

# Clouds

- [Introduction](#)
- [How Does it Work?](#)
- [Creating a Clouds Entry](#)
- [Database Schema](#)
- [API / Web Access](#)

## Introduction

With the introduction of the Clouds feature, you can supply details of your cloud, Amazon AWS or Microsoft Azure, and Open-Audit will use the relevant API to retrieve a list of locations, networks, and instances, then discover each of these instances.

Full cloud auditing with all the details from a regular Open-Audit audit AND details such as the machine type and size.



### Device Matching Rules

To use this feature you must enable the configuration items `match_mac` (for AWS) and `match_hostname` (for Azure). This will be done automatically the first time a cloud discovery is executed. For more on Open-Audit's matching rules see [HERE: Matching Devices](#)

## How Does it Work?

Once you have your cloud credentials, it is merely a matter of creating a new cloud and running the discovery - or scheduling first/future discoveries. Make sure you also have the credentials required to log on to your cloud machines, and it is all done for you just like a regular discovery.

Clouds are a feature available to Open-Audit Enterprise licensed customers.

## Creating a Clouds Entry

Simply go to menu -> Discover -> Clouds -> Create Clouds.

You will need to supply a cloud name and type and depending on the type selected, supply the relevant credentials. Screenshots are below showing the currently supported types of AWS and Azure.

- [Credentials for Amazon AWS used in Cloud Discovery](#)
- [Credentials for Microsoft Azure used in Cloud Discovery](#)

Open-Audit Enterprise 2.0.0 is licensed to Opnstack for 12345 Nodes - Expires 19-Jul-2019  
Purchase a license for more nodes by clicking [here](#) Powered by Opnstack

Open-Audit Enterprise 2.0.0 is licensed to Opnstack for 12345 Nodes - Expires 19-Jul-2019  
Purchase a license for more nodes by clicking [here](#) Powered by Opnstack

## Database Schema

The database schema can be found in the application if the user has database::read permission by going to menu: Admin -> Database -> List Tables, then clicking on the details button for the table.

## API / Web Access

You can access the collection using the normal Open-Audit JSON based API. Just like any other collection. Please see [The Open-Audit API](#) documentation for further details.

