



LAB: osTicket Lab and Integration

Login credential is given in the LAB guide, please follow it to access the VM machine.

Note

- We will perform the LAB on the same machine as it is, where SNMP-LibreNMS lab is done.
- Server Hostname will have an extension (osticket) in this LAB, example –
`groupX-server-osticket.apnictraining.net`
- `$` means a general user is having the CLI (command line interface)
- `#` means root user is having the CLI
- Do not attempt to perform the LAB with `#` root user unless it is explicitly mentioned.

Requirements

- HTTP server running Nginx or Apache. As like the previous LAB modules, here we will use Apache.
- PHP version 5.6 or greater. (We will use php 7.4 following previous LAB to avoid complexity)
- PHP extensions – mysqli, gd, gettext, imap, json, mbstring, and xml
- MySQL database version 5.0 or greater. (We will use MySQL v-15.1 following previous LAB to avoid complexity)

1. Update the System Repository

```
$ sudo apt update && sudo apt upgrade
```

2. Install Required Packages

First we have to update the package index, and then we will install the required packages for LAMP service:

```
$ sudo apt install software-properties-common
$ sudo add-apt-repository universe
$ sudo add-apt-repository ppa:ondrej/php
$ sudo apt update
$ sudo apt install apache2 php7.4 libapache2-mod-php7.4 \
php7.4-common php7.4-mysql php7.4-gmp \
php7.4-curl php7.4-intl php7.4-mbstring \
php7.4-xmllrpc php7.4-apcu php7.4-gd \
php7.4-bcmath php7.4-soap php7.4-ldap \
php7.4-imap php7.4-xml php7.4-cli \
php7.4-zip mariadb-server mariadb-client \
git curl
```

3. Configure PHP

Update the `php.ini` configuration file, following the below settings. Check the list of timezone info from the URL – <http://php.net/manual/en/timezones.php> .

```
$ sudo vim /etc/php/7.4/apache2/php.ini

file_uploads = On
allow_url_fopen = On
short_open_tag = On
memory_limit = 256M
upload_max_filesize = 100M
max_execution_time = 360
date.timezone = Australia/Brisbane
```

Execute the below command after changing the file. (ignore, if you have done LibreNMS LAB successfully)

```
$ sudo a2enmod php7.4
```

4. Create osticket Database

We will be using `training` as password for the LAB.

```
$ sudo mysql --user=root mysql

[mysql]> CREATE DATABASE osticket CHARACTER SET utf8 COLLATE utf8_unicode_ci;
[mysql]> CREATE USER 'osticket'@'localhost' IDENTIFIED BY 'training';
[mysql]> GRANT ALL ON osticket.* to 'osticket'@'localhost' WITH GRANT OPTION;
[mysql]> FLUSH PRIVILEGES;
[mysql]> exit
```

Restart the MySQL service:

```
$ sudo systemctl restart mysql
```

5. Download and Install osTicket

```
$ curl -sS https://getcomposer.org/installer | sudo php -- \
--install-dir=/usr/local/bin --filename=composer
```

Change to apache directory to download osticket, and change the permission+ ownership following the below commands.

```
$ cd /var/www/html
$ sudo git clone https://github.com/osTicket/osTicket osticket

$ cd osticket
$ sudo php manage.php deploy --setup /var/www/html/osticket/
$ sudo mv /var/www/html/osticket/include/ost-sampleconfig.php \
/var/www/html/osticket/include/ost-config.php

$ sudo chown -R www-data:www-data /var/www/html/osticket/
$ sudo chmod -R +x /var/www/html/osticket/
```

6. Configure Apache virtualhost for osTicket

```
$ sudo vim /etc/apache2/sites-available/osticket.conf

<VirtualHost *:80>
    DocumentRoot /var/www/html/osticket/
    ServerName groupX-server-osticket.apnictraining.net

    <Directory /var/www/html/osticket/>
        Options FollowSymlinks
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

Now enable newly configured virtual host `osticket` and restart the apache service.

```
$ sudo a2ensite osticket.conf
$ sudo a2enmod rewrite
$ sudo systemctl restart apache2.service
```

7. Web Installer Wizard

Open the browser and browse the server name; the osTicket setup wizard is there to complete the installation task. The screenshots are attached as a reference.

```
http://groupX-server-osticket.apnictraining.net
```

Pre-install checks, make sure the prerequisites are marked green.



Thank You for Choosing osTicket!

We are delighted you have chosen osTicket for your customer support ticketing system!

The installer will guide you every step of the way in the installation process. You're minutes away from your awesome customer support system!

Prerequisites

Before we begin, we'll check your server configuration to make sure you meet the minimum requirements to run the latest version of osTicket.

Required:

These items are necessary in order to install and use osTicket.

- ✓ PHP v7.2 or greater — (7.4.13)
- ✓ MySQLi extension for PHP — **module loaded**

Recommended:

You can use osTicket without these, but you may not be able to use all features.

- ✓ Gdlib Extension
- ✓ PHP IMAP Extension — *Required for mail fetching*
- ✓ PHP XML Extension (for XML API)
- ✓ PHP XML-DOM Extension (for HTML email processing)
- ✓ PHP JSON Extension (faster performance)
- ✓ Mbstring Extension — *recommended for all installations*
- ✓ Phar Extension — *recommended for plugins and language packs*
- ✓ Intl Extension — *recommended for improved localization*
- ✓ APCu Extension — (faster performance)
- ✓ Zend OPcache Extension — (faster performance)

[Continue »](#)

Need Help?

If you are looking for a greater level of support, we provide [professional installation services](#) and commercial support with guaranteed response times, and access to the core development team. We can also help customize osTicket or even add new features to the system to meet your unique needs. [Learn More!](#)

Click on the continue button, Provide necessary information as shown in the screenshots; ensure all the blank space are filled up.

Give the helpdesk a name and an email ID for correspondence.

osTicket Basic Installation

Please fill out the information below to continue your osTicket installation. All fields are required.

System Settings

The URL of your helpdesk, its name, and the default system email address

Helpdesk URL:

http://groupX-server-osticket.apnictraining.net

Helpdesk Name:

apnic-helpdesk

Default Email:

helpdesk@apnic.net

Primary Language:

English (United States) ▼

Next, input the database information as, database name `osticket` , user `osticket` , password `training` ; and then create the admin user named `apnic` , and password `training` .

Admin User

Your primary administrator account - you can add more users later.

First Name:
 ?

Last Name:
 ?

Email Address:
 ?

Username:
 ?

Password:
 ?

Retype Password:
 ?

Database Settings

Database connection information

MySQL Table Prefix:
 ?

MySQL Hostname:
 ?

MySQL Database:
 ?

MySQL Username:
 ?

MySQL Password:
 ?

Need Help? We provide [professional installation services](#) and commercial support. [Learn More!](#)

Click the `install now` button to finish the installation process.

Next, remove the `setup` directory and change the file permission for security purpose.

```
$ cd /var/www/html/osticket

$ sudo rm -rf setup
$ sudo chmod 644 include/ost-config.php
```



Congratulations!

Your osTicket installation has been completed successfully. Your next step is to fully configure your new support ticket system for use, but before you get to it please take a minute to cleanup.

Config file permission:

Change permission of ost-config.php to remove write access as shown below.

- **CLI:**
`chmod 0644 include/ost-config.php`
- **Windows PowerShell:**
`icacls include\ost-config.php /reset`
- **FTP:**
Using WS_FTP this would be right hand clicking on the file, selecting chmod, and then remove write access
- **Cpanel:**
Click on the file, select change permission, and then remove write access.

Below, you'll find some useful links regarding your installation.

Your osTicket URL:

<http://groupx-server-osticket.testapnictraining.net/>

osTicket Forums:

<https://forum.osticket.com/>

Your Staff Control Panel:

<http://groupx-server-osticket.testapnictraining.net/scp>

osTicket Documentation:

<https://docs.osticket.com/>

PS: Don't just make customers happy, make happy customers!

What's Next?

Post-Install Setup: You can now log in to [Admin Panel](#) with the username and password you created during the install process. After a successful log in, you can proceed with post-install setup. For complete and upto date guide see [osTicket wiki](#)

Commercial Support

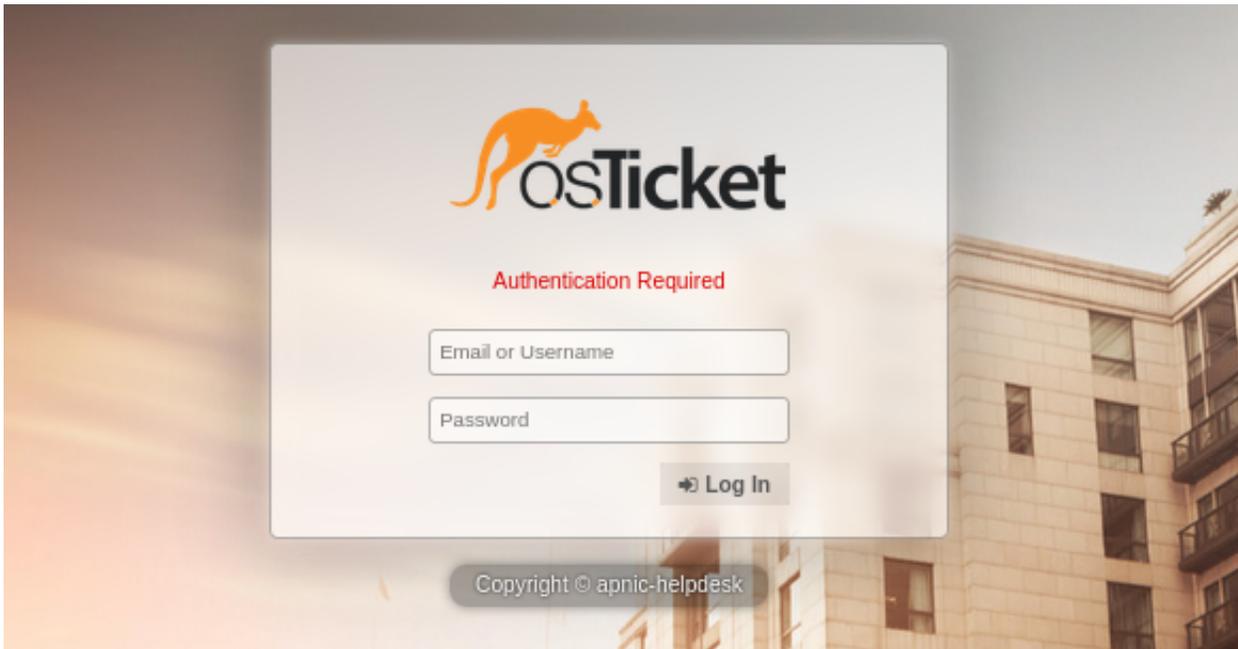
Available: Don't let technical problems impact your osTicket implementation. Get guidance and hands-on expertise to address unique challenges and make sure your osTicket runs smoothly, efficiently, and securely. [Learn More!](#)

Congratulation, your osTicket installation is done. Now browse the ticketing system to explore more.

Note This Login URL is for the Admin Dashboard.

`http://groupX-server-osticket.apnictraining.net/scp/login`

You will see a login prompt like below. Login and explore with username `apnic` and password `training`.

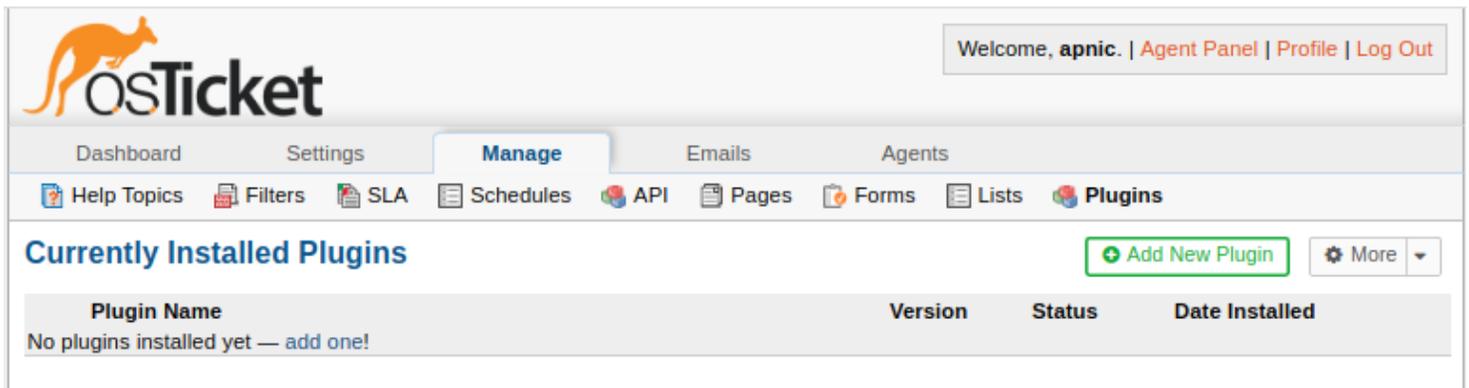


8. osTicket Integration with Slack

We need to download the Slack plugin from github to the plugin directory of osTicket, and enable it.

```
$ cd /var/www/html/osticket/include/plugins
$ sudo git clone https://github.com/clonemeagain/osticket-slack.git
```

Next, go to the web-console and select "Admin Panel > Manage > Plugins > Add New Plugin" ; You will see all the plugins that are already installed.



Now select the `Slack notifier` plugin to install at the web-console



Welcome, **apnic.** | [Agent Panel](#) | [Profile](#) | [Log Out](#)

Dashboard Settings **Manage** Emails Agents

[Help Topics](#) [Filters](#) [SLA](#) [Schedules](#) [API](#) [Pages](#) [Forms](#) [Lists](#) [Plugins](#)

Install a new plugin

To add a plugin into the system, download and place the plugin into the `include/plugins` folder. Once in the plugin is in the `plugins/` folder, it will be shown in the list below.

Install

Slack notifier
 Notify Slack on new ticket.
Version: 0.2
Author: Thammanna Jammada

A prompt will popup showing the status of a successful installation.



Welcome, **apnic.** | [Agent Panel](#) | [Profile](#) | [Log Out](#)

Dashboard Settings **Manage** Emails Agents

[Help Topics](#) [Filters](#) [SLA](#) [Schedules](#) [API](#) [Pages](#) [Forms](#) [Lists](#) [Plugins](#)

✔ Successfully installed a plugin

Currently Installed Plugins

➕ Add New Plugin
⚙ More ▾

Plugin Name	Version	Status	Date Installed
<input type="checkbox"/> Slack notifier	0.2	Disabled	12/8/20 8:58 PM

Select: [All](#) [None](#) [Toggle](#)

Page: [\[1\]](#)

As the plugin is `Disabled` by-default, we have to enable it, check the box and then select the `Enable` option from the `More` option.



Welcome, **apnic.** | [Agent Panel](#) | [Profile](#) | [Log Out](#)

Dashboard Settings **Manage** Emails Agents

[Help Topics](#) [Filters](#) [SLA](#) [Schedules](#) [API](#) [Pages](#) [Forms](#) [Lists](#) [Plugins](#)

✔ Successfully installed a plugin

Currently Installed Plugins

➕ Add New Plugin
⚙ More ▾

Plugin Name	Version	Status	Date I
<input checked="" type="checkbox"/> Slack notifier	0.2	Disabled	12/8/2

✔ Enable

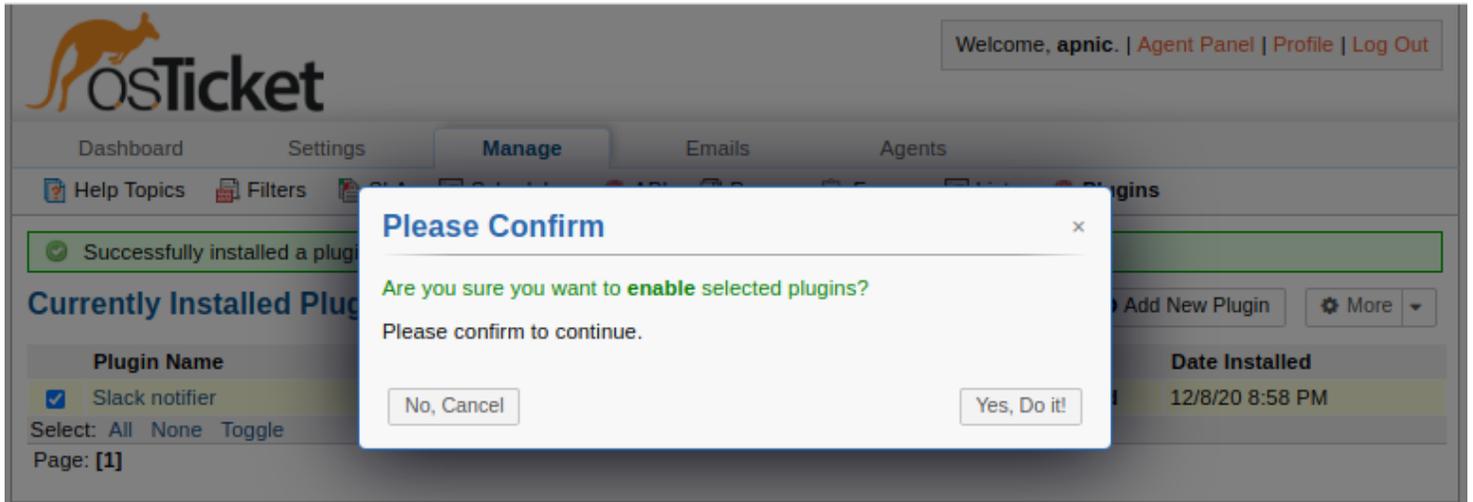
⊘ Disable

🗑 Delete

Select: [All](#) [None](#) [Toggle](#)

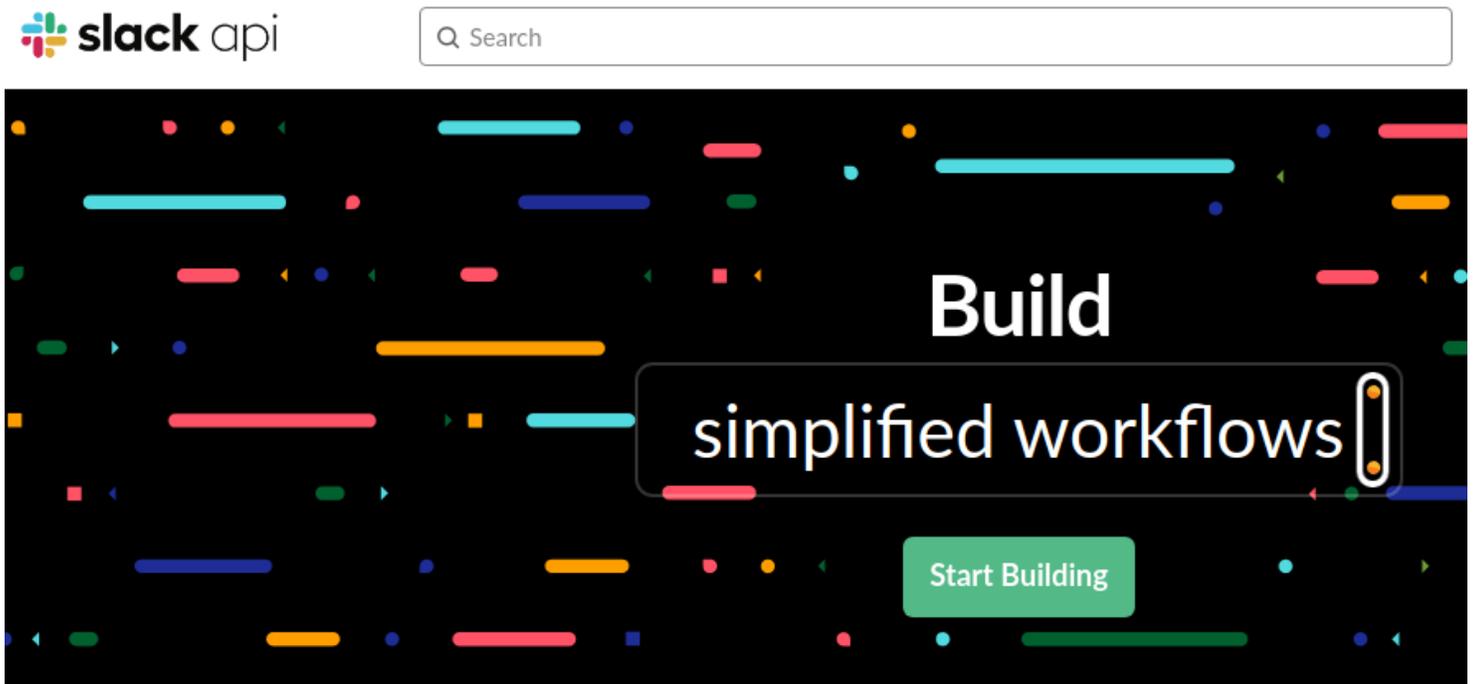
Page: [\[1\]](#)

It will ask for confirmation, click `Yes, Do it!` .



Slack plugin installation is done. Now we have to configure Slack to get alerts from osTicket.

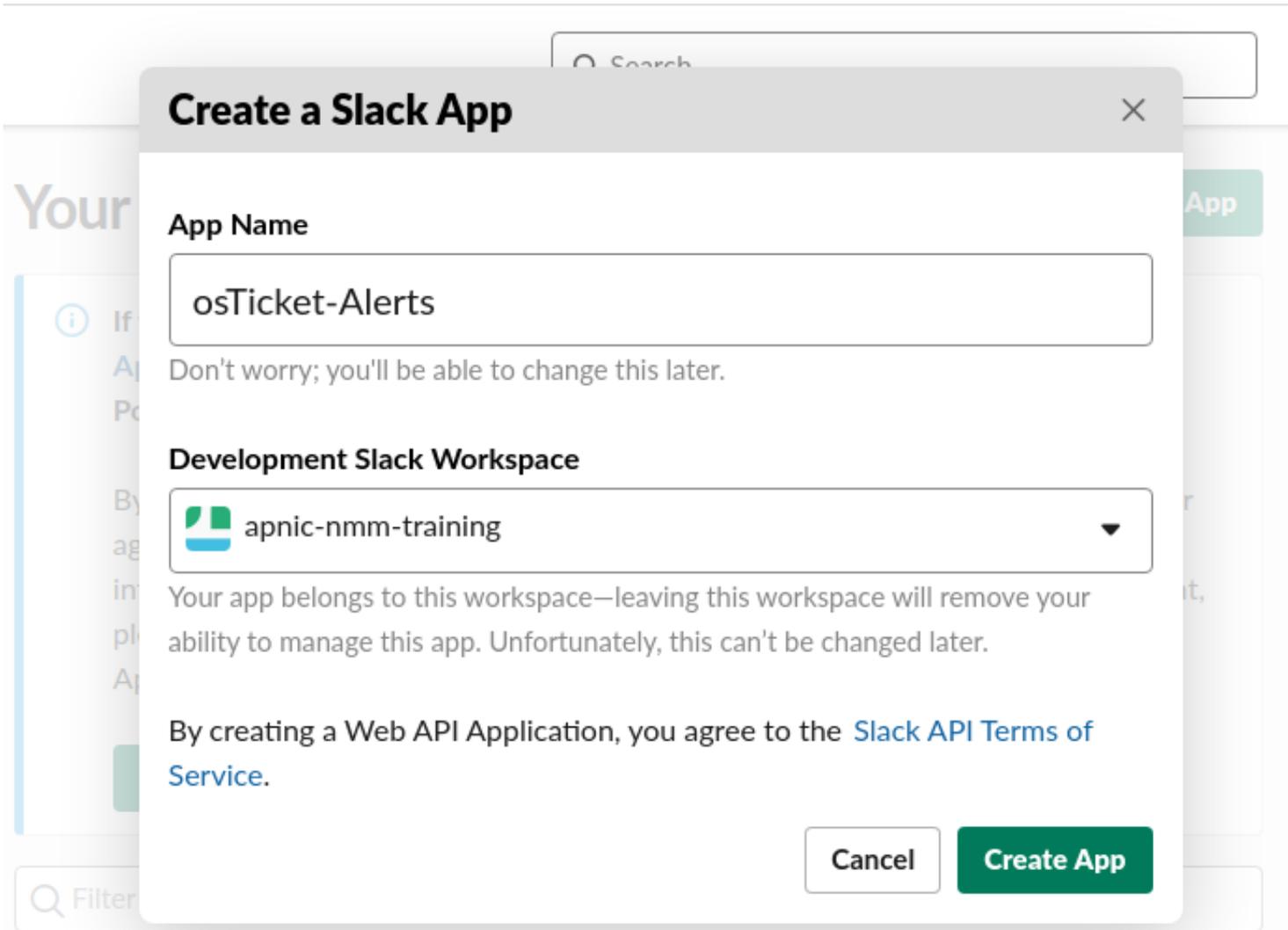
Next, navigate to `https://api.slack.com/` and select the option `"Start Building"`



It will prompt a pop-up window to create a new-app for the slack channel. Give the app a name; example

`"osTicket-Alerts"` and select your workspace from the drop-down menu

`"Development Slack Workspace"` , and then click `"Create App"`



Next, new window will come up, select "Incoming Webhooks" from the menu "Add features and functionality".

Settings

- Basic Information**
- Collaborators
- Install App
- Manage Distribution
- Submit to App Directory

Features

- App Home
- Org Level Apps
- Incoming Webhooks
- Interactivity & Shortcuts
- Slash Commands
- Workflow Steps
- OAuth & Permissions
- Event Subscriptions
- User ID Translation
- Beta Features

Building Apps for Slack

Create an app that's just for your workspace (or build one that can be used by any workspace) by following the steps below.

Add features and functionality

Choose and configure the tools you'll need to create your app (or review all [our documentation](#)).

Incoming Webhooks

Post messages from external sources into Slack.

Interactive Components

Add components like buttons and select menus to your app's interface, and create an interactive experience for users.

Slash Commands

Allow users to perform app actions by typing commands in Slack.

Event Subscriptions

Make it easy for your app to respond to activity in Slack.

Activate the option "Activate Incoming Webhooks" by clicking the **Off/On** button.

osTicket-Alerts

Settings

- Basic Information
- Collaborators
- Install App
- Manage Distribution
- Submit to App Directory

Features

- App Home
- Org Level Apps
- Incoming Webhooks**
- Interactivity & Shortcuts
- Slash Commands
- Workflow Steps

Incoming Webhooks

Activate Incoming Webhooks

Incoming webhooks are a simple way to post messages from external sources into Slack. They make use of normal HTTP requests with a JSON payload, which includes the message and a few other optional details. You can include [message attachments](#) to display richly-formatted messages.

Adding incoming webhooks requires a bot user. If your app doesn't have a [bot user](#), we'll add one for you.

Each time your app is installed, a new Webhook URL will be generated.

If you deactivate incoming webhooks, new Webhook URLs will not be generated when your app is installed to your team. If you'd like to remove access to existing Webhook URLs, you will need to [Revoke All OAuth Tokens](#).

Bottom of the page has an option "Add a new Webhook to workspace", it will prompt a new window.

Webhook URLs for Your Workspace

To dispatch messages with your webhook URL, send your [message](#) in JSON as the body of an `application/json` POST request.

Add this webhook to your workspace below to activate this curl example.

Sample curl request to post to a channel:

```
curl -X POST -H 'Content-type: application/json' --data '{"text":"Hello, World!"}' YOUR_WEBHOOK_URL_HERE
```

Webhook URL	Channel	Added By
-------------	---------	----------

No webhooks have been added yet.

[Add New Webhook to Workspace](#)



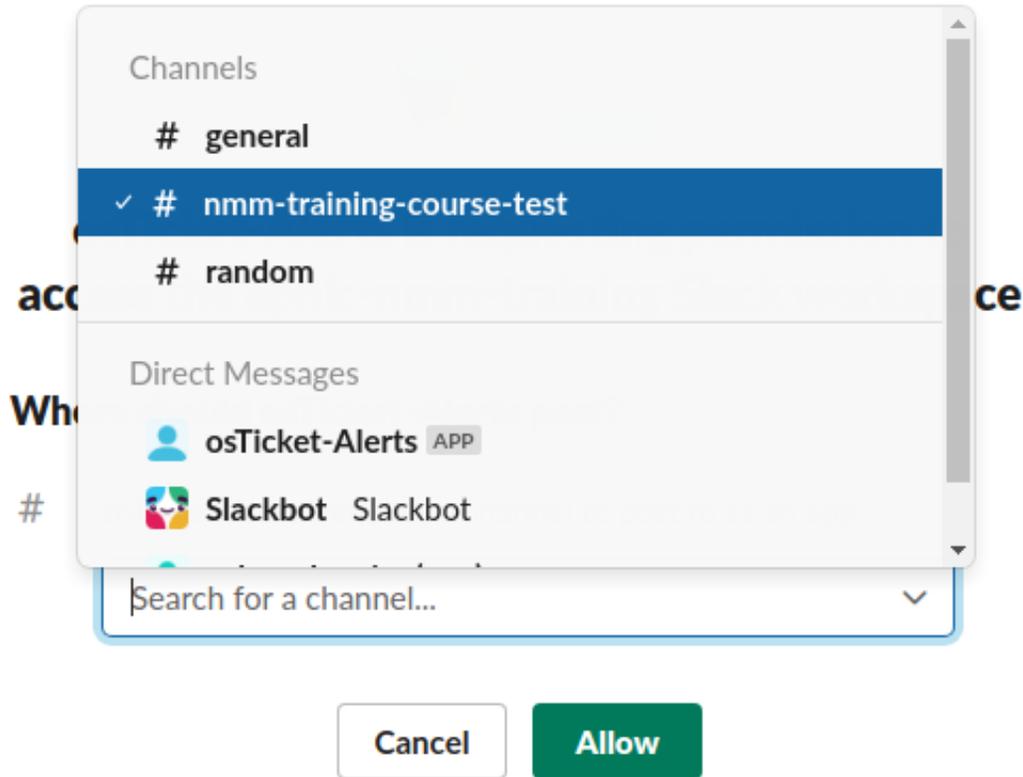
osTicket-Alerts is requesting permission to access the apnic-nmm-training Slack workspace

Where should osTicket-Alerts post?

osTicket-Alerts requires a channel to post to as an app

Cancel Allow

Select the `channel to post` from the drop down menu, and click the `"Allow"` button to complete the task.



Slack channel setup is done.

Now scroll down to the page and copy the "Webhook URL" and paste it to Slack notifier plugin at osTicket web-console, and save the settings.

✔ Successfully updated configuration

Manage Plugin — Slack notifier

Configuration

Slack notifier
Readme first: <https://github.com/clonemeagain/osticket-slack>

Webhook URL:

Ignore when subject equals regex:
Auto delimited, always case-insensitive

Message Template:
The main text part of the Slack message, uses Ticket Variables, for what the user typed, use variable: %
{slack_safe_message}

Slack integration is completed.

Now let us create a ticket to test the slack alerts. Go to web browser to browse the URL -

You will see a window, click on option, and raise a complaint.

 [Support Center Home](#)

 [Open a New Ticket](#)

 [Check Ticket Status](#)

Welcome to the Support Center

In order to streamline support requests and better serve you, we utilize a support ticket system. Every support request is assigned a unique ticket number which you can use to track the progress and responses online. For your reference we provide complete archives and history of all your support requests. A valid email address is required to submit a ticket.

[Open a New Ticket](#)

[Check Ticket Status](#)

For test purpose we have created a ticket like below.

Open a New Ticket

Please fill in the form below to open a new ticket.

Contact Information

Email Address *

Full Name *

Phone Number

 Ext:

Help Topic

▼ *

Ticket Details

Please Describe Your Issue

Issue Summary *

Rich text editor toolbar: <> | ↑ | **A** | Aa | **B** | / | U | ~~S~~ | ☰ | 🖼️ | 📺 | 🗑️ | 🔗 | ✖️

This is a test to check slack alerts.

unsaved

Drop files here or [choose them](#)

[Create Ticket](#) [Reset](#) [Cancel](#)

After a successful ticket submission, the web panel will show up an acknowledgment note.

[Support Center Home](#) [Open a New Ticket](#) [Check Ticket Status](#)

✔ Support ticket request created

Group X,

Thank you for contacting us.

A support ticket request has been created and a representative will be getting back to you shortly if necessary.

Support Team

And you will get a notification in your slack channel like below -



osTicket-Alerts APP 11:09 PM

New Ticket #324743 created

Testing the Slack Alert from osTicket

Group X (ticketingsystemtest@gmail.com) in Maintenance Report a Problem

This is a test to check slack alerts.

via osTicket Slack Plugin | Today at 11:09 PM

9. osTicket Integration with LibreNMS

We will integrate LibreNMS alerts with osTicket so that any alert gets a ticket created automatically.

First, we need to generate the API keys from osTicket web-console, allowing the LibreNMS server's IP.

Go to the "Admin Panel > Manage > API", then click on the "ADD New API Key" button.

The screenshot shows the osTicket Admin Panel interface. At the top left is the osTicket logo. At the top right, it says "Welcome, apnic. | Agent Panel | Profile | Log Out". Below the logo is a navigation bar with tabs: Dashboard, Settings, Manage (selected), Emails, and Agents. Underneath the navigation bar is a secondary menu with icons for Help Topics, Filters, SLA, Schedules, API (selected), Pages, Forms, Lists, and Plugins. The main content area is titled "API Keys" and features a green "Add New API Key" button and a "More" dropdown menu. Below this is a table with the following data:

API Key	IP Address	Status	Date Added	Last Updated
<input type="checkbox"/> A82981863A2FB446841516A119EED6B6		Active	12/8/20	12/9/20 12:11 AM

Below the table, there are controls: "Select: All None Toggle" and "Page: [1] 2 3 4 5 6 7 8 9 10 »".

Input the IP address of the LibreNMS server, from where the alerts will ingest, check the "Can Create Tickets" option, and click the "Add Key" button.

OSTicket

Welcome, **apnic.** | [Agent Panel](#) | [Profile](#) | [Log Out](#)

Dashboard Settings **Manage** Emails Agents

[Help Topics](#) [Filters](#) [SLA](#) [Schedules](#) [API](#) [Pages](#) [Forms](#) [Lists](#) [Plugins](#)

Add New API Key

API Key is auto-generated. Delete and re-add to change the key.

Status: Active Disabled

IP Address:

Services: Check applicable API services enabled for the key.

Can Create Tickets (XML/JSON/EMAIL)

Can Execute Cron

Internal Notes: Be liberal, they're internal

Copyright © 2006-2020 link3 All Rights Reserved.

API key is created.

OSTicket

Welcome, **apnic.** | [Agent Panel](#) | [Profile](#) | [Log Out](#)

Dashboard Settings **Manage** Emails Agents

[Help Topics](#) [Filters](#) [SLA](#) [Schedules](#) [API](#) [Pages](#) [Forms](#) [Lists](#) [Plugins](#)

Successfully added an API key.

API Keys

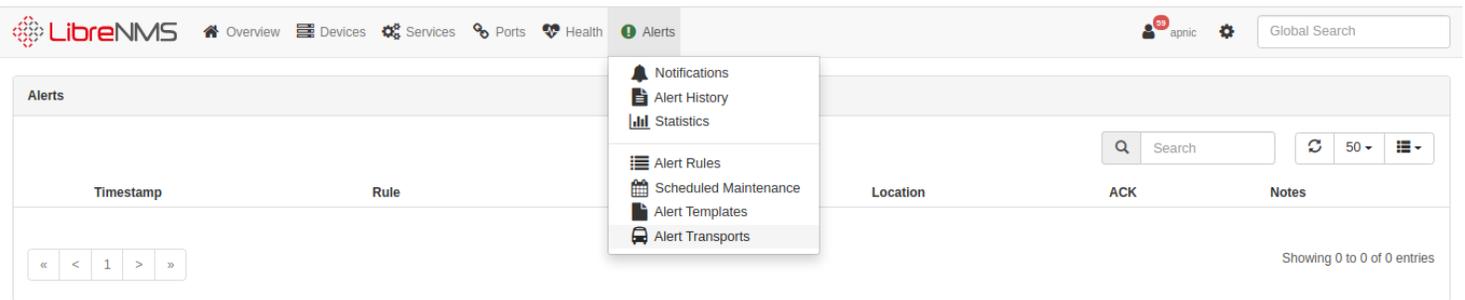
API Key	IP Address	Status	Date Added	Last Updated
<input type="checkbox"/> A82981863A2FB446841516A119EED6B6		Active	12/8/20	12/9/20 12:11 AM
<input type="checkbox"/> 54376F47E37C7EF6755EEFDD44C85A6F	192.168.1.1	Active	12/9/20	12/9/20 5:57 PM

Select: All None Toggle

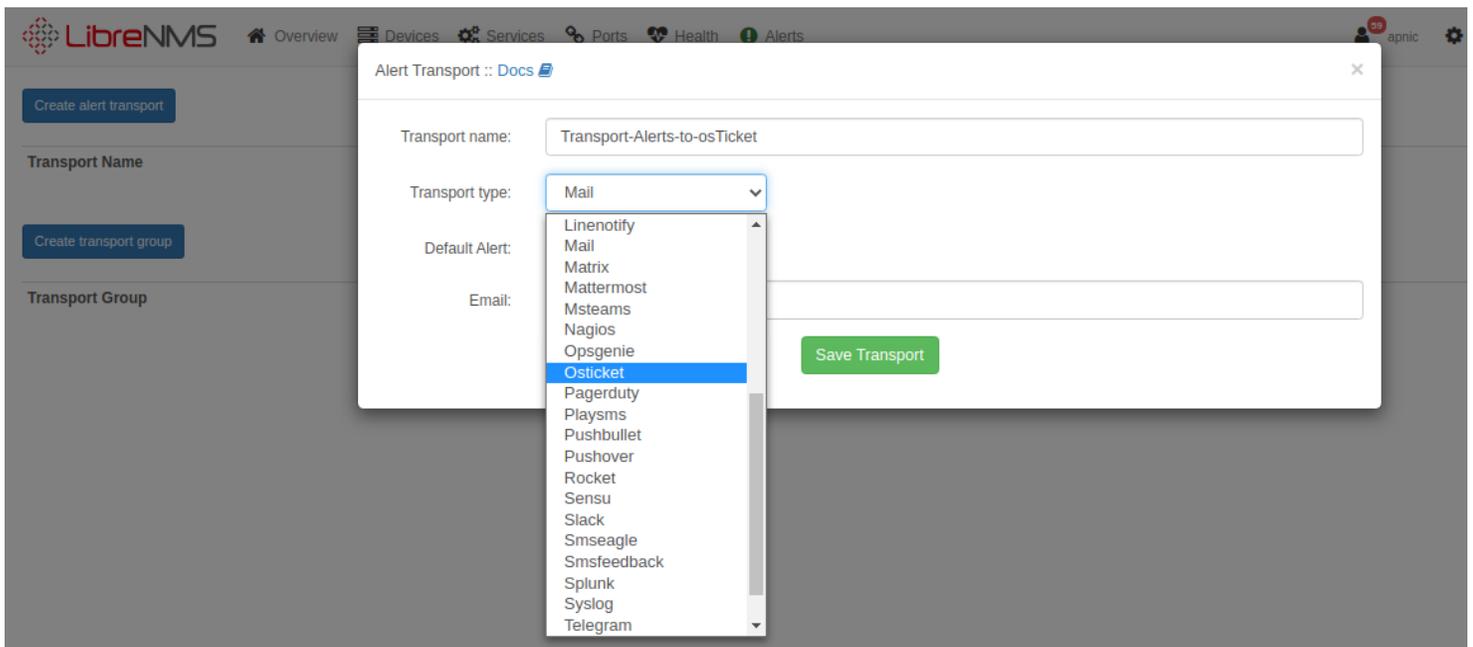
Page: [1] 2 3 4 5 6 7 8 9 10 »

Copyright © 2006-2020 link3 All Rights Reserved.

Next, go back to LibreNMS web-console, and define the "Alert Transports" from Alerts tab.

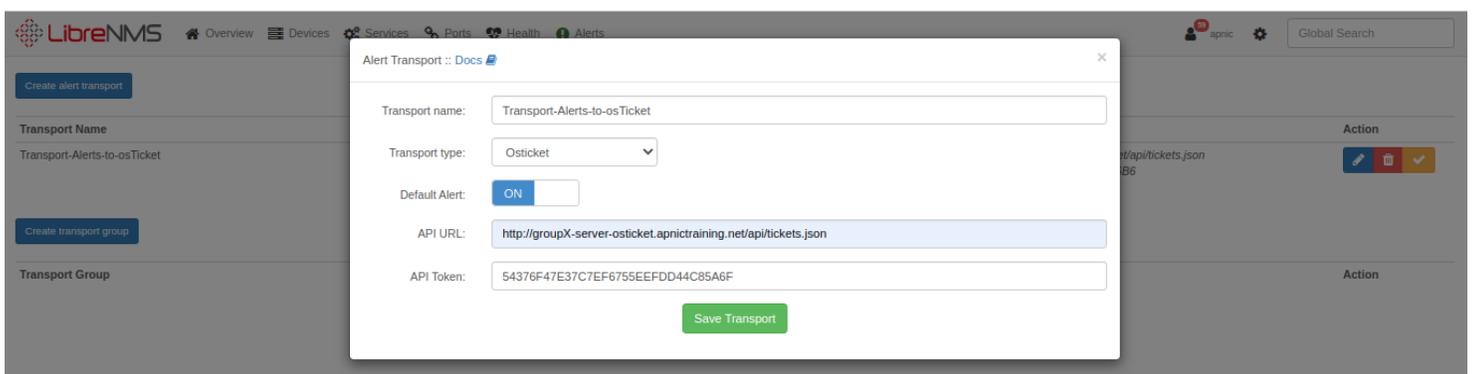


Click on the "Create alert transport" option, a pop-up window will come up, fill up the gap providing "Transport name", and then select Osticket from the drop-down menu, turn on the Default Alert option by clicking the Off/On button.



Now go back to the osTicket web-console, and copy the API Key from there, and input the key into the "API Token" and place the API URL as below.

Note: Replace the server name with your server name accordingly.



Click the "Save Transport" button.

Next, go to the LibreNMS server CLI mode, navigate `Osticket.php` file inside `/opt/librenms/LibreNMS/Alert/Transport/` directory. Edit the line 41 and 42, according to below guideline.

```
$ sudo vim /opt/librenms/LibreNMS/Alert/Transport/Osticket.php
```

Default file will look like this:

```
$protocol = [  
    'name' => 'LibreNMS',  
    'email' => $email,  
    'subject' => ($obj['name'] ? $obj['name'] . ' on ' . $obj['hostname'] : $obj['title']),  
    'message' => strip_tags($obj['msg']),  
    'ip' => $_SERVER['REMOTE_ADDR'],  
    'attachments' => [],  
];
```

After change the file will look like this:

```
$protocol = [  
    'name' => 'LibreNMS-Alerts-Test-to-osTicket',  
    'email' => 'training@apnic.net',  
    'subject' => ($obj['name'] ? $obj['name'] . ' on ' . $obj['hostname'] : $obj['title']),  
    'message' => strip_tags($obj['msg']),  
    'ip' => $_SERVER['REMOTE_ADDR'],  
    'attachments' => [],  
];
```

Note: If we dont change the email ID (line 42) here in this file, osTicket platform will show an error as below.

API Error (400)

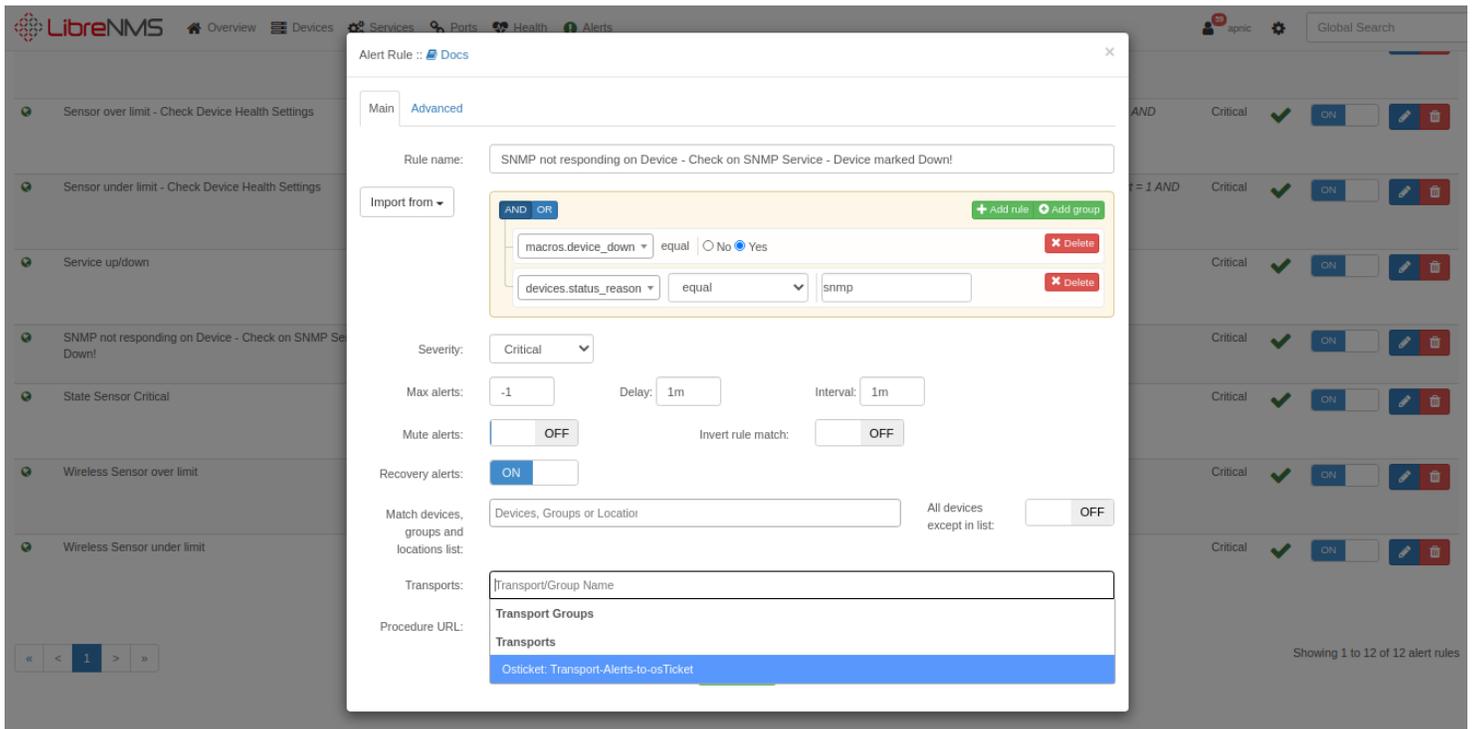
Unable to create new ticket: validation errors: user: Incomplete client information

[54376F47E37C7EF6755EEFDD44C85A6F]

Log Date: *Wednesday, December 9, 2020 at 5:04 PM* **IP Address:** *192.168.1.1*

One last thing to do before the test:

- Go to the "Alert Rules" option from the Alerts tab on LibreNMS.
- Edit the "SNMP not responding on Device - Check on SNMP Service - Device marked Down!" rules to define the "Transports" option sync to "Transport-Alerts-to-osTicket," so that if the monitored SNMP service stopped responding, it would generate the alerts, transport it to osTicket and then you will see a ticket is created and send that alert into the slack channel.



The API integration from LibreNMS to osTicket is done. Its time to check if the alerts are transported to osTicket, and get the ticket alert at slack channel as well. Stop the SNMPD service for a while at LibreNMS server.

Where you will see, few output like below.

Ticket created at osTicket.



Welcome, **apnic**. | [Admin Panel](#) | [Profile](#) | [Log Out](#)

Dashboard Users Tasks **Tickets** Knowledgebase

[Open](#) [My Tickets](#) [Closed](#) [Search](#) [New Ticket](#)

[advanced] Sort

Open

Ticket	Last Updated	Subject	From	Priority	Assigned To
<input type="checkbox"/> 810342	12/9/20 5:26 PM	SNMP not responding on Device - Check ...	LibreNMS-Alerts-Test-to-osTicket	Normal	
<input type="checkbox"/> 915887	12/9/20 5:25 PM	SNMP not responding on Device - Check ...	LibreNMS-Alerts-Test-to-osTicket	Normal	

Alert exported at Slack channel.

New Ticket [#810342](#) created

SNMP not responding on Device - Check on SNMP Service - Device marked Down!
 on [192.168.1.2](#)

LibreNMS-Alerts-Test-to-osTicket (training@apnic.net) in Support __

```
Nothing has happened Severity: critical Time elapsed: 4m 49s Timestamp:
2020-12-09 11:25:04 Unique-ID: 28 Rule: SNMP not responding on Device -
Check on SNMP Service - Device marked Down! Faults: #1: sysObjectID =>
.1.3.6.1.4.1.8072.3.2.10; sysDescr => Linux
http://groupX-server.apnictraining.net 3.10.0-1127.el7.x86_64 #1 SMP Tue Mar
```

[Show more](#)

 via osTicket Slack Plugin | Today at 5:26 PM

=====
End of Lab
 =====