

# Part 2. StackStorm ハンズオン 障害自動検知と自動修復 (AR)

～ Sensu を活用した監視と障害検知 ～



大山 裕泰 / DMM.com ラボ

2017/11/30 InternetWeek 2017 運用自動化ハンズオン ～StackStormで実践するインフラ運用革命～

# 免責事項

- 第三者の製品・サービスについて、特定の製造者やサービス提供者につき、製品やサービスを評価するものではありません。
- 当社事例は、あくまで当社事案であり各社のシステム・サービス要件等によって、機能、パフォーマンスその他の面で該当しない場合があります。
- 本プレゼンテーションは当社構築した StackStorm に関する技術者の現時点での感想に基づいています。
- 無断複製・転載を禁じます。

# ハンズオン全体の目的

StackStorm Exchange を活用した AR の実現方法を体験する

- Sensu を活用した監視と障害検知
- Napalm を活用した vSRX へのオペレーション

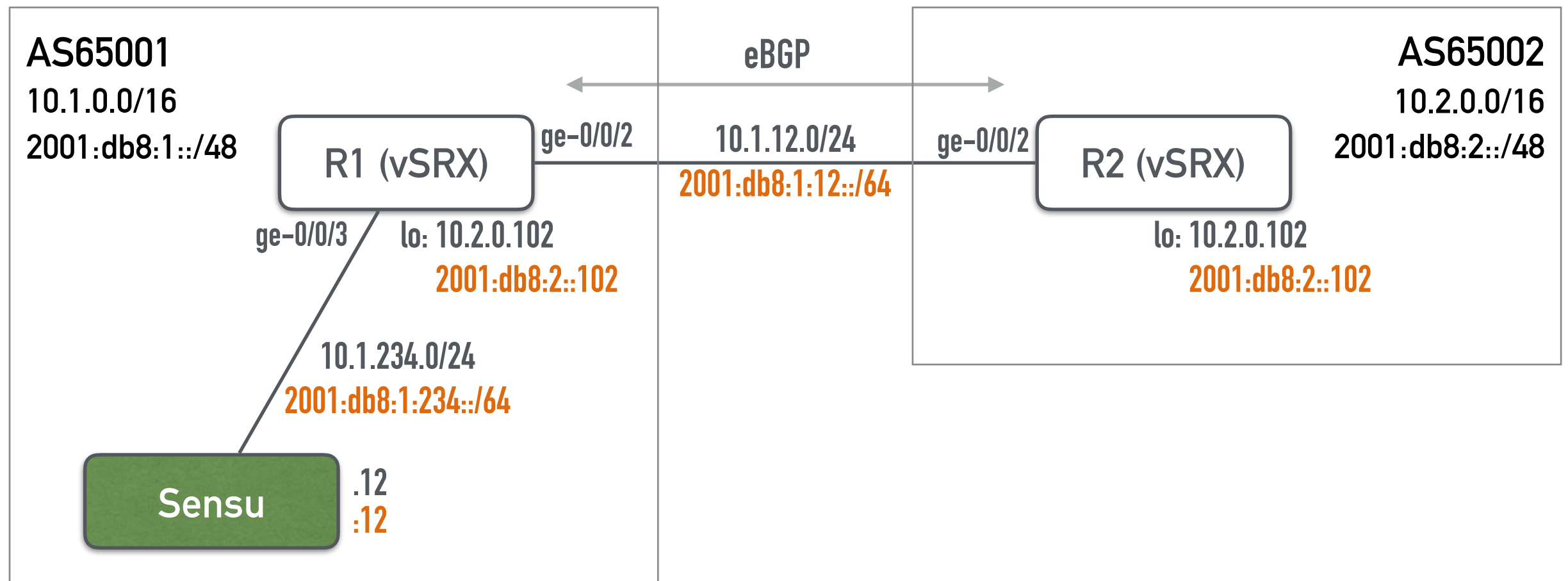
# 実施内容

1. Sensu を活用した監視と障害検知 (担当：大山)
2. 障害検知からの vSRX の AR (担当：萬治)
3. WorkFlow による複雑なオペレーション (担当：萬治)

# ハンズオン実施環境

AS65001 のエンジニアという想定で以下が設定されている

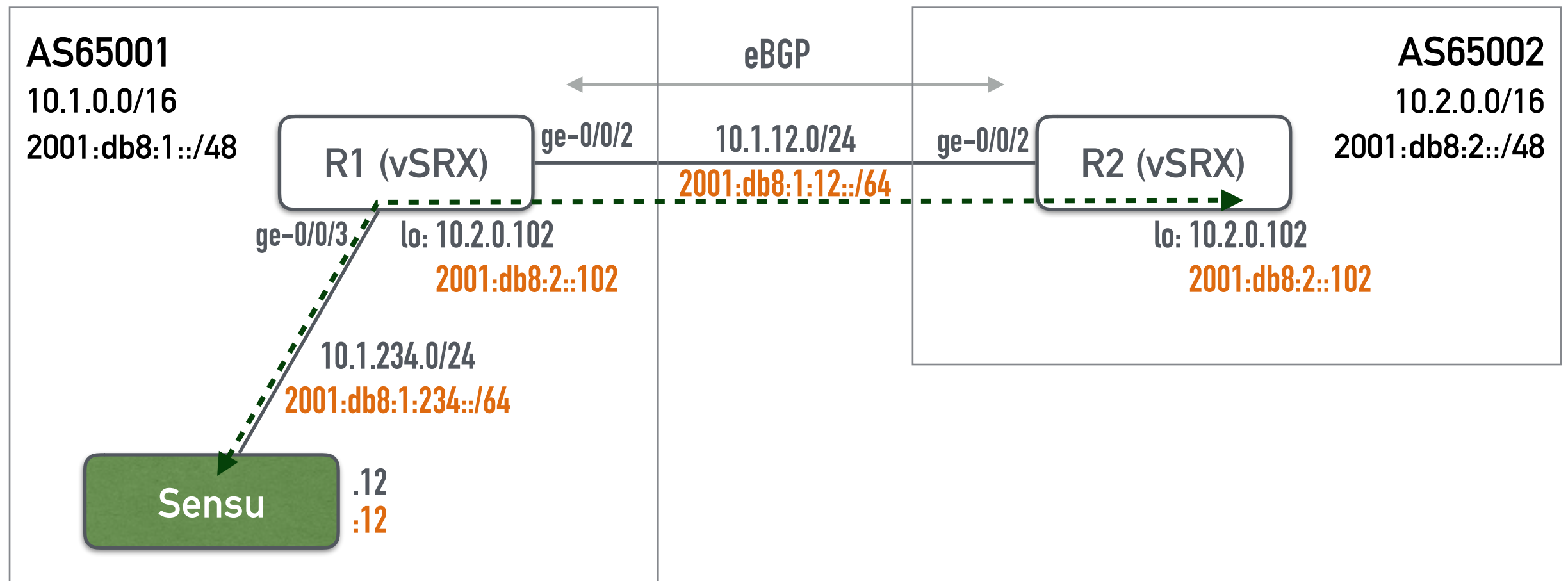
- AS65002 とのピア設定
- Sensu を使って R2 の lo0 に対して疎通監視



ハンズオン環境のネットワーク構成

# ハンズオン実施概要

ge-0/0/2 の I/F がダウンした際 Sensus で障害を検知し  
検知した障害を StackStorm に通知・ハンドリングする



ハンズオン環境のネットワーク構成

## Part.1

# Sensu を活用した監視と障害検知

# モニタリング関連のプラグイン

様々なモニタリングシステム・サービスと連携可能

check_mk	datadog	dripstat
icinga2	mmonit	nagios
newrelic	opsgenie	orion
<b>sensu</b>	zabbix	

2017/11 時点で公開されているモニタリング関連 pack 一覧



# モニタリング関連のプラグイン

様々なモニタリングシステム・サービスと連携可能

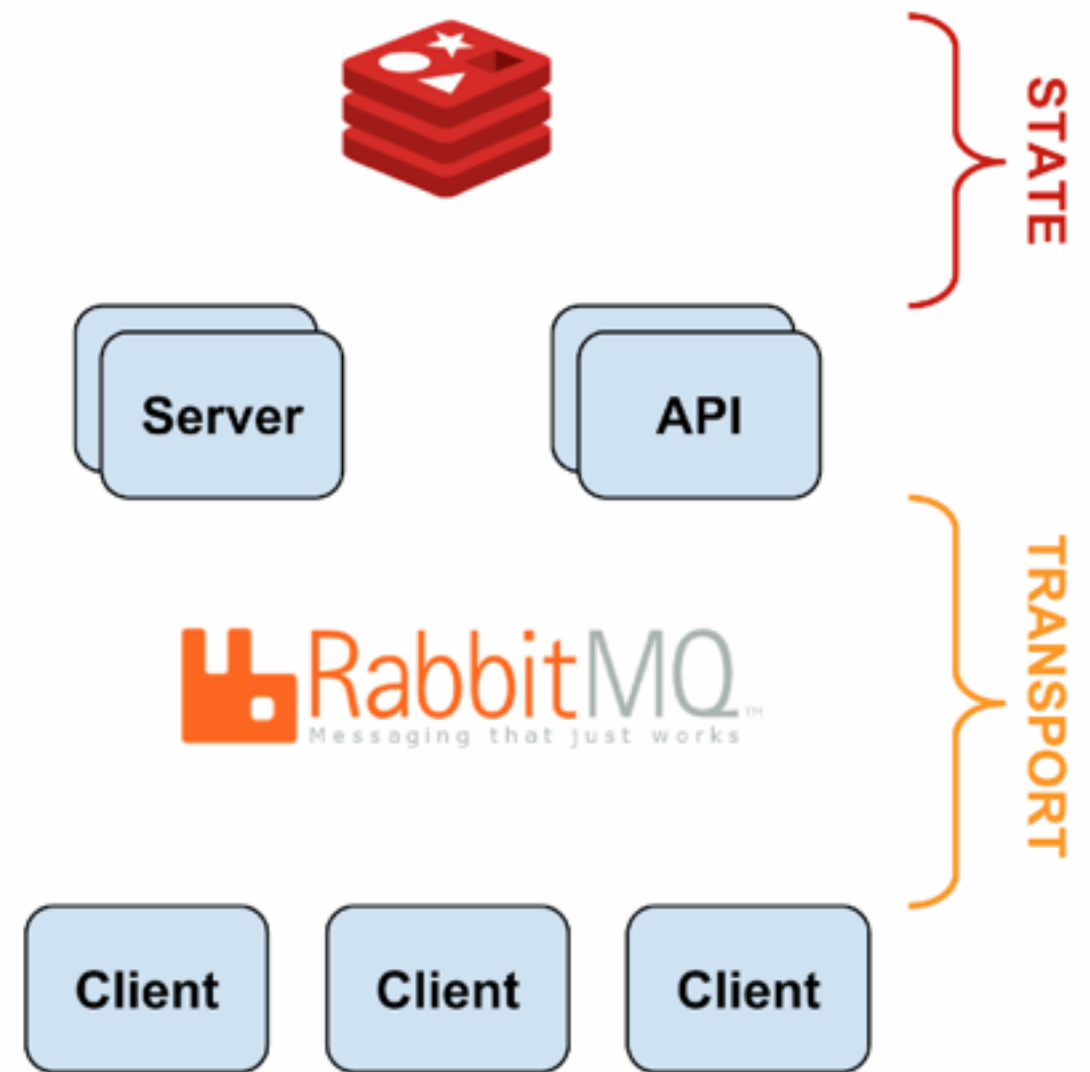
```
$ st2 pack install sensu
```

check_mk	datadog	dripstat
icinga2	mmonit	nagios
newrelic	opsgenie	orion
<b>sensu</b>	zabbix	

2017/11 時点で公開されているモニタリング関連 pack 一覧

# Sensu 概要

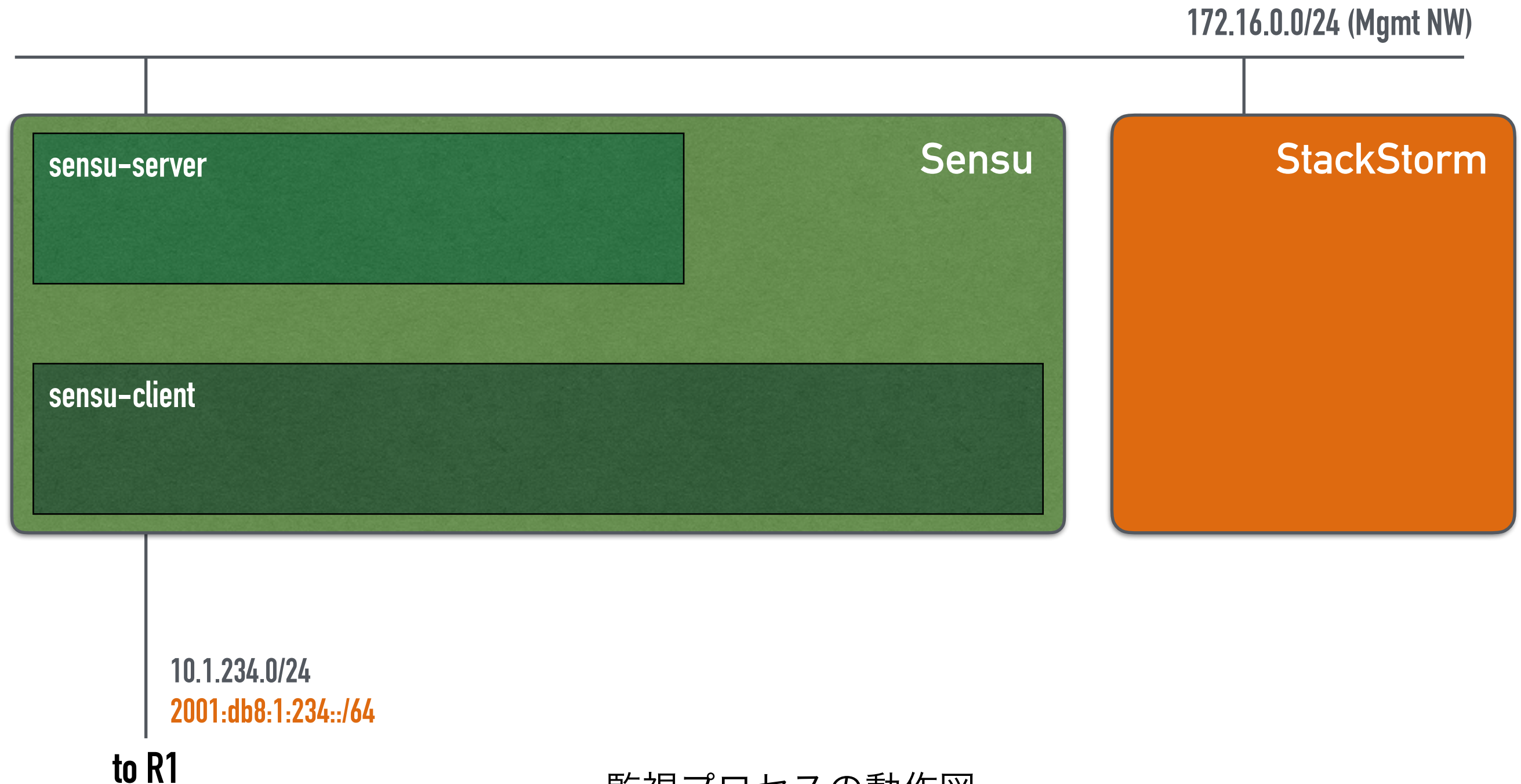
Sensu-Server の指示に基づき Sensu-Client が  
監視・イベント通知を行う



(出典： [Sensu Architecture - Sensu Documentation](#))

# Sensu による監視プロセス

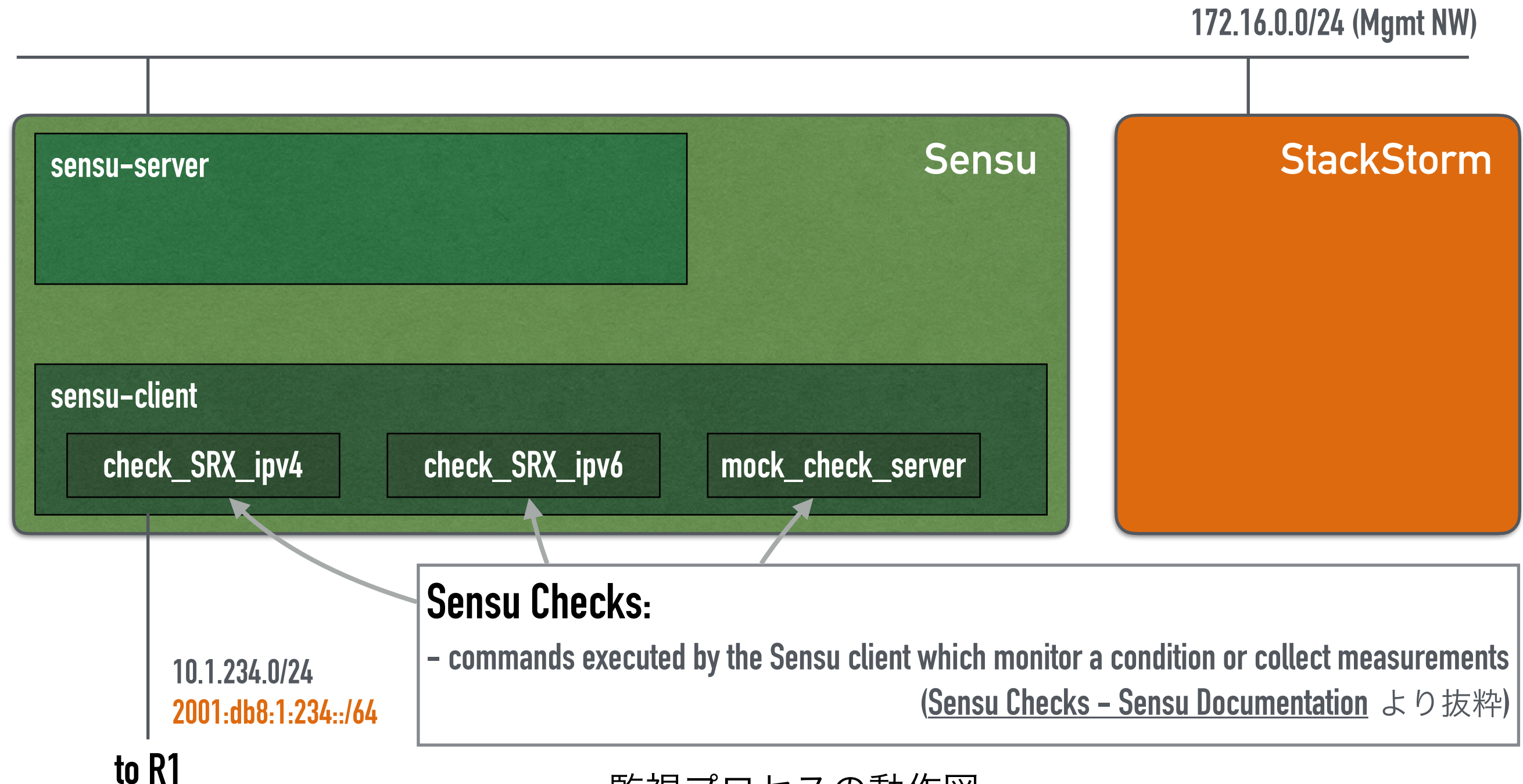
sensu-client が BGP ピアの確立を確認



監視プロセスの動作図

# Sensu による監視プロセス

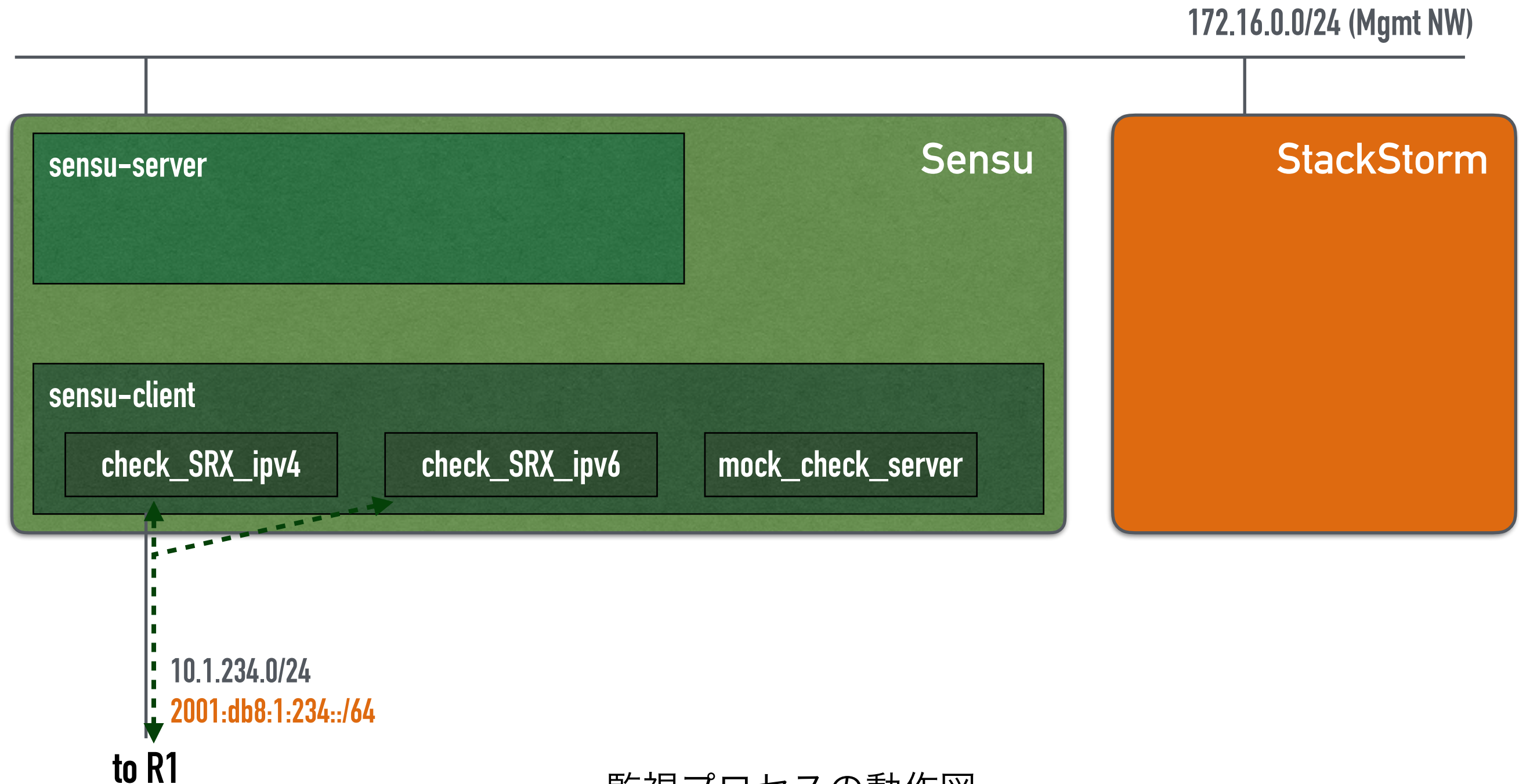
sensu-client が BGP ピアの確立を確認



監視プロセスの動作図

# Sensu による監視プロセス

sensu-client が BGP ピアの確立を確認

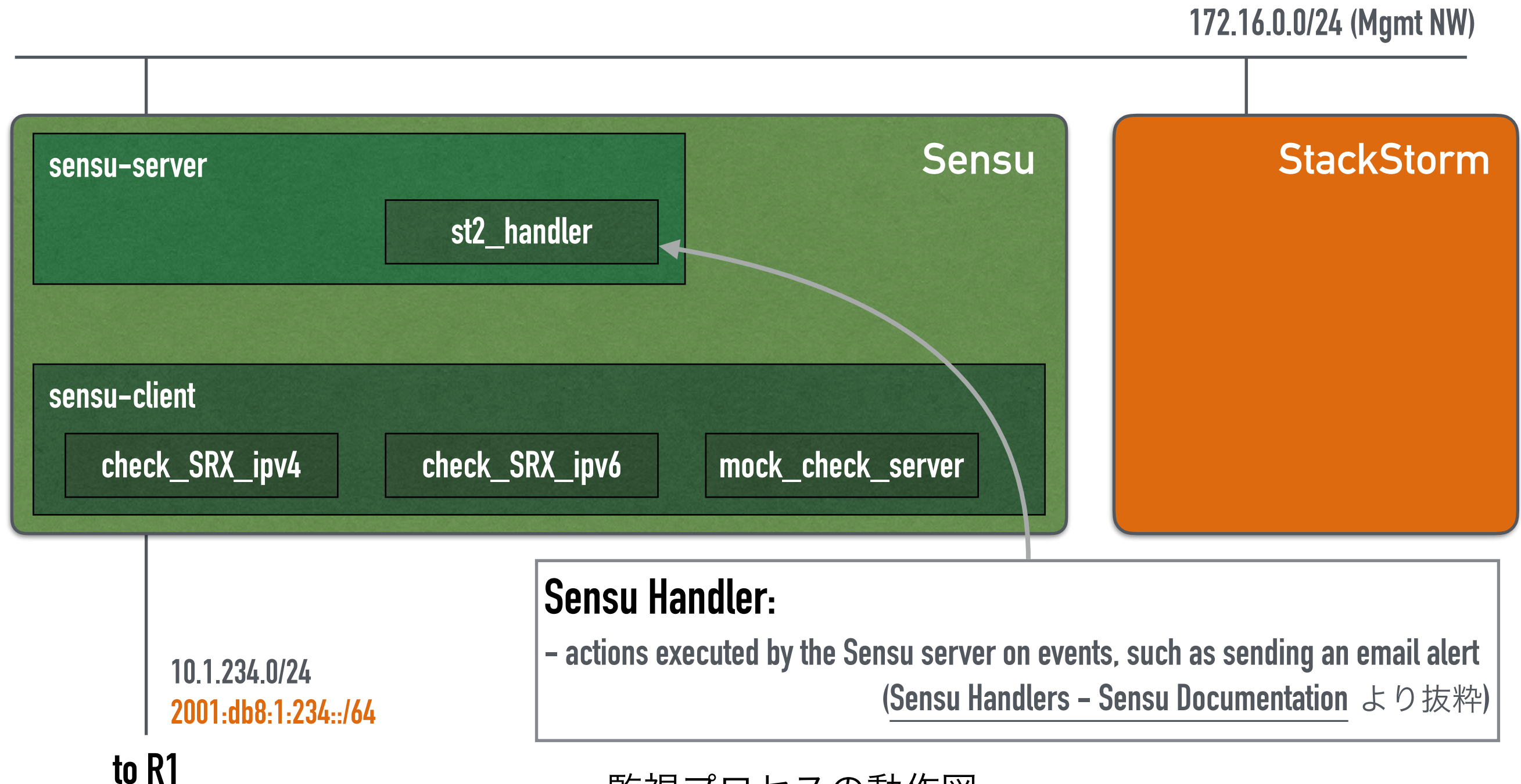


監視プロセスの動作図



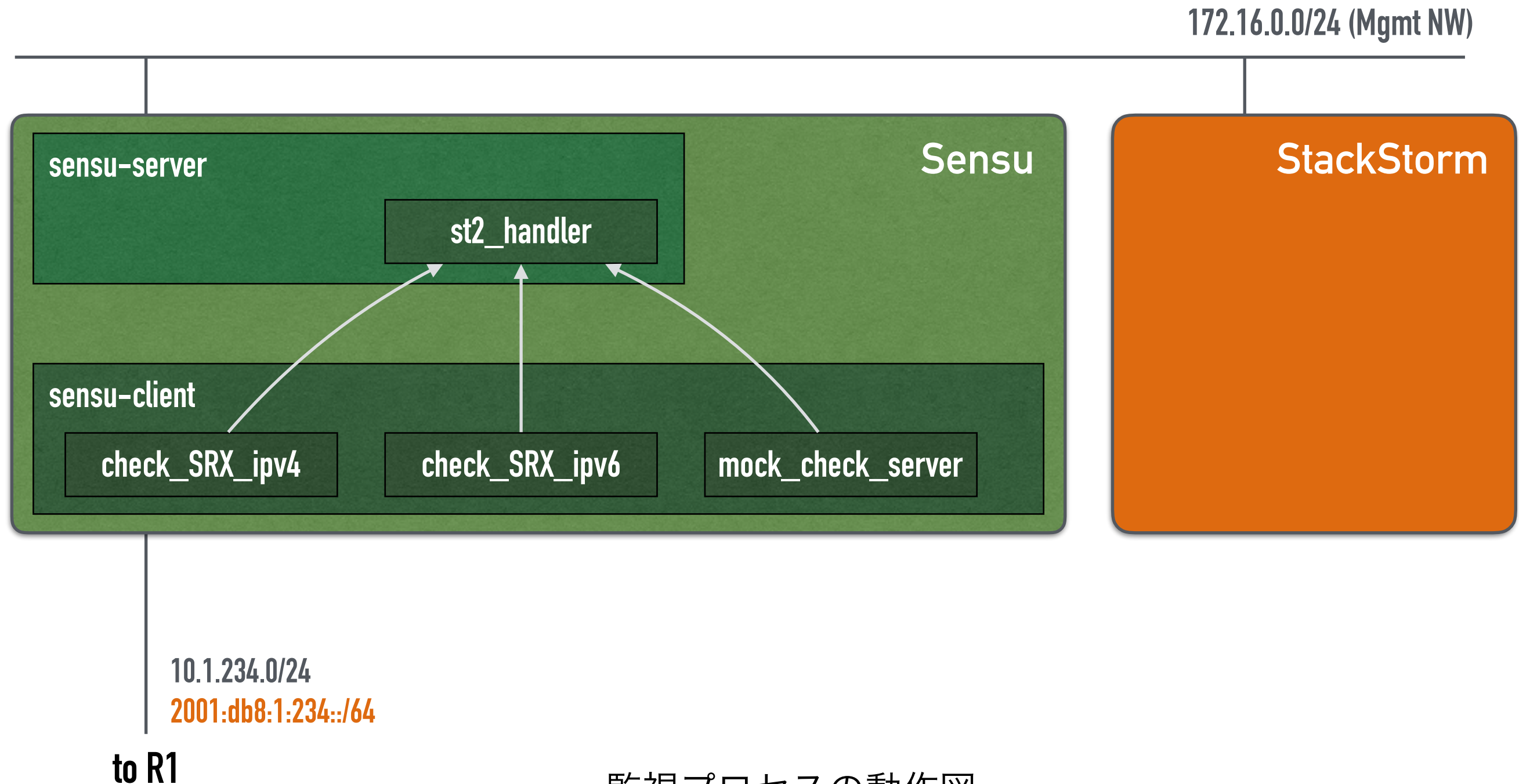
# Sensu による監視プロセス

sensu-client が BGP ピアの確立を確認



# Sensu による監視プロセス

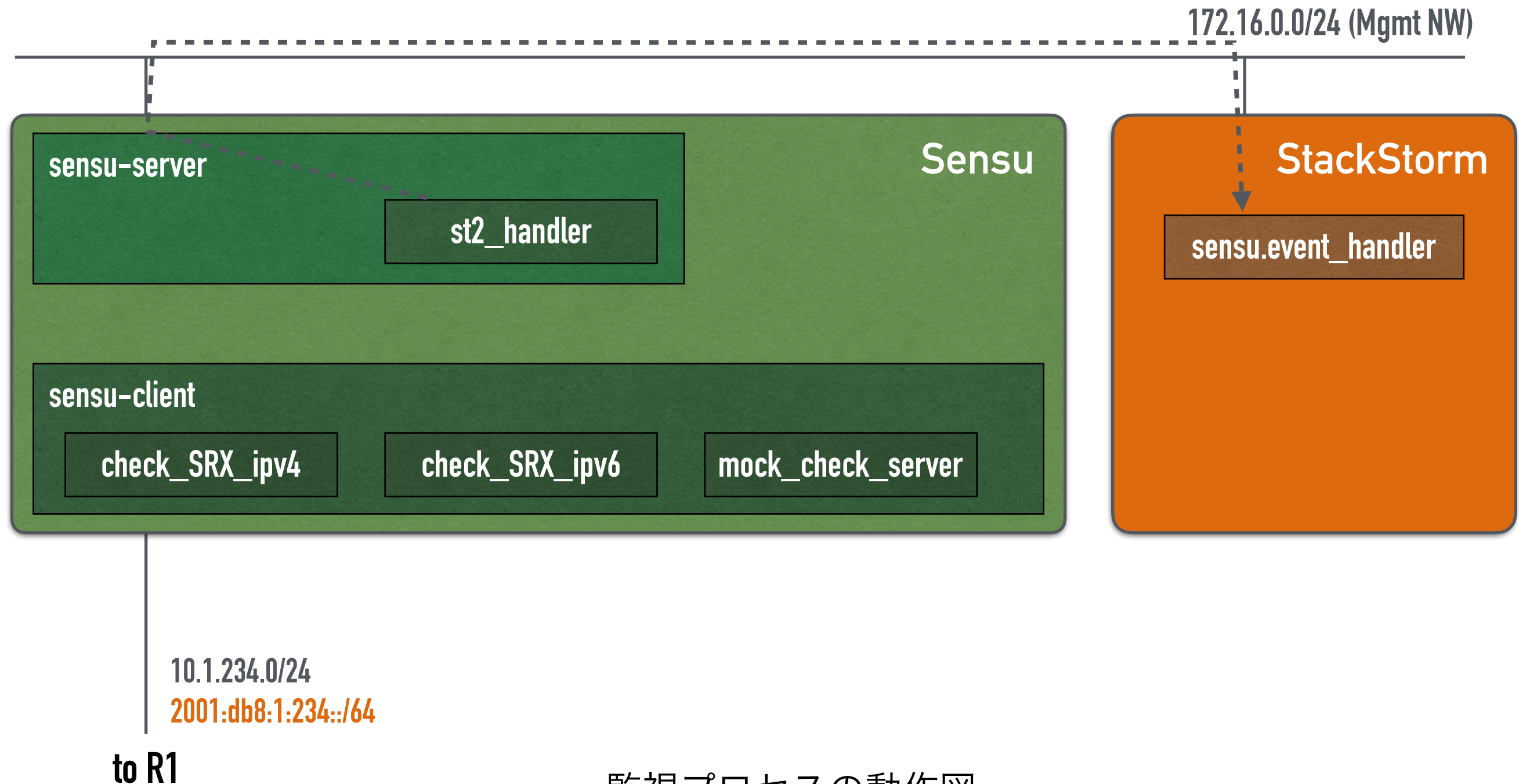
sensu-client が BGP ピアの確立を確認



監視プロセスの動作図

# Sensu による監視プロセス

sensu-client が BGP ピアの確立を確認

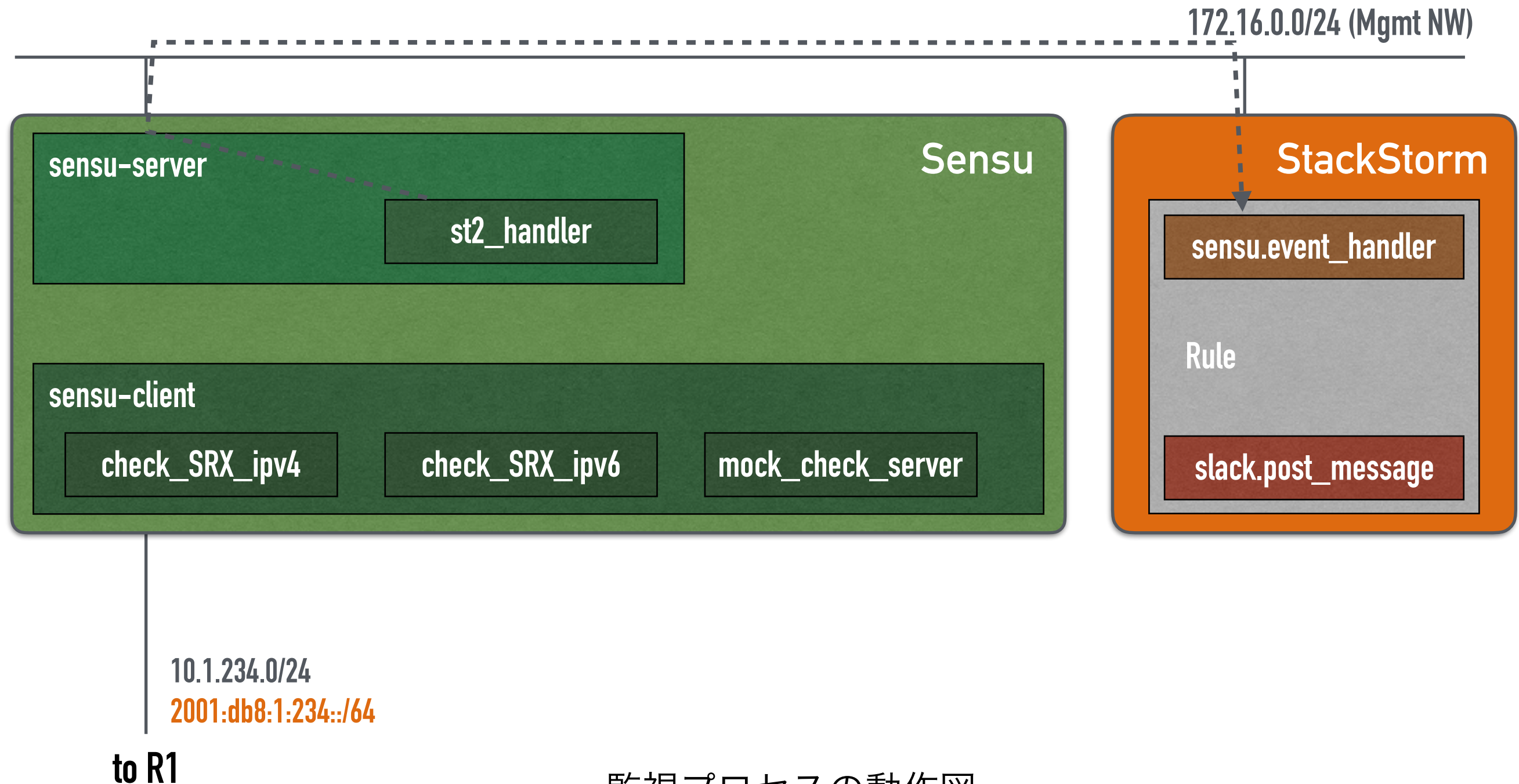


監視プロセスの動作図



# Sensu による監視プロセス

sensu-client が BGP ピアの確立を確認



監視プロセスの動作図

# WEB-UI

- `https://< VM の IP>:3000/`

# 設定方法

- イベントハンドリングのための Rule の設定

## まとめ

Sensu と Sensu を利用した監視・イベント検知を学ぶ

continuing to Part.2

# 障害検知からの vSRX の AR