

AWS上での監視サーバー(Zabbix)構築【8.監視登録(SNMP Trap)】

AWS



AWS上での監視サーバー(Zabbix)構築【8.監視登録(SNMP Trap)】

2021.10.08 2021.09.16

監視サーバーをAWS上で構築し、CML上のネットワーク機器/サーバーを監視します。監視ソフトウェアはZabbixを利用します。

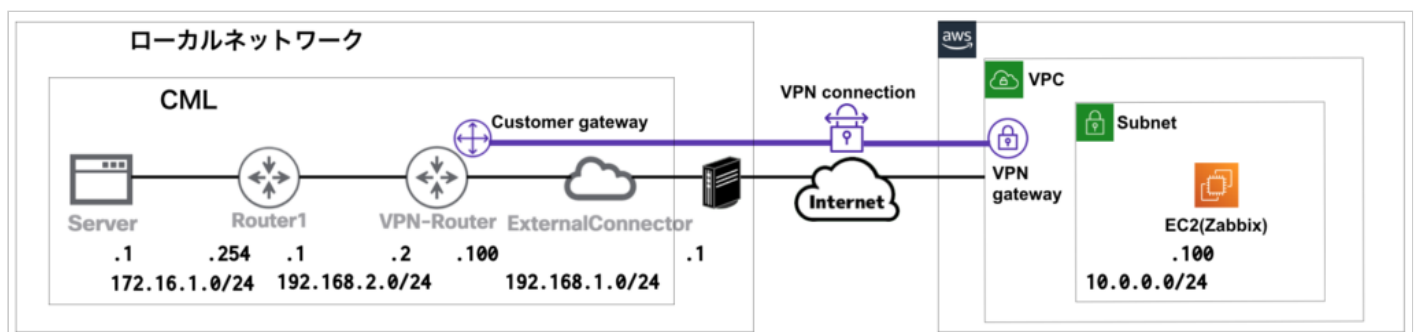
[【前回】AWS上での監視サーバー\(Zabbix\)構築【7.監視登録\(SNMP\)】](#)

[【次回】AWS上での監視サーバー\(Zabbix\)構築【9.メール通知\(SNS\)】](#)

ネットワーク構成

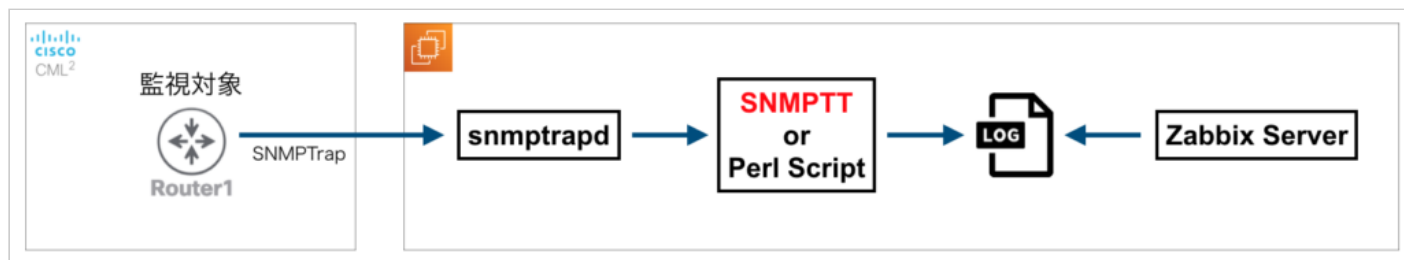
下記のネットワーク環境を構築し、AWS上のEC2(Zabbixサーバー)から、CML上のネットワーク機器/サーバーを監視できるようにしていきます。

[【参考】AWSサイト間VPNの構築（1.AWSの基本設定）](#)



SNMP Trap監視の仕組み

ZabbixでのSNMPTrapを利用する監視は、下記の仕組みで行われます。



1. 監視対象機器がSNMPTrapを送出する。
2. サーバー上のsnmptrapdがSNMPTrapを受け取る。
3. SNMPTT or PerlScriptがSNMPTrapを加工してログファイルへ書き出す。
4. ZabbixServerがログファイルを読み取り、条件に沿って障害として検知する。

ここでは、SNMPTTを利用した監視方法を説明します。

SNMPTTのインストール・設定

EPELレポジトリのインストール

SNMPTTの利用に必要なEPELというリポジトリをインストールします。

AmazonLinuxの場合、yumでのインストールはエラーとなるため、“amazon-linux-extras”を利用します。

```
sudo amazon-linux-extras install -y epel
```

明示的にEPELを指定した時のみ、EPELリポジトリを利用するようにします。

```
sudo cp -p /etc/yum.repos.d/epel.repo /etc/yum.repos.d/epel.repo.backup
sudo vi /etc/yum.repos.d/epel.repo
```

(下記を編集)

[epel]

enabled=1

↓

enabled=0

※参考※

yumを利用してインストールしようとする则下記のエラーが表示され、"amazon-linux-extras"を利用するように促されます。

```
[ec2-user@ip-10-0-0-100 ~]$ sudo yum install -y epel-release
読み込んだプラグイン:extras_suggestions, langpacks, priorities, update-motd
amzn2-core
| 3.7 kB 00:00:00
パッケージ epel-release は利用できません。
エラー: 何もみません

epel-release is available in Amazon Linux Extra topic "epel"

To use, run
# sudo amazon-linux-extras install epel
```

SNMPTTのインストール・設定

SNMPTTをEPELリポジトリを利用してインストールします。

```
sudo yum --enablerepo=epel install -y snmptt
```

snmp/syslog関連のツールをインストールします。

```
sudo yum --enablerepo=epel install -y net-snmp net-snmp-utils net-snmp-perl perl-Sys-Syslog
```

snmptt.iniを編集します。

```
sudo cp -p /etc/snmp/snmptt.ini /etc/snmp/snmptt.ini.backup
sudo vi /etc/snmp/snmptt.ini
```

(下記を編集)

[General]

mode = standalone

↓

mode = daemon

```
net_snmp_perl_enable = 0
```

```
↓
```

```
net_snmp_perl_enable = 1
```

```
net_snmp_perl_best_guess = 0
```

```
↓
```

```
net_snmp_perl_best_guess = 2
```

```
#date_time_format =
```

```
↓
```

```
date_time_format = %Y/%m/%d %H:%M:%S
```

```
[DaemonMode]
```

```
sleep = 5
```

```
↓
```

```
sleep = 1
```

```
[Debugging]
```

```
DEBUGGING = 0
```

```
↓
```

```
DEBUGGING = 1
```

```
DEBUGGING_FILE =
```

```
# DEBUGGING_FILE = /var/log/snmpd/snmpd.debug
```

```
↓
```

```
# DEBUGGING_FILE =
```

```
DEBUGGING_FILE = /var/log/snmpd/snmpd.debug
```

```
DEBUGGING_FILE_HANDLER =
```

```
# DEBUGGING_FILE_HANDLER = /var/log/snmpd/snmpdhandler.debug
```

```
↓
```

```
# DEBUGGING_FILE_HANDLER =
```

```
DEBUGGING_FILE_HANDLER = /var/log/snmpd/snmpdhandler.debug
```

```
[TrapFiles]
```

```
/etc/snmp/snmpd.conf
```

```
↓
```

```
/etc/snmp/snmpd_base.conf
```

snmptt_base.confファイルに検知したいSNMPTrapの内容を記述します。今回は、リンクダウン/リンクアップを検知できるようにします。

```
sudo vi /etc/snmp/snmptt_base.conf

(下記を記述)
EVENT general .1.3.6.1.6.3.1.1.5.3 "linkDown" Critical
FORMAT ZBXTRAP $aA $ar $*

EVENT general .1.3.6.1.6.3.1.1.5.4 "linkUp" Normal
FORMAT ZBXTRAP $aA $ar $*
```

上記で定義している変数の意味は下記の通りです。ネットワーク機器からのSNMPTrapの場合、“\$aA”と“\$ar”は同じIPアドレスが表示されます。

変数	意味
\$aA	agent-addrのIPアドレス
\$ar	IPアドレス
\$*	全てのSNMPTrap内メッセージ

snmptrapdの設定

snmptrapdを編集します。（起動オプションの設定）

```
sudo cp -p /etc/sysconfig/snmptrapd /etc/sysconfig/snmptrapd.backup
sudo vi /etc/sysconfig/snmptrapd

(下記を追記)
OPTIONS="-m +ALL -Lsd -On"
```

snmptrapd.confを編集します。（SNMPTTを実行するための設定）

```
sudo cp -p /etc/snmp/snmptrapd.conf /etc/snmp/snmptrapd.conf.backup
sudo vi /etc/snmp/snmptrapd.conf

(下記を追記)
```

```
authCommunity    log, execute, net public
authCommunity    log, execute, net cmlpublic
perl do "/usr/share/snmp/snmpthandler-embedded"
```

Zabbix Sever の設定

zabbix_server.confを編集します。（読み取るログファイルの変更、SNMPTrapperの有効化）

```
sudo cp -p /etc/zabbix/zabbix_server.conf /etc/zabbix/zabbix_server.conf.backup_1
sudo vi /etc/zabbix/zabbix_server.conf
```

（下記を編集）

```
SNMPTrapperFile=/var/log/snmptrap/snmptrap.log
```

↓

```
SNMPTrapperFile=/var/log/snmpth/snmpth.log
```

```
# StartSNMPTrapper=0
```

↓

```
StartSNMPTrapper=1
```

サービスの起動

SNMPPTTとsnmptrapdを起動し、自動起動設定を追加します。

```
sudo systemctl start snmpth snmptrapd
sudo systemctl enable snmpth snmptrapd
```

ZabbixServerを再起動します。

```
sudo systemctl restart zabbix-server
```

SNMPPTTの動作確認

サーバーから手動でSNMPTrapを送出し動作確認を行います。

```
sudo snmptrap -v 2c -c public 127.0.0.1 '' .1.3.6.1.6.3.1.1.5.3
```

snmptt.logに表示されることを確認します。

```
sudo cat /var/log/snmptt/snmptt.log
```

```
[ec2-user@ip-10-0-0-100 ~]$ sudo cat /var/log/snmptt/snmptt.log
2021/09/14 02:07:41 .1.3.6.1.6.3.1.1.5.3 Critical "linkDown" 127.0.0.1 -
ZBXTRAP 127.0.0.1 127.0.0.1
```

Zabbixの設定

Router1のGigabitEthernet0/3のリンクダウンを検知するための設定を行ないます。（汎用的な設定ではありませんが、動作確認として検知対象を絞って設定します。）

アイテムの登録

Router1の「アイテム」をクリックします。

The screenshot shows the Zabbix web interface. The top navigation bar includes 'ZABBIX', '監視データ', 'インベントリ', 'レポート', '設定', and '管理'. Below this, there's a sub-navigation bar with 'ホストグループ', 'テンプレート', 'ホスト', 'メンテナンス', 'アクション', 'イベント相関関係', 'ディスカバリ', and 'サービス'. The main content area is titled 'ホスト' and shows a list of hosts. The 'Router1' host is selected, and the 'Items' tab is active. A table lists items for Router1, Server, and VPN-Router. The 'Router1' item is highlighted with a red box.

名前	アプリケーション	アイテム	トリガー	グラフ	ディスカバリ	Web	インターフェース	テンプレート	ステータス	エージェントの状態	エージェント番号	情報	タグ
Router1	アプリケーション 9	アイテム 14	トリガー 6	グラフ	ディスカバリ 8	Web	192.168.2.1: 10050	Template Net Cisco IOS SNMPv2 (Template Module Cisco CISCO-ENVMON-MIB SNMPv2, Template Module Cisco CISCO-MEMORY-POOL-MIB SNMPv2, Template Module Cisco CISCO-PROCESS-MIB SNMPv2, Template Module Cisco Inventory SNMPv2, Template Module EtherLike-MIB SNMPv2, Template Module Generic SNMPv2, Template Module Interfaces SNMPv2)	有効	2018	SNMP	OK	なし
Server	アプリケーション 1	アイテム 3	トリガー 3	グラフ	ディスカバリ	Web	172.16.1.1: 10050	Template Module ICMP Ping	有効				
VPN-Router	アプリケーション 9	アイテム 14	トリガー 6	グラフ	ディスカバリ 8	Web	192.168.1.100: 10050	Template Net Cisco IOS SNMPv2 (Template Module Cisco CISCO-ENVMON-MIB SNMPv2, Template Module Cisco CISCO-MEMORY-POOL-MIB SNMPv2, Template Module Cisco CISCO-PROCESS-MIB SNMPv2, Template Module Cisco Inventory SNMPv2, Template Module EtherLike-MIB SNMPv2, Template Module Generic SNMPv2, Template Module Interfaces SNMPv2)	有効	2018	SNMP	OK	なし

3件のうち3件を表示しています

「アイテムの作成」をクリックします。

ZABBIX

監視データインベントリレポート設定管理

ホストグループテンプレートホストメンテナンスアクションイベント相関関係ディスクバリサービス

アイテム

すべてのホスト / Router1 有効 ZBX SNMP ZBX IPMI アプリケーション 9 アイテム 14 トリガー 6 グラフ ディスカバリブルール 8 Webシナリオ

アイテムの作成

フィルター

ホストグループ

検索文字列を入力

選択

タイプ

すべて

データ型

すべて

状態

すべて

ホスト

Router1

選択

監視期間

履歴

ステータス

すべて

アプリケーション

選択

トリガー

すべて

名前

テンプレート

すべて

キー

ディスクバリ

すべて

適用

リセット

サブフィルター フィルターしたデータにのみ影響があります

アプリケーション

General 6 Inventory 3 Status 6

タイプ

SNMPv2エージェント 9 SNMPトラップ 1 Zabbixインターナル 1 シンプルチェック 3

データ型

ログ 1 数値 (整数) 3 数値 (浮動小数) 2 文字列 6

トリガーあり

トリガーあり 6 トリガーなし 6

履歴

7d 3 14d 11

トレンド

9d 3 1y 3

監視期間

30s 1 1m 4 1h 6

☐ ウィザード

名前 *

トリガー

キー

監視期間

履歴

トレンド

タイプ

アプリケーション

ステータス

情報

☐ ***

Template Module Generic SNMPv2: Device contact details

system.contact

1h

2w

SNMPv2エージェント

General

有効

☐ ***

Template Module Generic SNMPv2: Device description

system.descr

1h

2w

SNMPv2エージェント

General

有効

下記の通り設定し、更新をクリックします。

項目	内容
名前	SNMPTrap(Gi0/3-LinkDown) ※任意の名前
タイプ	SNMPトラップ
キー	snmptrap[linkDown.*GigabitEthernet0/3]
データ型	ログ

ZABBIX 監視データ インベントリ レポート 設定 管理

ホストグループ テンプレート ホスト メンテナンス アクション イベント相関関係 ディスカバリ サービス

アイテム

すべてのホスト / Router1 有効 ZBX SNMP JMX IPMI アプリケーション 9 アイテム 15 トリガー 6 グラフ ディスカバリールール 8 Webシナリオ

アイテム 保存前処理

* 名前

SNMPTrap(Gi0/3-LinkDown)

* タイプ

SNMPトラップ

* キー

snmptrap[linkDown.*GigabitEthernet0/3]

選択

* ホストインターフェース

192.168.2.1 : 161

データ型

ログ

* ヒストリの保存期間

Do not keep history Storage period 90d

ログの時間の形式

アプリケーションの作成

アプリケーション

-なし-

CPU

Fans

General

Inventory

Memory

Network interfaces

Power supply

Status

Temperature

説明

有効 ☒

更新 複製 監視データ取得 ヒストリとトレンドを削除 削除 キャンセル

トリガーの登録

「トリガー」→「トリガーの作成」をクリックします。

ZABBIX

監視データ インベントリ レポート **設定** 管理

ホストグループ テンプレート **ホスト** メンテナンス アクション イベント相関関係 ディスカバリ サービス

トリガー

すべてのホスト / Router1 有効 ZBX **SNMP** ZBX (PMU) アプリケーション 9 アイテム 19 **トリガー 1** グラフ ディスカバリールール 8 Webシナリオ

トリガーの作成

フィルター

ホストグループ 検索文字列を入力 選択

タグ And/Or Or 含む 等しい 値 追加

ホスト Router1 検索文字列を入力 選択

名前

深刻度 未分類 情報 警告 継承したもの すべて はい いいえ 軽度の障害 軽度の障害 重度の障害 致命的な障害 Discovered すべて はい いいえ 依存関係がある すべて はい いいえ

状態 すべて ノーマル 不明

ステータス すべて 有効 無効

値 すべて 正常 障害

適用 リセット

深刻度	値	名前	条件式	ステータス	情報	タグ
情報	正常	Template Module Cisco Inventory SNMPv2: Device has been replaced (new serial number received)	{Router1:system.hw.serialnumber.diff()}=1 and {Router1:system.hw.serialnumber.strlen()}>0	有効		
警告	正常	Template Module ICMP Ping: High ICMP ping loss 依存先: Router1: Unavailable by ICMP ping	{Router1:icmppingloss.min(5m)}>{\$ICMP_LOSS_WARN} and {Router1:icmppingloss.min(5m)}<100	有効		
警告	正常	Template Module ICMP Ping: High ICMP ping response time 依存先: Router1: High ICMP ping loss Router1: Unavailable by ICMP ping	{Router1:icmppingsec.avg(5m)}>{\$ICMP_RESPONSE_TIME_WARN}	有効		
警告	正常	Template Module Generic SNMPv2: No SNMP data collection 依存先: Router1: Unavailable by ICMP ping	{Router1:zabbix[host.snmp.available].max(\$SNMP_TIMEOUT)}=0	有効		
重度の障害	正常	Template Module ICMP Ping: Unavailable by ICMP ping 依存先: Router1: No SNMP data collection	{Router1:icmpping.max(*3)}=0	有効		
警告	正常	Template Module Generic SNMPv2: (HOST.NAME) has been restarted 依存先: Router1: No SNMP data collection	{Router1:system.uptime(sysUpTime).last()}<10m	有効		

下記の通り設定し、更新をクリックします。

項目	内容
名前	LinkDown(Gi0/3)-SNMPTrap ※任意の名前
深刻度	重度の障害 ※任意の深刻度
条件式	{Router1:snmptrap[linkDown.*GigabitEthernet0/3].regexp(linkDown)}=1
手動クローズを許可	チェックを入れる

ZABBIX 監視データ インベントリ レポート 設定 管理

ホストグループ テンプレート ホスト メンテナンス アクション イベント相関関係 ディスカバリ サービス

トリガー

すべてのホスト / Router1 有効 ZBX SNMP JMX IPMI アプリケーション 9 アイテム 15 トリガー 7 グラフ ディスカバリールール 8 Webシナリオ

トリガー タグ 依存関係

* 名前 LinkDown(Gi0/3)-SNMPTrap

深刻度 未分類 情報 警告 軽度の障害 重度の障害 致命的な障害

* 条件式 {Router1:snmptrap[linkDown.*GigabitEthernet0/3].regexp(linkDown)}=1 追加

条件式ビルダー

正常イベントの生成 条件式 復旧条件式 なし

障害イベント生成モード 単一 複数

正常時のイベントクローズ すべての障害 タグの値が一致したすべての障害

手動クローズを許可 ☒

URL

説明

有効 ☒

更新 複製 削除 キャンセル

ネットワーク機器の設定

Router1でTrapを送出するための設定を行います。

```
[Router1]
snmp-server community cmlpublic RO
snmp-server host 10.0.0.100 version 2c cmlpublic
snmp-server enable traps
```

監視検知の確認

サーバー側でsnmptt.logを確認できるようにしておきます。

```
sudo tail -f /var/log/snmptt/snmptt.log
```

Router1のGi0/3をshutdownします。

```
[Router1]
int Gi0/3
shut
```

snmptt.logに表示されたことを確認します。

```
02:13:49 .1.3.6.1.6.3.1.1.5.3 Critical "linkDown" 192.168.2.1 - ZBXTRAP 192.168.2.1
192.168.2.1 4 GigabitEthernet0/3 ethernetCsmacd administratively down
```

「監視データ」→「障害」の画面で、Router1の障害として検知されていることを確認します。



時刻	深刻度	障害種類	ステータス	情報	ホスト	障害	継続期間	確認済	アクション	タグ
11:48:44	重症の障害	障害			Router1	LinkDown(Gi0/3)-SNMPTrap	5s	いいえ		

以上で、AWS上での監視サーバー(Zabbix)構築【8.監視登録(SNMP Trap)】の説明は完了です！