for open|filtered. This change aligns the discovery scan options with Nmap return strings.

## Auditing Time Reduction When Using sudo

When auditing a device using sudo, we no longer have to wait for the configuration item discovery_ssh_timeout (previously 300 seconds) to timeout. We now check every 2 seconds for our response and when received, proceed. This has made another large difference to audit times when using sudo.

## Windows Users Apache Service

As well as this, there has been a change targeted specifically to Windows Open-AudIT Servers. Because of the issue's we have run into using the default service account, you will now get a big warning stating you should change the service account to a "real" account. This is because by default the service account cannot access network resources. IE - copy the audit script to thew target and run it. The "old" way of running the script on the Open-AudIT server itself and specifying the target still works and is enabled by a config item - discovery_use_vintage_service, which is set to 'n' by default. One reason for this is that the discovery script contains sections that do not and can not work remotely. Think starting an executable. That won't work as WMI can target the remote machine, but running an executable from the audit script would run it where the script is running - the Open-AudIT server.

## The Default Network Address

Because of our new way of running discovery, we no longer need to set the Default Network Address. The scripts are run on the target devices and create a file (as opposed to submit_online=y). That file is then copied to the server and processed, rather than submitted using the URL (that was created from the default_network_address). The only reason to set the Default Network Address for Discovery is if you're using discovery_use_vintage_service. The now normal use of this is only for the "Audit My PC" functionality.

## Auto Delete our Audit Script

Now when discovery runs, the audit script deletes itself on the target, hence we leave nothing present on the target device.

## No More "New" Devices Where We Have No Information

We have added a new configuration option called match_ip_no_data. If we discover a device and that IP is already in the database ***and*** we have no audit data about that device, assume it is the same device, so do not create another (usually duplicate) device.

## SNMP Route Retrieval

We now retrieve the first (configuration item discovery_route_retrieve_limit) routes from a device when using SNMP.

And there's even more improvements. Make sure you read the Release Notes for Open-AudIT v3.3.0 to stay across it all.

Mark Unwin.