【事例発表】
複数監視マネージャのZabbix集約について

2022年11月17日

NTTコミュニケーションズ株式会社

今井　聡

自己紹介

■名前

今井　聡（いまい　さとる）

■所属

NTTコミュニケーションズ株式会社　ソリューションサービス部

■経歴

2013年　NTTコムテクノロジーに入社
　　社内システム維持開発担当に所属
2015年　会社名・組織変更でNTTコムソリューションズに移動
2018年　保守運用部門に異動
2021年　NTTコミュニケーションズに吸収合併

■得意分野

・サーバやNWの保守運用 (メイン業務)
・ServiceNow,Selenium,Ansibleを利用した自動実装
・ネットワークのリバースエンジニアリング

自己紹介

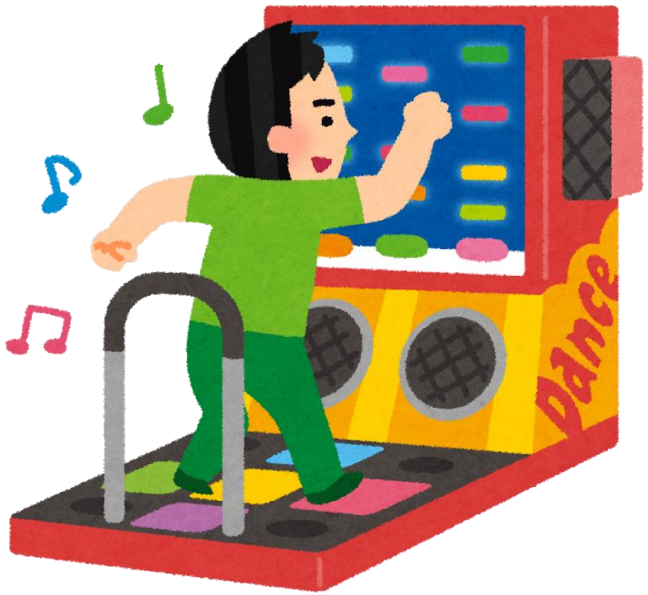
■出身

長野県諏訪郡下諏訪

「東方風神録」や「君の名は」の聖地

■趣味

・飲み会（友達を呼んで宅飲み、赤羽などで飲み歩き）
・バーチャルYoutuber視聴（ホロライブの宝鐘マリン＆ラプラスダークネス押し）
・ゲーム（Destiny2、スプラトゥーン3、Apexなど）
・音楽ゲーム（ダンスダンスレボリューション）



1.発表概要

1.発表概要

■実施したこと

物理サーバに仮想監視環境(Hyper-V)を構築し、そこで冗長構成のZabbixを構築。
弊社保守センタが保守運用している23社の監視マネージャ（WATT・Nagios
・Cacti）の監視を統合。
（監視情報：38,800ホスト、124,800アイテム、350vps）

■今回の発表内容

①Zabbix構築で発生した問題、工夫したこと
②他の監視マネージャからZabbixへ移行する上で大変だったこと、工夫したこと
③運用フェーズで発生した問題、工夫したこと

2.はじめてのZabbix構築

2.はじめてのZabbix構築

2-1.なぜZabbixへ複数案件の監視を統合したか

2-2.Zabbixへ統合イメージ

2-3.どんなZabbixを作ったのか

2-4.Zabbix物理環境

2-5.ストレージの速度がでない問題

2-6.ソースIP問題

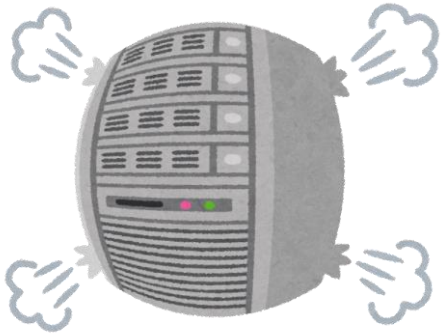
2-7.RAID構成の工夫

2-8.Windows Server 2019 Coreでの構築

2-1.なぜZabbixへ複数案件の監視を統合したか

もう処理できません

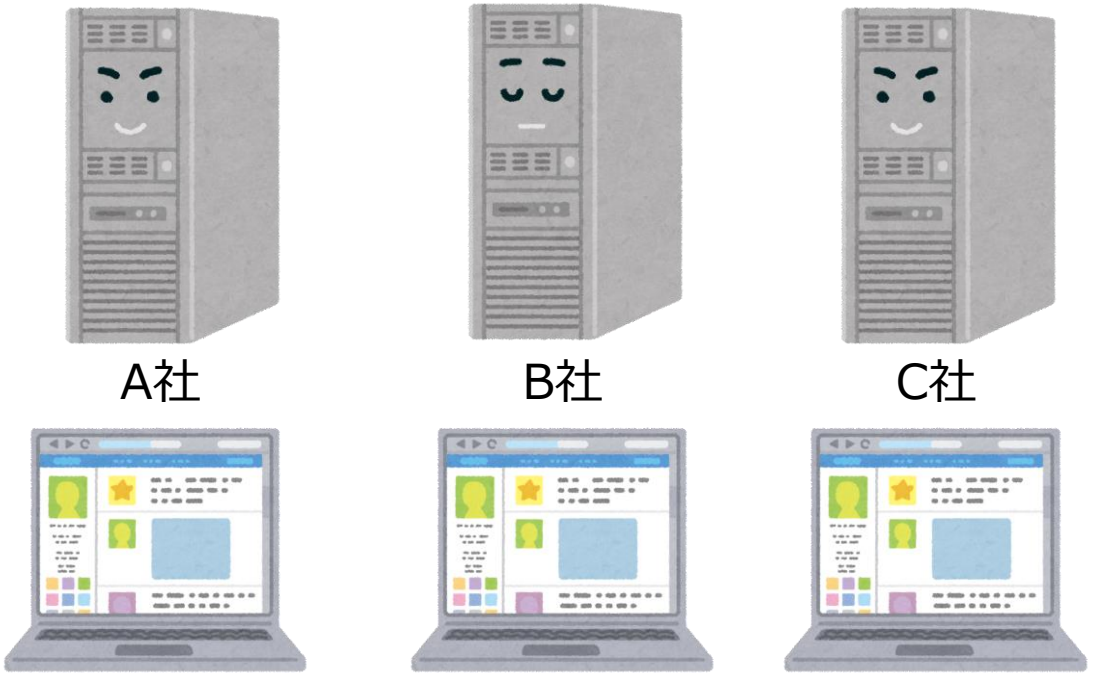
監視マネージャがつながらないので直してください



監視マネージャ
（Nagios,WATT,Cacti）

データセンターに行って直さなきゃ

2-1.なぜZabbixへ複数案件の監視を統合したか



案件ごとに見る画面が違うから確認も大変だし、
サーバや端末は場所や維持費かかるなぁ

2-1.なぜZabbixへ複数案件の監視を統合したか

コスト削減したいから自動化よろしく

監視マネージャの種類が多いし、
設定がばらばらだから
自動化が大変

なんでこんな種類の
監視マネージャ使ってるんだよ…

2-1.なぜZabbixへ複数案件の監視を統合したか

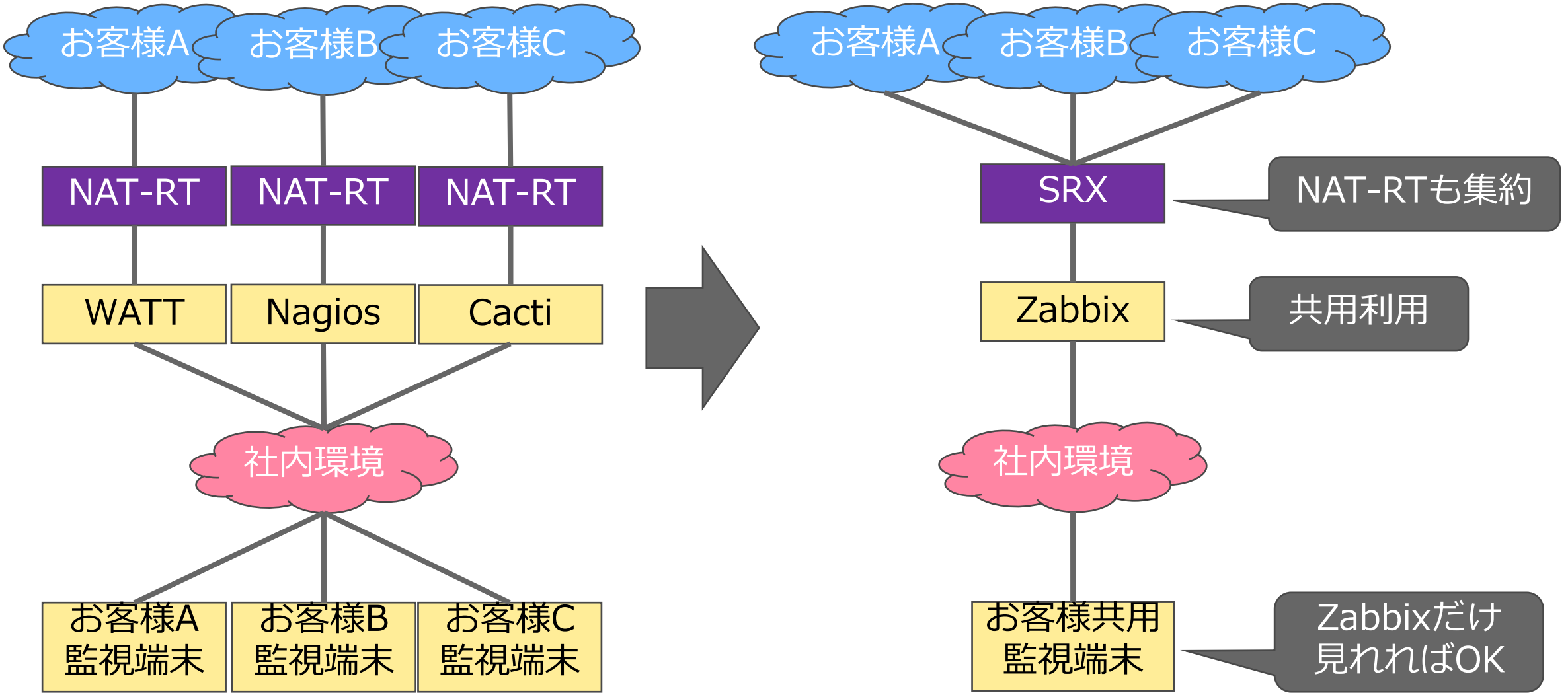
自分が楽をするためにも、複数案件（10案件程度）を１つのZabbixに統合するか。

予算いただければ監視マネージャを統合します

予算つけるから、全30案件を18ヶ月以内に移行してほしい

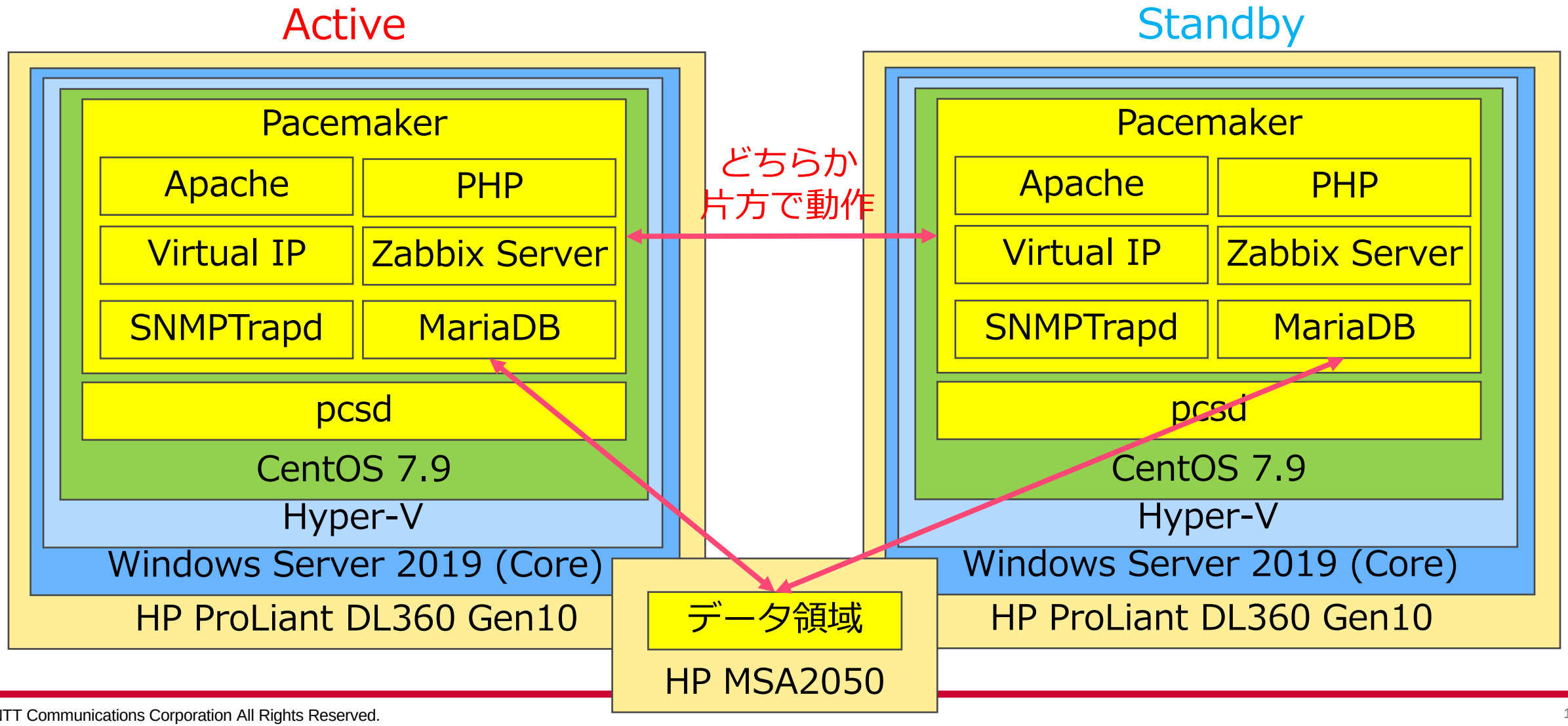
なんか案件数が増えてる

2-2.Zabbixへ統合イメージ

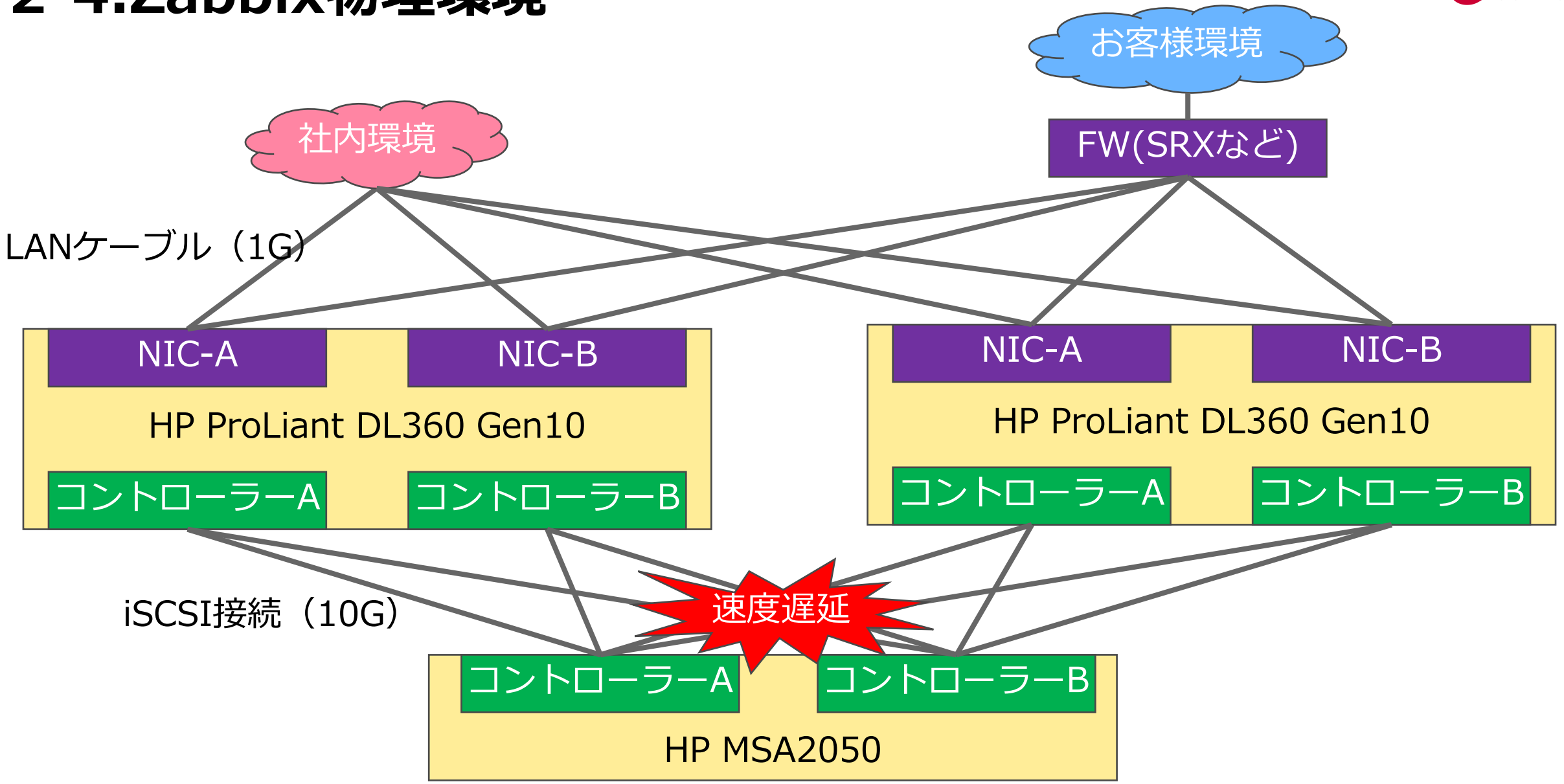


2-3.どんなZabbixを作ったのか

■スペック：Zabbix5.0　CPU：4コア　メモリ：45GBディスク：500GB　ストレージ：1TB



2-4.Zabbix物理環境



2-5.ストレージの速度がでない問題

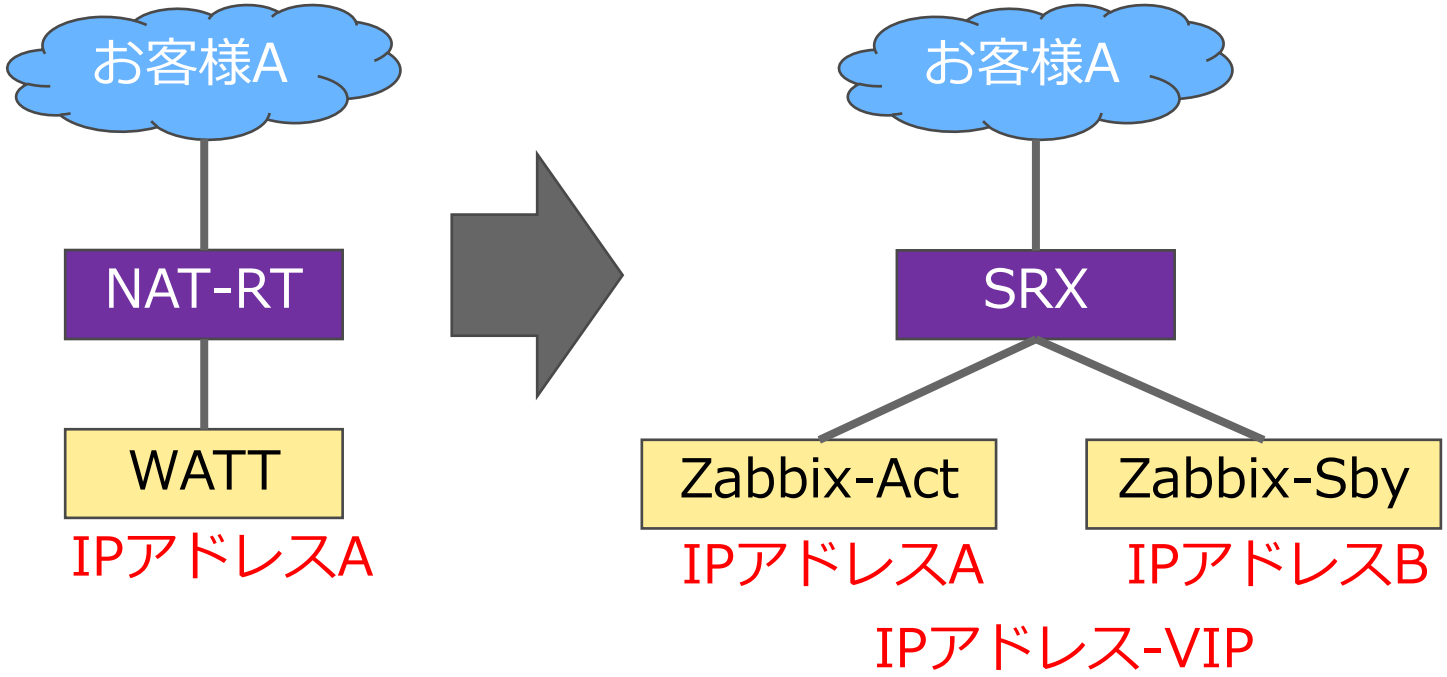
■原因調査や切り分け

・DACケーブルの交換
・ストレージのコントローラーを切り替え
・サーバやストレージのOSをアップデート
・ストレージのプールやマッピングの組み直し

■原因

HP社と一緒に２～３ヶ月間調査した結果、ストレージにてiSCSIをDACケーブルで利用した際に、発生する未知のバグであることが判明。→　パッチを提供いただき、ストレージの遅延問題が解消

2-6.ソースIP問題



■解消方法

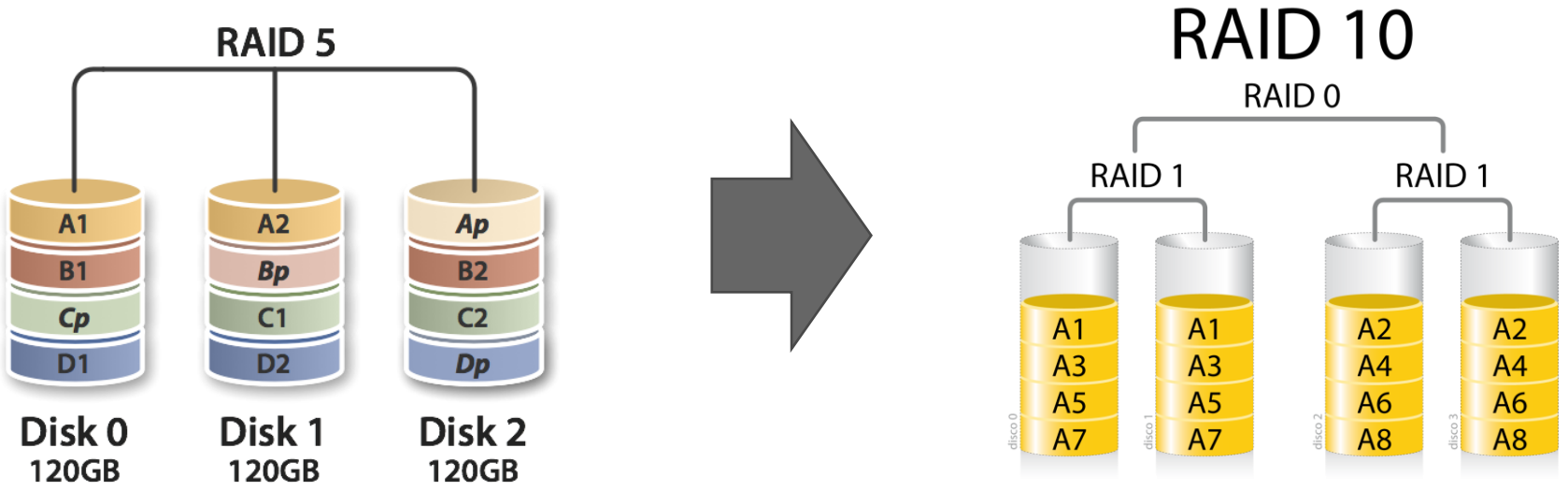
ZabbixにてSourceIP設定を実施して、

Zabbixから監視するIPアドレスを１つに統一

設定方法
■ディレクトリ
/etc/zabbix/
■ファイル
zabbix_server.conf
■設定内容
SourceIP=IPアドレス-VIP

2-7.RAID構成の工夫

ZabbixのスペックはDBの速度によって頭打ちになることが多いことから一番早いRAID10を選択



画像出典元：WIKIMEDIA COMMONS様 https://commons.wikimedia.org/

2-8.Windows Server 2019 Coreでの構築

■メリット

・必要なリソースが少ない

・モジュールが少ないのでセキュリティリスクが低く、
アップデート頻度が少ない

■デメリット

・GUIではなくCLIのため、操作が大変

→Hyper-Vを入れるだけならWindows Server Coreもあり！

3.他監視マネージャからZabbixへの移行

3.他監視マネージャからZabbixへの移行

3-1.Zabbixへの移行計画

3-2.ホスト・ユーザーグループによるロール管理

3-3.監視テンプレートの共用利用

3-4.ホスト一括登録

3-5.ホスト一括登録にて問題

3-6.大量監視によるキュー増加問題

3-7.Nagios移行時の工夫

3-8.WATT移行時の工夫

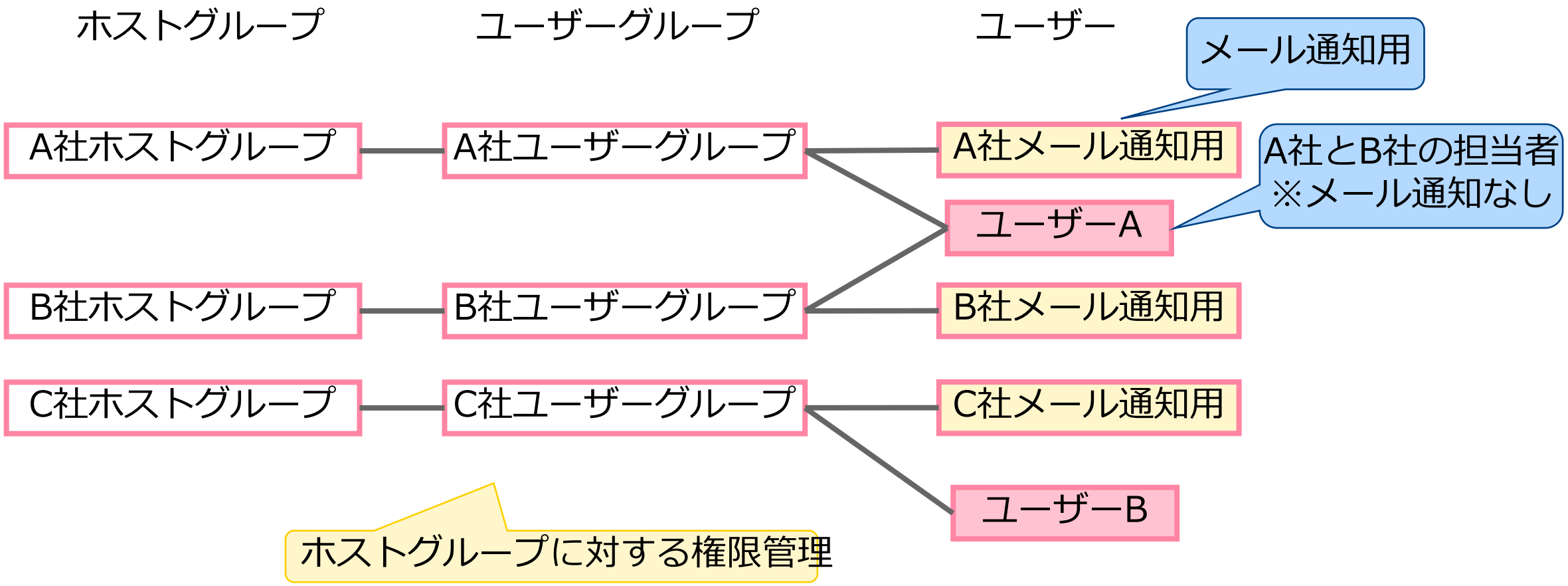
3-1.Zabbixへの移行計画

30案件を18ヶ月以内に移行してほしい

一ヶ月に1案件以上のペースか

しかも、一つのサーバにNagios、Cactiなど
複数の監視マネージャが乗ってる！？

3-2.ホスト・ユーザーグループによるロール管理



3-3.監視テンプレートの共用利用

■**監視テンプレート**

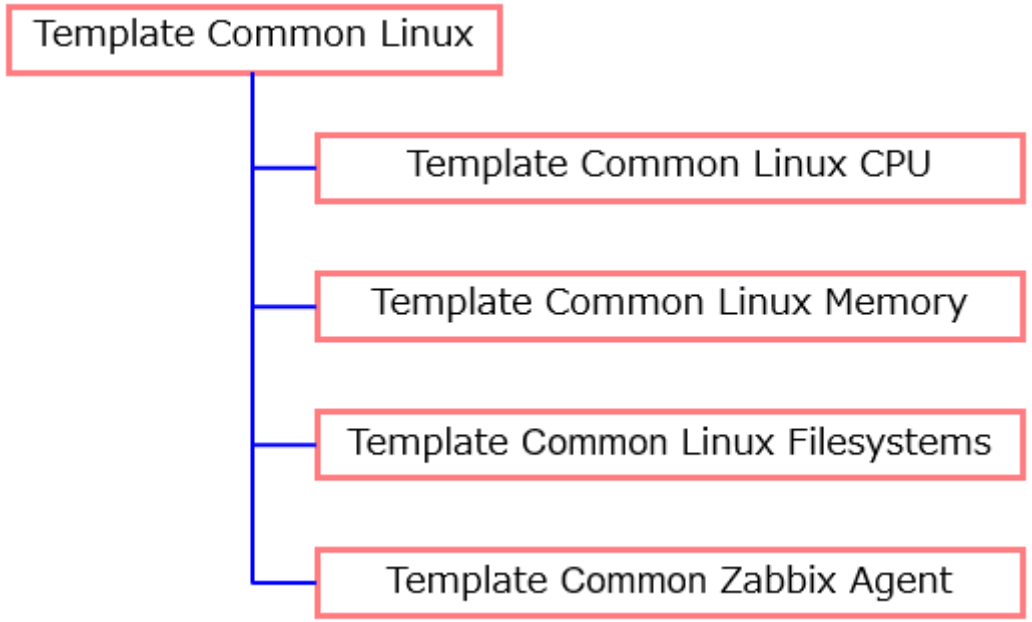
監視設定はテンプレート利用を必須として、テンプレート外での個別アイテム作成などはNG。

■共用テンプレート

死活監視、リソース監視（CPU、Mem、Disk）、TCP監視、プロセス監視、SNMP Trap監視

■個社テンプレート

ログ監視、スクリプト監視



3-4.ホストー括登録

■概要

弊社(NTTCom)のZabicomサービスにあるホストー括登録ツールを利用してCSVファイルで
ホストやインベントリなどの情報を一括登録

■動作の流れ

1.CSVでホストやインベントリなどの
　情報を作成

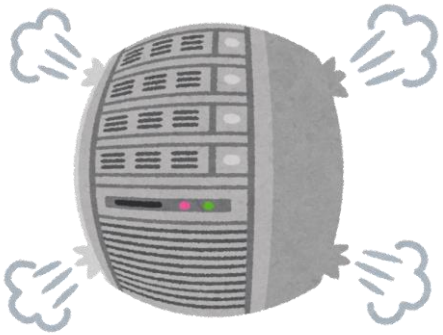
2.ダッシュボードの専用画面でCSV
　ファイルをアップロード

3.「登録開始」ボタン押下

3-5.ホスト一括登録にて問題

10,000ホストの一括登録実行！！

もう処理できません



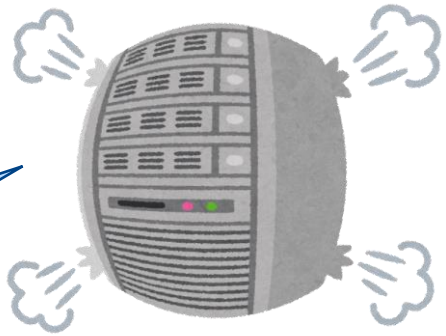
あれ、Zabbixが止まってるぞ

Cacheがオーバーして、処理ができないため、プロセスが起動不可となっていることが判明。

3-5.ホストー括登録にて問題

Cacheなら再起動すれば直るはず

キューを再読み込みしたら再発したよ



失敗した処理を再実行する仕組みはすごいが…ぐぬぬ

Zabbixで利用可能なCacheサイズを2倍にして、サービス再起動

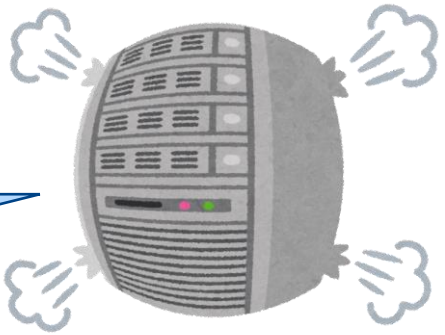
横着はダメ、絶対

設定方法
■ディレクトリ
/etc/zabbix/
■ファイル
zabbix_server.conf
■設定内容
CacheSize=300M

3-6.大量監視によるキュー増加問題

監視リソースに余裕があるから
監視追加しても大丈夫でしょ

Ping監視項目数38000件と
かプロセスが足りません



あれ、ZabbixのPing監視が
一部できていないし、キュー
が増えてるぞ

Pingプロセス数を100に増やして、
サービス再起動

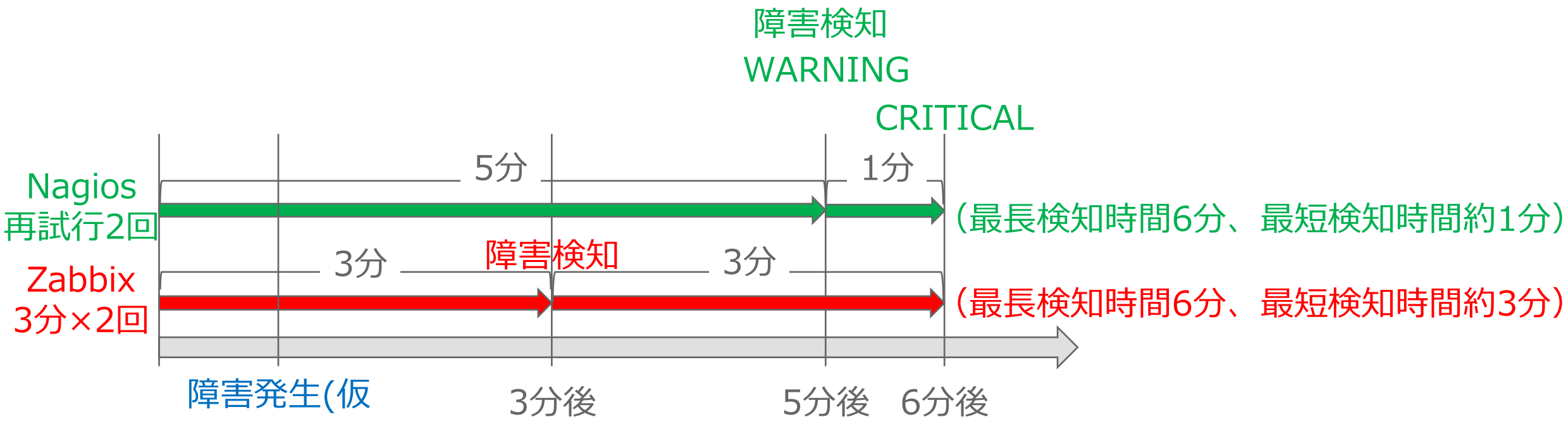
設定方法
■ディレクトリ
/etc/zabbix/
■ファイル
zabbix_server.conf
■設定内容
StartPingers=100

3-7.Nagios移行時の工夫

■監視間隔の違い

Nagiosでは障害発生時に監視間隔とは別にリトライ間隔がある。

そのため、Zabbixへ移行する際には、最長検知時間から逆算して監視間隔を調整。



3-8.WATT移行時の工夫

■簡易Ping確認

WATTには2StepでホストへPingを打ち、その結果を確認するといった機能ある。

Zabbixでも同様の機能を提供するため、スクリプトを実装して2Stepで実施できるようにした。

■スクリプト

名前：Ping3回

タイプ：スクリプト

次で実行：Zabbixサーバー

コマンド：fping -c 3 {HOST.CONN};

■実行結果

Ping/Ping 3回

```
fping -c 3 [redacted];

[redacted] : [0], 84 bytes, 19.0 ms (19.0 avg, 0% loss)
           : [1], 84 bytes, 12.2 ms (15.6 avg, 0% loss)
           : [2], 84 bytes, 14.1 ms (15.1 avg, 0% loss)

[redacted] : xmt/rcv/%loss = 3/3/0%, min/avg/max = 12.2/15.1/19.0
```

4.Zabbix運用フェーズ

4.Zabbix運用フェーズ

4-1.SNMP Trapの回復メールの通知抑止

4-2.故障通知メールにTraceroute結果を添付

4-3.スクリプト実行結果監視

4-4. High memory utilization 対応

4-1.SNMP Trapの回復メールの通知抑止

■概要

SNMP Trapは故障で残り続けるため、手動で復旧させているが、
復旧通知が飛んでしまい無駄なメール確認が必要になることから復旧通知と出さないようにする。

■方法

1.復旧通知を抑止したいトリガーにタグをつける

例：タグ名：SNMPTrap-Notification、タグ値：No

4-1.SNMP Trapの回復メールの通知抑止

2.アクションをタグの有無で分割する

(1)タグありアクション

①実行条件

タグ名 等しい SNMPTrap-Notification

②実行内容

■実行内容

(2)タグなしアクション

①実行条件

タグ名 等しくない SNMPTrap-Notification

②実行内容

■実行内容

■復旧時の実行内容

4-2.故障通知メールにTraceroute結果を添付

Tracerouteの結果を故障通知メールに載せたい

①リモートコマンド：Traceroute

故障 → アイテム → トリガー → イベント → アクション①

②結果をzabbix senderで送付

アイテム → インベントリ → アクション②

③Trapper itemで受領

④インベントリの情報を含めて
メール送付

4-2.故障通知メールにTraceroute結果を添付

■概要

故障箇所特定の迅速化ため、Zabbixから実施したtracerouteの結果を通知メールに添付。

tracerouteだけではなく、他のコマンドも実施可能

■動作の流れ

1.障害が発生し、アクション1のリモートコマンドを実行

2.アクション1のリモートコマンドの結果を対象ホストのインベントリ「INVENTORY.NOTES1」へ転送

3.対象ホストのアイテム「Trapper item」でtracerouteの結果を受領し、インベントリの「INVENTORY.NOTES1」へ格納

4.アクション2で、インベントリの「INVENTORY.NOTES1」を引用してユーザーにメッセージを送信

4-2.故障通知メールにTraceroute結果を添付

■アクション

①現在のホストでリモートコマンドを実行

```
cd /tmp;traceroute -d <対象IPアドレス> > VAR.txt;sleep 30;VAR=`sed s/*/-/g
VAR.txt`;zabbix_sender -z <ZabbixServerのIP> -s <対象ホスト> -k
INVENTORY.NOTES1 -o "$VAR"
```

②ユーザーにメッセージを送信

メール

*のままだと意図しない内容が含まれるため、-に置換

■メディア（メール）

テンプレート内に以下の内容を含める

Traceroute結果：{INVENTORY.NOTES}

4-2.故障通知メールにTraceroute結果を添付

■ホスト

●インベントリ

自動

●アイテム

名前：Trapper item

タイプ：Zabbixトラッパー

キー：INVENTORY.NOTES1

データ型：ログ

ヒストリの保存期間：90d

有効：チェック

4-3.スクリプト実行結果監視

■概要

NW機器のインターフェースpackets outputのスクリプト監視

packets output以外の値も監視可能

■動作の流れ

1.アイテムでスクリプト「timeout-drop.sh」を実行

2.スクリプトが結果を取得してアイテムのヒストリに返却

■**ファイル**

timeout-drop.sh　←実行スクリプト

timeout_drop.log　←実行結果一時格納

4-3.スクリプト実行結果監視

■timeout-drop.sh

対象ホストへログインして、インターフェース情報を取得し、一時ファイル（timeout_drop.log）へ格納。

grepなどで必要な箇所だけ切り取り

例：grep 'packets output' timeout_drop.log | awk '{print substr($0,0,index($0,"packets output"))}' | rev | cut -c 2- | rev | cut -b 6-

■アイテム

名前：Ping(VIP)

タイプ：外部チェック

キー：timeout-drop.sh[FastEthernet1/0/2]

データ型：数値(整数)

有効：チェック

もっとスマートなやり方があるはず、、、

4-4. High memory utilization 対応

■概要

ダッシュボードのZabbix server healthで以下のエラーが発生

High memory utilization (>90% for 5m)

■原因

topコマンドでメモリを調査したところphp-fpmが無駄にメモリを保有していることが判明

■暫定対応

php-fpm プロセスの再起動を実施

■恒久対応

cronでphp-fpm プロセスの定期再起動を実施

最後に

もともとは30案件あったシステムも、残り4ヶ月で7案件処理する予定です

これからも引き続き頑張ります。

また、新しい事例などありましたら、共有させていただきます。

ご清聴ありがとうございました。