IT インフラポートフォリオ

氏名：菅原　千尋

作成日：2025.12.06

1. プロジェクト概要

自宅 PC を対象に Zabbix を用いた監視環境を構築し、CPU・メモリ・ネットワークなど基本的なリソース監視を実装しました。

図１　構築した Web 上の Zabbix ダッシュボード

インフラ運用に必須となる監視設計の理解と、障害対応フローの整理を目的としたプロジェクトです。

2. 目的

インフラ運用の基礎となる「異常の早期検知」と「障害対応プロセス」を理解し、監視設計・監視項目の選定・アラート条件の設定といった実務に近い流れを再現することを目的としました。

現場のトラブルを見据えて対応に備えることを目標としました。

3. システム構成図

【構成概要】

Zabbix Server（Ubuntu VM）と監視対象 PC（Windows 11）を家庭内 LAN で接続し、

Zabbix Agent を用いてリソース情報（CPU／メモリ／ネットワーク）を取得する構成としました。

Zabbix Server には Web UI を導入し、ブラウザから状態確認・閾値管理・アラート履歴を確認できるようにしています。

【構成要素】

・Zabbix Server（Ubuntu VM）

・Zabbix Agent（監視対象 PC / Windows 11）

・家庭内 LAN（Wi-Fi ルーター）

・管理端末（Web UI 閲覧用 PC）

3. 使用技術

・OS：Windows 11（監視対象）、Ubuntu VM（Zabbix Server）

・監視ツール：Zabbix

・ネットワーク：ローカルネットワーク（家庭内 LAN）
・補助ツール：Zabbix Agent、Zabbix Web UI
・その他：VM(Oracle VirtualBox)

4. 監視項目

【監視項目の詳細】
本プロジェクトでは、インフラ運用で最も基本となる以下の項目を監視対象としました。

・CPU 使用率

- Zabbix Agent を利用し、一定時間の平均使用率を取得。
- 閾値：80%以上でアラート通知。

・メモリ使用率

- OS の空きメモリ量を取得。
- 閾値：75%以上でアラート。

・ディスク空き容量

- 主要ドライブの空き容量を定期チェック。
- 閾値：15%以下で警告。

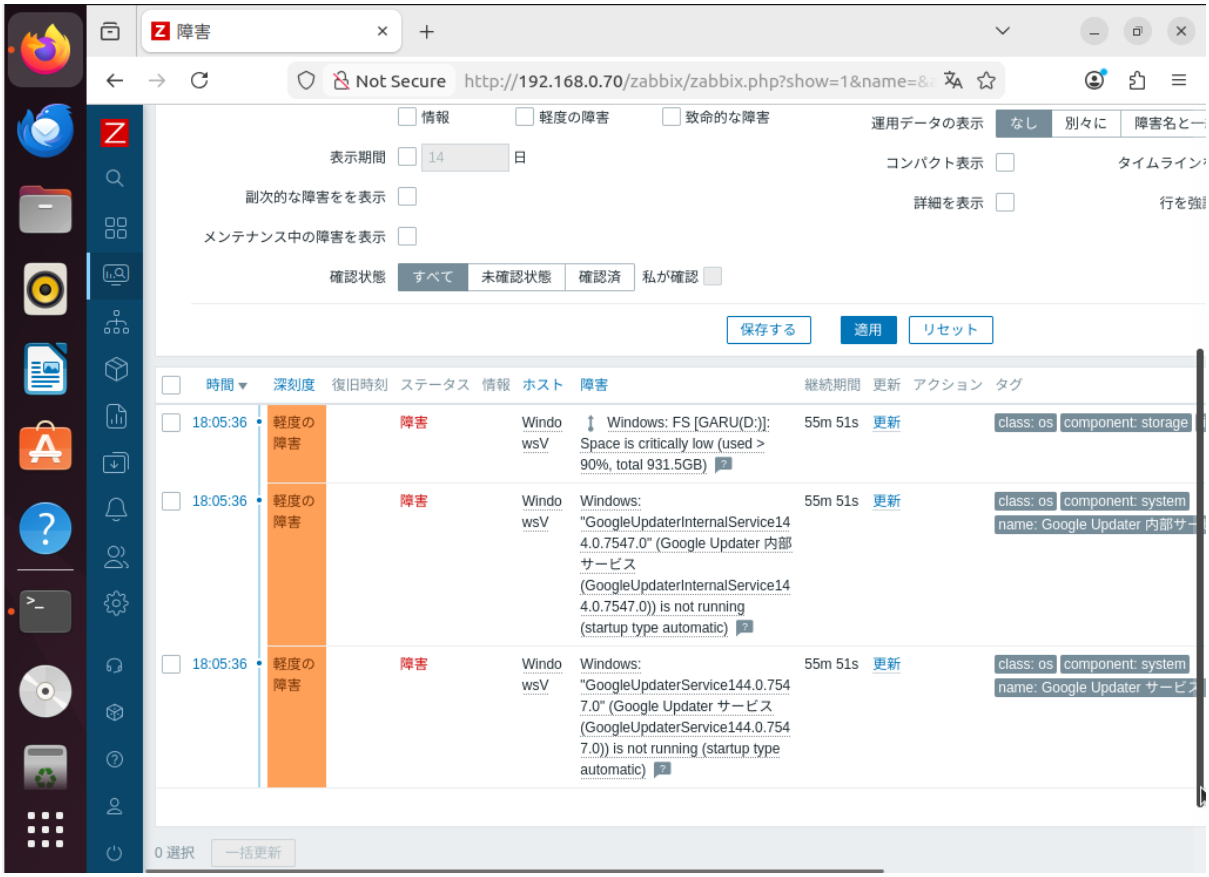

図２　空き容量の閾値を超えたことについて障害表示が発生

・ネットワーク疎通（Ping 監視）

- Zabbix Server から監視対象 PC へ ICMP 疎通を確認。

- 一定回数失敗した場合にアラート。

・サービス稼働確認（任意）

- 必要に応じて Windows サービス監視を追加可能。

6.　実施内容

【1. Zabbix Server の構築】

・Ubuntu VM 環境上に Zabbix Server・Zabbix Web UI をインストール。

・DB（MariaDB）をセットアップし、Zabbix の初期設定を実施。

・ホスト追加やテンプレート割当のための基本設定を整備。

【2. 監視対象 PC への Zabbix Agent 導入】

・Windows 10 に Zabbix Agent をインストール。

・agent.conf にてサーバの IP アドレスやホスト名を設定。

・サービス化して常時監視できるように設定。

【3. 監視項目の設定と閾値の調整】

・CPU・メモリ・ディスク・Ping などの基本テンプレートを適用。

・実際の利用状況を見ながら閾値を調整。

・リアルタイムでのリソース推移をグラフ表示で確認。

【4. アラート通知のテスト】

・CPU 負荷を意図的に発生させ、アラートが正しく発火するか確認。

・アラート履歴のログ確認を実施。

【5. 結果と改善ポイントの整理】

・監視項目が正常に取得できたことを確認。

・監視が弱い部分を洗い出し、次回改善案として認知・把握。

→ドライブの容量が認識よりはるかに逼迫しており、閾値の空き容量 15%以下どころではなかった。

一旦整理するなどして、閾値を改めて見直して次回に活かしたい。

7. アラートと対応フロー

本プロジェクトでは、実際のインフラ保守業務を想定し、アラート発生時の対応フローを以下のように整理しました。

---

1. アラート発生（例：CPU 使用率 80%超）

・Zabbix が閾値超過を検知し、アラートを発報。

・ダッシュボード上にステータスが表示される。

2. 初動確認（一次切り分け）

・過去のメトリクス推移を確認（急上昇か慢性的か）

・該当プロセスの特定（タスクマネージャ等）

・メモリ・ディスク使用状況も合わせて確認

3. 原因調査（詳細切り分け）

・高負荷プロセスの停止／再起動

・不要アプリの終了

・ネットワークの混雑状況（Ping 遅延）を確認

4. 暫定対応・恒久対応の整理

・暫定対応：該当プロセスの強制停止、一時的リソース解放など

・恒久対応：

- 不要常駐アプリの削除
- タスクスケジューラの整理
- メモリ容量の見直し
- 閾値設定の調整

5. 再発防止と設定の見直し

・アラートの感度（閾値）が適切か再検証

・運用ログを整理し、次回改善に反映

8. まとめ

本プロジェクトでは、Zabbix を用いて自宅 PC の基本監視環境を構築し、インフラ運用で重要となる監視設計・アラート設定・障害対応の流れを一通り再現しました。

特にアラート発生後の切り分けや再発防止のプロセスを整理したことで、業務で求められる「現場対応力」や「状況判断力」を強化できたと考えています。

今後は、監視項目の拡張（ディスク IO、サービス監視、SNMP 機器の追加）や 通知の自動化（メール通知など）にも取り組み、より実運用に近い監視環境の構築を目指します。