

自宅サーバの監視体制の構築

Building a home server monitoring system

AD22074 山村 律貴

AD22074 Ritsuki Yamamura

芝浦工業大学 無線研究部

Shibaura Institute of Technology, Ham radio club

1. 動機

我が家の自宅サーバで公開中の Stratum1 の Network Time Protocol サーバ（以下 NTP サーバと表記）である ntp.pcdestroyer330.com の接続数が増加しており、サーバの状態の監視体制を強化する必要性が出てきたため。

2. 目的

自宅サーバで公開しているサービスの利用者側に不利益がでる状況を把握してその状況を未然に防ぐ、あるいは即時に検知するための環境を構築する。

3. 環境構築

今回はサーバの状態などを取得し、そのデータを処理して Web 画面に表示する、異常が発生したらメールで通知するという機能を持ったサーバを構築していく。しかし、ゼロから構築するのは難易度が高いため、オープンソースソフトウェアの統合システム監視ソフトウェアである Zabbix を用いることにする。

3.1 インストール

サーバをインストールするハードウェアの性能、OS、Zabbix のバージョンは以下のとおりである。

- CPU : 4 コア 4 スレッド
- メモリ : 8GB
- SSD : 16GB
- OS : Debian 12.2
- Zabbix : 6.0 LTS

Zabbix をインストールする際は公式サイトに記述している方法を利用し、インストールする際に必要なデータベースには PostgreSQL を、ウェブサーバには Apache を選択する。

3.2 設定

サーバに監視対象、監視項目について設定する。監視対象は NTP サーバ、監視機能付きの電源タップ (Power Distribution Unit, 以下 PDU と表記)、無停電電源装置 (Uninterruptible Power Supply, 以下 UPS と表記) の3つであり、監視方法は簡易ネットワーク管理プロトコル (Simple Network Management Protocol, 以下 SNMP と表記) を用いる。

3.3 管理画面の作成

監視対象の状態を一目で把握できるような監視画面 (ダッシュボード) を作成する。

4. 結果

NTP サーバの状態、自宅サーバ全体の障害の有無が一目で把握できる監視画面を作成することができた。

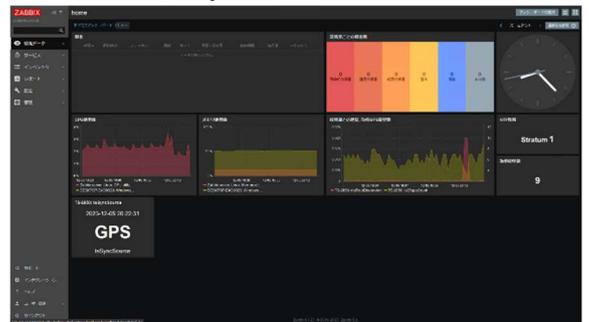


図 1 作成した監視画面

5. 展望

監視画面を作成することは出来たが、監視対象の設定の部分であまりよろしくない設定方法をしてしまった部分があるので、今後は底を修正していきたい。

6. 参考文献

• Zabbix 「Download and install Zabbix」
https://www.zabbix.com/download?zabbix=6.0&os_distribution=debian&os_version=12&components=server_frontend_agent&db=pgsql&ws=apache (参照日 2023 年 12 月 8 日)

用語解説

- 自宅サーバ

自宅に設置されたサーバのことであり、インターネットに公開し Web サーバなどのサービスを運用する、家庭内 LAN で自宅内の機器を管理するのに使われる。個人が趣味で運用することが多い。

- Network Time Protocol (NTP)

ネットワーク上のコンピュータシステム間で時刻同期させるための通信プロトコルのこと。NTP サーバの負荷軽減のため、NTP サーバには時刻源とサーバの距離による階層構造を持っている。時刻源である原子時計や GPS を Stratum0 として、時刻源に直結されたサーバを Stratum1、Stratum1 を参照しているサーバを Stratum2、というように時刻源から離れるほど Stratum の値が大きくなる。Stratum の最大値は 16 である。

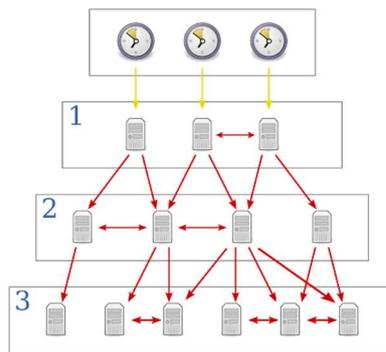


図 1 NTP の階層構造

- Global Positioning System(GPS)

人工衛星 (GPS 衛星) から発せられた電波を受信し、現在位置を特定するもの

- シリアルコンソール

映像出力やキーボード、マウスを接続するポートがない機器(図 2 のようなもの。この機器では RS-232 と書かれているところがシリアルポートである。)



図 2 映像出力端子のない機器の例

- Zabbix
Zabbix 社が公開しているオープンソースの統合監視ソフトウェアのこと。監視対象からのデータ収集やしきい値の判定、アラート通知のアクションなどを行う。
- PostgreSQL
オープンソースで開発されているデータベース管理システムの一つ。
- Apache
世界中で使用されている Web サーバーソフトウェア (HTTP サーバー) のこと。
- Power Distribution Unit (PDU)
ラックに収納された IT 機器に電源を供給する機材のこと。我が家で使用している機材は電圧、電流の測定、外部からの電源の入り切りをすることができる。
- Uninterruptible Power Supply (UPS)
停電が起きてしまったときに電気を一定時間供給し続けるための装置。大きく分けて 3 つの種類がある。
 - ・ 常時インバータ給電方式 UPS
商用電力の状態に関わらずインバータによって、常に調整された安定した出力を接続機器へ出力する方式。常にインバータ回路から供給されているため、停電発生時にも運転切り換え時の断時間を発生することがない。
 - ・ ラインインタラクティブ UPS
商用電源が正常な通常時は商用電力をそのまま接続機器へ供給すると共に双方向インバータを充電器として制御してバッテリーへ充電を行い、停電発生時には双方向インバータによりバッテリー電力を交流に変換して負荷機器へ供給する方式。
 - ・ 常時商用給電方式 UPS
商用電源が正常な通常時はその商用電源をそのまま接続された機器へ給電する UPS 方式。停電が発生すると、蓄電されていたバッテリー電力がインバータにより直

流から交流に変換され、出力が商用電源からインバータ側に切り換えられて接続機器へ電力を供給する。

- Simple Network Management Protocol (SNMP)

IP ネットワーク上のルータやスイッチ、サーバ、端末など様々な機器をネットワーク経由で遠隔から監視・制御するためのプロトコルの 1 つ。管理者が操作や管理のために用いるソフトウェアを「SNMP マネージャ」、監視や制御の対象となる個々の機器に導入されるソフトウェアを「SNMP エージェント」という

- Object ID (OID)

あらゆるオブジェクト、概念、または「モノ」に世界的に重複しない一意な識別子(番号)を割り当てるために作られた識別子メカニズムのこと。国際電気通信連合 (ITU) および ISO/IEC により、共同で標準化されている。

例：.1.3.6.1.4.1.955.1.5.3.3.3.4.1.0

- Management Information Base (MIB)

ネットワーク上で遠隔から機器を監視・管理する際に用いられる、監視対象の機器が自らの設定や状態についてまとめたデータ集合のこと。現在の状態や設定を格納する個々の値を「オブジェクト」(object) と呼び、それぞれのオブジェクトについて「OID」(Object ID) と呼ばれる識別符号が割り当てられる。

- パケットキャプチャー

ネットワークを流れる転送されるデータとともに通信の送信元や宛先、プロトコルなどの情報が含まれているパケットを収集すること。ネットワーク障害などのトラブルの調査やセキュリティの問題をテストなどに利用される。また、不正な通信を検出することもできる。

2023研究発表 (1).txt

- ・ OSのインストール

```
# シリアルコンソール経由でDebian12. 2. 0をインストール
# 115200bps, 8bit, 1bitストップ, パリティなし, フローコントロールなし
```

```
install console=ttyS0,115200n8
```

- ・ 初期設定

```
# ログインユーザー : root
```

```
$ apt update
$ apt upgrade -y
```

```
# sudoがインストールされていなかったのでインストール
$ apt install -y sudo
# "ユーザー名"をsudoグループに追加
$ gpasswd <ユーザー名> sudo
# セキュリティー向上のためrootアカウントを無効化
$ passwd -l root
$ logout
```

- ・ Zabbixのインストール(<ユーザー名>でログイン)

```
# postgresqlをインストール
$ sudo apt install -y postgresql
# zabbixのパッケージをダウンロード
$ sudo wget
https://repo.zabbix.com/zabbix/6.0/debian/pool/main/z/zabbix-release/zabbix-release\_6.0-5+debian12\_all.deb
# ダウンロードしたzabbixのパッケージをインストール
```

1/8 ページ

2023研究発表 (1).txt

```
$ sudo dpkg -i zabbix-release_6.0-5+debian12_all.deb
$ sudo apt update
# 必要なパッケージをインストール
$ sudo apt install -y zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-sql-scripts zabbix-agent
# zabbixという名前のpostgresqlのユーザーを作成
$ sudo -u postgres createuser --pwprompt zabbix
# zabbixユーザ権限でzabbixという名前のDatabaseを作成
$ sudo -u postgres createdb -0 zabbix zabbix
# server.sql.gzの内容をzabbix Databaseに流し込む
$ zcat /usr/share/zabbix-sql-scripts/postgresql/server.sql.gz | sudo -u zabbix psql zabbix
# zabbixの設定ファイルにDatabaseのパスワードを記述
$ sudo vi /etc/zabbix/zabbix_server.conf
    - # DBPassword=
    + DBPassword="パスワード"

# ローカル環境の設定
$ sudo dpkg-reconfigure locales
# zabbix-server, zabbix-agent, apache2の再起動
$ sudo systemctl restart zabbix-server zabbix-agent apache2
# zabbix-server, zabbix-agent, apache2の自動起動の有効化
$ sudo systemctl enable zabbix-server zabbix-agent apache2

- 監視対象の設定
# NTPサーバーから取得できる標準MIBであるSNMPv2-MIBを使って取得できるOIDとその値を表示
$ snmpwalk -v 2c -c <NTPサーバーで設定したSNMPコミュニティ名> <NTPサーバーのIPアドレス>

SNMPv2-MIB::sysDescr.0 = STRING: TS-2850 Time Server
```

2/8 ページ

2023研究発表 (1).txt

```
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::enterprises.955.1.5
SNMPv2-MIB::sysUpTime.0 = Timeticks: (44782805) 5 days, 4:23:48.05
SNMPv2-MIB::sysContact.0 = STRING:
SNMPv2-MIB::sysName.0 = STRING: time-server
SNMPv2-MIB::sysLocation.0 = STRING:
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (3) 0:00:00.03
```

NTPサーバー固有のOIDとその値を表示

```
$ snmpwalk -v 2c -c <NTPサーバーで設定したSNMPコミュニティ名> <NTPサーバーのIPアドレス>
```

```
SNMPv2-SMI::enterprises.955.1.5
```

```
SNMPv2-SMI::enterprises.955.1.5.1.1.0 = INTEGER: 0
SNMPv2-SMI::enterprises.955.1.5.2.1.1.0 = STRING: "TS-2850 GPS Time Server"
SNMPv2-SMI::enterprises.955.1.5.2.1.2.0 = STRING: "GPS/TJJY"
SNMPv2-SMI::enterprises.955.1.5.2.1.3.1.0 = STRING: "3.0020"
SNMPv2-SMI::enterprises.955.1.5.2.1.3.2.0 = STRING: "SGTA1020"
SNMPv2-SMI::enterprises.955.1.5.2.1.4.0 = INTEGER: 9
SNMPv2-SMI::enterprises.955.1.5.2.1.5.1.0 = INTEGER: 1
SNMPv2-SMI::enterprises.955.1.5.2.1.5.2.0 = STRING: "GPS"
SNMPv2-SMI::enterprises.955.1.5.2.2.1.0 = STRING: "00"
SNMPv2-SMI::enterprises.955.1.5.2.2.2.0 = INTEGER: 1
SNMPv2-SMI::enterprises.955.1.5.2.2.3.0 = INTEGER: -19
SNMPv2-SMI::enterprises.955.1.5.2.2.4.0 = STRING: "127.127.20.0"
SNMPv2-SMI::enterprises.955.1.5.2.2.5.0 = STRING: "client"
SNMPv2-SMI::enterprises.955.1.5.2.2.6.0 = STRING: "0.00000"
SNMPv2-SMI::enterprises.955.1.5.2.2.7.0 = STRING: "0.00034"
SNMPv2-SMI::enterprises.955.1.5.3.1.1.0 = INTEGER: 112010
```

2023研究発表 (1).txt

```
SNMPv2-SMI::enterprises.955.1.5.3.1.2.0 = STRING: "20231208-142746"  
SNMPv2-SMI::enterprises.955.1.5.3.1.3.0 = STRING: "Synchronized to GPS"  
SNMPv2-SMI::enterprises.955.1.5.3.2.1.0 = INTEGER: 0  
SNMPv2-SMI::enterprises.955.1.5.3.2.2.0 = INTEGER: 0  
SNMPv2-SMI::enterprises.955.1.5.3.2.3.0 = INTEGER: 0  
SNMPv2-SMI::enterprises.955.1.5.3.2.4.0 = INTEGER: 0  
SNMPv2-SMI::enterprises.955.1.5.3.3.2.1.0 = STRING: "11:03"  
SNMPv2-SMI::enterprises.955.1.5.3.3.2.2.0 = INTEGER: 24  
SNMPv2-SMI::enterprises.955.1.5.3.3.2.3.0 = INTEGER: 0  
SNMPv2-SMI::enterprises.955.1.5.3.3.2.4.0 = INTEGER: 0  
SNMPv2-SMI::enterprises.955.1.5.3.3.3.1.0 = INTEGER: 1  
SNMPv2-SMI::enterprises.955.1.5.3.3.3.2.0 = INTEGER: 1  
SNMPv2-SMI::enterprises.955.1.5.3.3.3.3.0 = INTEGER: 5  
SNMPv2-SMI::enterprises.955.1.5.3.3.3.4.1.0 = STRING: "35.8925N"  
SNMPv2-SMI::enterprises.955.1.5.3.3.3.4.2.0 = STRING: "139.4930E"  
SNMPv2-SMI::enterprises.955.1.5.3.3.3.4.3.0 = STRING: "+49"  
SNMPv2-SMI::enterprises.955.1.5.3.3.3.4.3.0 = No more variables left in this MIB View (It is past the end of the MIB tree)
```

ここからはWEB画面から設定

- ・ NTPサーバーを監視対象のホストとして追加する

設定 > ホスト > ホストの作成

ホスト名 : TS-2850
表示名 : TS-2850
テンプレート :

//NTPサーバーのサーバー名

//NTPサーバーのサーバー名

//該当するテンプレートが存在しないため、テンプレートは選択

しない

2023研究発表 (1).txt

グループ : Templates/Server hardware //このホストをTemplates/Server hardwareに追加
インターフェース
タイプ : SNMP //SNMPによる監視をするため、SNMPを
選択
IPアドレス : <NTPサーバーのIPアドレス> //監視対象のサーバーのIPアドレスを
記述
DNS名 : //DNSの設定はしていないため、DNS名
は記述しない
接続方法 : IP //ホストをIPアドレスで指定する
ポート : 161 //SNMPのデフォルトの宛先ポートであ
る161番ポートを指定
SNMPバージョン : SNMPv2 //使用するSNMPのバージョンを選択
SNMPコミュニティ : <NTPサーバーで設定したSNMPコミュニティ名> //SNMPで管理するネットワークシステ
ムの範囲を指定

・SNMPによって取得・監視するデータを記述
snmpwalkで取得できた値を使用する

設定 > ホスト > TS-2850 アイテム > アイテムの作成

名前 : ntpLeapIndicator //アイテムの名称を
指定
タイプ : SNMPエージェント //監視方法を指定
キー : ntp.leapindicator //トリガーが参照す
る固有の文字列を設定
データ型 : 数値(整数) //取得するデータの
型を指定
ホストインターフェース : <NTPサーバーのIPアドレス>:161 //監視対象

2023研究発表 (1).txt

SNMP OID	: SNMPv2-SMI::enterprises.955.1.5.2.2.1.0	//監視する項目の
OIDを指定		
単位	: 無し	//データの単位を指
定		
監視間隔	: 1m	//データの監視感覚
を指定		
履歴の保存期間	: 90d	//取得したデータの
保存期間		
トレンドの保存期間	: 365d	//1時間に1回平均化
されたデータの保存期間		
値のマッピング	: 無し	//アイテム値をわか
りやすく表示するための設定を記述		
ホストインベントリフィールドの自動設定	: なし	//インベントリフィ
ールドを設定		
説明	: 現在の Leap Indicator	//アイテムの説明を
記述		

これと同様の設定をsnmpwalkで取得できたOID全てについて設定する

- ・ 監視対象の機器に障害が発生したか否かを判定するためのトリガーを設定する

設定 > ホスト > TS-2850 トリガー > トリガーの作成

名前	: ntp server: gps antenna error	//トリガーの名称を指定
イベント名	: ntp server: gps antenna error	//イベントの名称を指定
運用データ	: 無し	//運用データを指定
深刻度	: 致命的な障害	//障害の深刻度を指定
条件式	: in(last(/TS-2850/ts25.gpsantenna), 1)=0	//障害発生を指定。この場合、

2023研究発表 (1).txt

復旧条件式 : `in(last(/TS-2850/ts25. gpsantenna), 1)=1` //障害の復旧条件を指定。この場合、
直近1分間のts25. gpsantenna(GPSアンテナの状態。1で正常、1以外で異常)の値が1(正常)と一致している

正常イベントの生成 : 条件式 //正常イベントの生成方法を指定
障害イベント生成モード : 単一 //障害イベントの生成モードを指定
正常時のイベントクローズ : すべての障害 //正常時のイベントクローズの方法を指定

指定

説明 : GPSアンテナの故障 //トリガーの説明を記述

名前 : ntp server: ntp error
イベント名 : ntp server: ntp error
運用データ : 無し
深刻度 : 致命的な障害
条件式 : `in(last(/TS-2850/ntp. stratum), 1)=0` //直近1分間のntp. stratum(現在の
Stratumの値)の値が1(正常)と一致していない

復旧条件式 : `in(last(/TS-2850/ntp. stratum), 1)=1` //直近1分間のntp. stratum(現在の
Stratumの値)の値が1(正常)と一致している

正常イベントの生成 : 条件式
障害イベント生成モード : 単一
正常時のイベントクローズ : すべての障害
説明 : Stratum異常

名前 : ntp server: timesource(GPS) error
イベント名 : ntp server: timesource(GPS) error
運用データ : 無し
深刻度 : 致命的な障害
条件式 : `last(/TS-2850/ts. syncsource) <> {$SYNCSORCE} and`

7/8 ページ

`first(/TS-2850/ts. syncsource, 5m) <> {$SYNCSORCE}` //5分間以上連続でts. syncsource(現在同期している時刻源)の文字列が
{ \$SYNCSORCE } (GPS) と一致していない

2023研究発表 (1).txt

```
first(/TS-2850/ts. syncsource, 1m) <> {$SYNCSORCE} and first(/TS-2850/ts. syncsource, 2m) <> {$SYNCSORCE} and  
first(/TS-2850/ts. syncsource, 3m) <> {$SYNCSORCE} and first(/TS-2850/ts. syncsource, 4m) <> {$SYNCSORCE} and  
first(/TS-2850/ts. syncsource, 5m) <> {$SYNCSORCE} //5分間以上連続でt
```

復旧条件式 : last(/TS-2850/ts. syncsource)={\$SYNCSORCE} //直近1分間の

ts. syncsource(現在同期している時刻源)の文字列が{\$SYNCSORCE}(GPS)と一致している

正常イベントの生成 : 条件式

障害イベント生成モード : 単一

正常時のイベントクローズ : すべての障害

説明 : タイムソース異常