Monitoring Bot Setup

Overview of How To Set-up and Run an oSnap Monitoring Bot The oSnap Monitoring Bot monitors all transactions to your oSnap module. It also verifies that proposed transactions are valid according to the rules of the oSnap module (ie. the snapshot quorum and voting period meet the oSnap rules). Any unverified proposed transactions should be reviewed further and possibly disputed.

All oSnap propsals are already monitored by bots run by UMA's core team and a decentralized group of human verifiers. You may wish to run this bot so you can monitor your oSnap module yourself.

This tutorial walks through an example of how to setup the oSnap monitoring bot from UMA'sprotocol repository that watches for emitted contract events and sends alerts.

The Node package method described below is also available onYoutube .

Installation

Docker method

The instructions assume you haveDocker installed and its server daemon is running.

Get the latest UMA protocol image that among others includes the required monitoring bot:

```

Copy dockerpullumaprotocol/protocol:latest

```

Node package method

The instructions assume you have already the latest long term support version ofNode.js .

Initialize new project directory and install@uma/monitor-v2 package that contains the oSnap monitoring bot:

```

Copy mkdirmonitor-osnap cdmonitor-osnap npminit-y npminstall@uma/monitor-v2

```

Basic configuration

All configuration for the monitor bot should be set in a .env file in your working directory. Please see basic configuration variables below:

```

Copy

# Note on formatting: Do not enclose variable values in quotes as this is not supported when running the bot in Docker container.

# Name of the bot identifier as shown in alerts.

BOT_IDENTIFIER=oSnapMonitorBot

# the address of the deployed Optimistic Governor that this bot will monitor. This does not necessarily need to be deployed through factory.
```

OG_ADDRESS=

network number (ie. Ethereum mainnet = 1)

CHAIN_ID=

a single RPC node URL replacing X in variable name with CHAIN_ID. This is considered only if matching NODE_URLS_X in the advanced config section is not provided.

NODE_URL_X=

boolean enabling/disabling monitoring transactions proposed (false by default).

TRANSACTIONS_PROPOSED_ENABLED=true

boolean enabling/disabling monitoring transactions executed (false by default).

TRANSACTIONS_EXECUTED_ENABLED=true

boolean enabling/disabling monitoring proposal executed (false by default).

PROPOSAL_EXECUTED_ENABLED=true

boolean enabling/disabling monitoring proposal deleted (false by default).

PROPOSAL_DELETED_ENABLED=true

boolean enabling/disabling monitoring set collateral and bond amount (false by default).

SET_COLLATERAL_BOND_ENABLED=true

boolean enabling/disabling monitoring set rules (false by default).

SET_RULES_ENABLED=true

boolean enabling/disabling monitoring set liveness (false by default).

SET_LIVENESS_ENABLED=true

boolean enabling/disabling monitoring set identifier (false

by default).

SET_IDENTIFIER_ENABLED=true

boolean enabling/disabling monitoring set escalation manager (false by default).

SET_ESCALATION_MANAGER_ENABLED=true

used to simulate proposed transaction execution on Tenderly. If any of these are missing, the bot will skip the simulation.

TENDERLY_USER= TENDERLY_PROJECT= TENDERLY_ACCESS_KEY=

```

Running the bot

Docker method

Instruct docker to run the oSnap monitor bot by appendingCOMMAND environment variable to the same.env file where other configuration is stored:

```

Copy

Command to run when starting docker container:

COMMAND=node./packages/monitor-v2/dist/monitor-og/index.js

```

Start the monitoring bot from the same working directory where the.env configuration file is located:

```

Copy dockerrun-d--env-file.env--nameosnap-monitor--rmumaprotocol/protocol:latest

```

This should start the oSnap module monitoring bot in a looping mode detached from console where all configured events will be sent to the provided notification channels.

Stop the running bot container with:

```

Copy dockerstop-t0osnap-monitor

```

Node package method

Start the monitoring bot from the root of your project directory where node package was installed and.env configuration file is stored:

```

Copy nodenode_modules/@uma/monitor-v2/dist/monitor-og/index.js

```
```

This starts the oSnap module monitoring bot in a looping mode where all configured events will be logged in the console.

Previous Migrate to oSnap Safe AppNext Setting Custom Bond and Liveness Parameters Last updated2 months ago On this page * Installation * Basic configuration * Running the bot