Table of Contents

Node Monitoring Tools for Stewards

Plugin Manager

Currently, indy-node emits different events via the Plugin Manager when certain criteria are met. The Plugin Manager tries to import all pip packages which names start with "sovrinnotifier*". Each of these packages is required to expose "send_ message" interface which is used to pass the event with the associated message to the package for further handling.

The Plugin Manager code is located at here.

Events Emitted

- `nodeRequestSpike` : `NodeRequestSuspiciousSpike`
- `clusterThroughputSpike` : `ClusterThroughputSuspiciousSpike`
- `clusterLatencyTooHigh` : `ClusterLatencyTooHigh`
- `nodeUpgradeScheduled` : `NodeUpgradeScheduled`
- `nodeUpgradeComplete` : `NodeUpgradeComplete`
- `nodeUpgradeFail` : `NodeUpgradeFail`
- `poolUpgradeCancel` : `PoolUpgradeCancel`

Email Plugin

Email Plugin

Prerequisites

SMTP server must be running on localhost.

Install SMTP server (if you don't have one already)

The most simple way on Ubuntu is to use sendmail:

```
$ sudo apt-get install sendmail
```

To check that it's working execute:

```
echo "Subject: sendmail test" | sendmail -v youremail@example.com -f alert@noreply.com
```

If you get a email on your youremail@example.com then sendmail is working.

Install

```
$ pip3 install sovrinnotifieremail
$ Add SOVRIN_NOTIFIER_EMAIL_RECIPIENTS=youremail@example.com to your /etc/environment
```

You are required to set system environment variable `SOVRIN_NOTIFIER_EMAIL_RECIPIENTS`.

Configuration

The package depends on two environment variables:

- `SOVRIN_NOTIFIER_EMAIL_RECIPIENTS` (required)
- `SOVRIN_NOTIFIER_EMAIL_SENDER` (optional)

SOVRIN_NOTIFIER_EMAIL_RECIPIENTS

`SOVRIN_NOTIFIER_EMAIL_RECIPIENTS` should be a string in a format of:

```
recipient1@adress.com [optional list of events the recipient is going to get],recipient2@adress.com [event list], ...
```

If no list was provided the recipient is going to get notifications for all events.

Example:

```
steward1@company.com event1 event2, steward2@company.com, steward3@company.com event3
```

This way steward1 is going to get notifications for event1 and event2, steward2 is going to get all possible notifications and steward3 is going to get notifications for event3 only.

The current list of events can be found above.

SOVRIN_NOTIFIER_EMAIL_SENDER

By default every email notification is going to be from alert@noreply.com. You can change this by setting `SOVRIN_NOTIFIER_EMAIL_SENDER`. May be useful for email filters.

Email delivery frequency

By default you will not get a email with the same topic more than once an hour. This is defined by `SILENCE_TIMEOUT`. It can be overridden by setting `SOVRIN_NOTIFIER_SILENCE_TIMEOUT` environment variable. Emails regarding update procedure are always delivered.

AWS SNS Plugin

AWS SNS Plugin

Prerequisites

- A AWS SNS topic created with permissions to publish to it.
- A installed Sovrin Validator instance.

Setup

Install the python package for sovrin-notifier-awssns. This should be only be installed using pip3.

```
    pip3 install sovrinnotifierawssns
```

Configuration

To configure AWS Credentials you will need to know the values for: `aws_access_key_id` and `aws_secret_access_key`. Follow the steps documented here Boto3 Configuring Credentials.

Use either of the following ways:

- Environment variables `AWS_ACCESS_KEY_ID, AWS_SECRET_ACCESS_KEY`
- Shared credential file (~/.aws/credentials)
- Boto2 config file (/etc/boto.cfg and ~/.boto)

Configure AWS Region you will need to know the value where the SNS Topic is hosted e.g. `us-west-1, us-west-2, sa-east-1`

To achieve this:

- Set a Environment variable `AWS_DEFAULT_REGION`
- Set region using file (~/.aws/config)

Define environment variable `SOVRIN_NOTIFIER_AWSSNS_TOPICARN` on the Validator and set valid AWS SNS TopicARN as the value.

Events

Events that cause a notification:

```
    NodeRequestSuspiciousSpike,
    ClusterThroughputSuspiciousSpike,
    ClusterLatencyTooHigh,
    NodeUpgradeScheduled,
    NodeUpgradeComplete,
    NodeUpgradeFail,
    PoolUpgradeCancel
```

Hints

The home directory for the account that runs `sovrin-node.service` on a Validator is `/home/sovrin/`. So the aws credentials/config files must be created in `/home/sovrin/.aws` folder

To set an environment variable on the Validator you must add it to the file `/home/sovrin/.sovrin/sovrin.env` and restart the Validator. The TopicARN must be defined in this file.

To restart the Validator on a Ubuntu system you must execute the command `sudo systemctl restart sovrin-node.service` while not logged in as a `sovrin ` user.

Example

This simple script will complete the setup, assuming that the sovrinnotifierawssns package is already installed:

```
#!/bin/bash
sudo mkdir /home/sovrin/.aws
sudo sh -c "printf \"[default]\nregion=us-west-2\" >
/home/sovrin/.aws/config"
sudo sh -c "printf \" .[default]\naws_access_key_
id=AKIAIGKGW3CKRXKKWPZA\naws_secret_access_key=<YOUR_
SECRET_KEY>\" > /home/sovrin/.aws/credentials"
sudo sh -c "printf \"SOVRIN_NOTIFIER_AWSSNS_TOPICARN-
N=arn:aws:sns:us-west-2:034727365312:validator-health-mon-
itor-STN\" >> /home/sovrin/.sovrin/sovrin.env"
sudo chown -R sovrin:sovrin /home/sovrin/.aws /home/sov-
rin/.sovrin/sovrin.env
sudo systemctl restart sovrin-node
```