

WHO AM I

```
『name": "Hayato Watanabe",
"company": "Zabbix Japan LLC",
"department": ["Support Engineer", "Trainer"],
"likes": ["Jiro", "△Camp", "⊷Roadracer"]
}
```

Agenda

- 4.0までの障害通知の流れ
- Webhookの活用化
- 導入例 / 作成例
- 大丈夫 (๑ ̇̀ ਖ ̇¸)



4.0までの障害通知の流れ: 主な選択肢2つ

- メールでの障害通知
- スクリプトを使用しチャットツール等への障害通知
- 異なるメッセージを使用したい場合、アクションを複数に分ける必要がある

アクション

メディア

メディアタイプ

通知

条件: FileServer001で障害発生時 以下をユーザAへ送信

件名: CPU高負荷

メッセージ:

5分間CPU高負荷検知 ホスト: FileServer001

值: 98.5 %

涌知者:

ユーザAの

ユーザメディアへ送信





以下をユーザAへ送信

件名: FileServer001 CPU高負荷

条件: FileServer001で障害発生時

メッセージ:

ホスト: FileServer001

值: 98.5%

涌知者:

送信内容多め

ユーザAの

ユーザメディアへ送信

送信内容少なめ





4.0までの障害通知の流れ: 課題

- スクリプトを使用した障害通知にはZabbixサーバ/Zabbixプロキシ上にスクリプト の配置が必要
 - Zabbixのフロント(Web)運用担当者とインフラ担当者運用担当者を分けにくい。
 - Pythonやruby等の使用も可能。各スクリプトのバージョン影響を受けやすい。
- 「とりあえず」メールで障害になった情報を全て受け取るという 運用にはそろそろ限界を感じ始めている
 - チャットツール等への通知がしてみたい。そのためにはスクリプトの作成が必要。
 - Zabbix上で発生した問題をプロジェクト内で自動的に管理出来るような仕組みが欲しい。

flexible project management



4.4からの変化点: ZBXNEXT-5386 メディアタイプにwebhookをサポート

- 障害通知の方法に、webhookをサポート
- RestAPIを提供するツールとの連携
- Webブラウザ上での操作のみで通知方法の追加



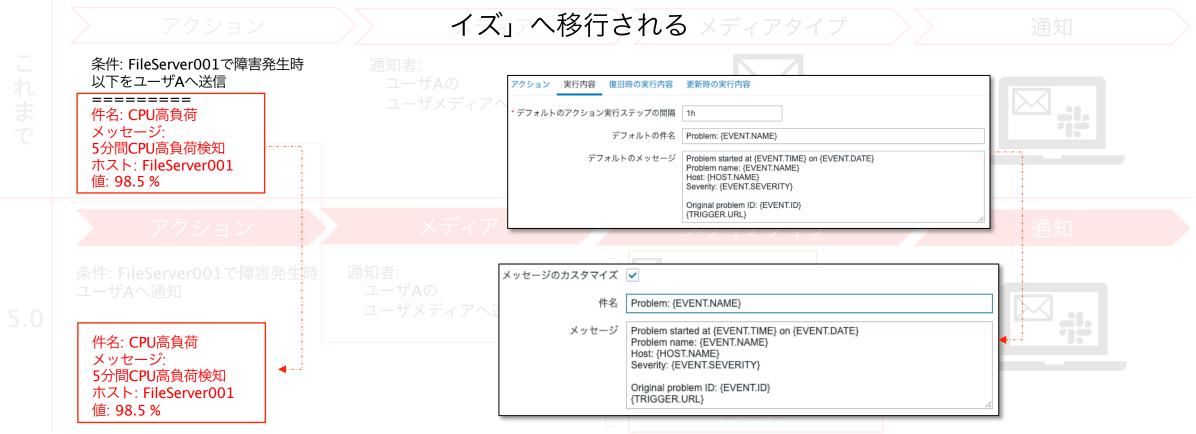
LTS(長期サポート版)の5.0が登場: 新しくなったメディアスクリプトの活用

• アクションで設定していた件名やメッセージ設定をメディアタイプへ移行



LTS(長期サポート版)の5.0が登場: 新しくなったメディアスクリプトの活

アクションで設定していたバージョンアップ時は従来どおりアクション側の件名・ メッセージを使用できるように、「メッセージのカスタマ



- LTS(長期サポート版)の 新しくなったメディア
 - 5.0からの機能を活用する為には
 - アクションで設定していた件名やディアタイプの設定
 - ユーザ設定で、送信したいユーザのメディアへ メディアタイプを割り当て

条件: FileServer001で障害発生時 以下をユーザAへ送信

=======

件名: CPU高負荷

メッセージ: 5分間CPU高負荷検知

ホスト: FileServer001

值: 98.5 %

ユーザAの• アクションを作成

1つの障害通知設定(アクション)を使用し、メッセージの内容を通 知するメディアタイプ設定(メール、Slackなどの通知方法)に変更 できる

件名: CPU高負荷 メッセージ: 5分間CPU高負荷検知 ホスト: FileServer001 值: 98.5 %



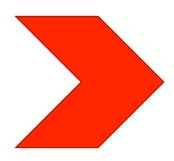
ポッセージ: ホスト「FileServer001」 で5分間CPU高負荷検知。 值: 98.5 %



メディアタイプはどれくらい増えたの?

5.0での活用化: 1つのアクションで様々な通知を提案

- **4.0**のメディアタイプ
 - Email
 - Jabber
 - SMS



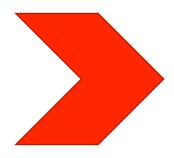
- **5.0**のメディアタイプ
 - ? ? ?

機能追加に伴い、どれだけ標準実装のメディアタイプが増えたかご存知ですか?

5.0での活用化: 1つのアクションで様々な通知を提案

- **4.0**のメディアタイプ
 - Email
 - Jabber
 - SMS

* Jabberは廃止されました



- Discord
- Email
- Email (HTML)
- iLert
- Jira
- Jira ServiceDesk
- Jira with CustomFields
- Mattermost
- MS Teams
- Opsgenie
- OTRS
- PagerDuty
- Pushover

- Redmine
- ServiceNow
- SIGNL4
- Slack
- SMS
- SolarWinds Service Desk
- SysAid
- Telegram
- TOPdesk
- Zammad
- Zendesk

21種類のWebhookメディアタイプを追加

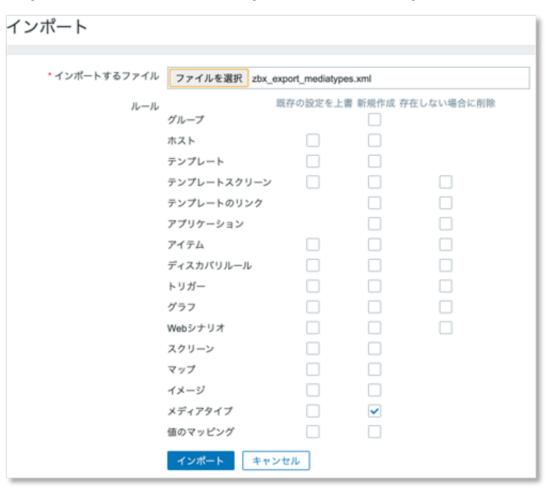
Zabbixアップグレードの場合 テンプレートは旧環境を引き継ぎます。

つまり、新テンプレートは 入っていない可能性があります。 大丈夫 (๑•`ㅂ•′) 卖

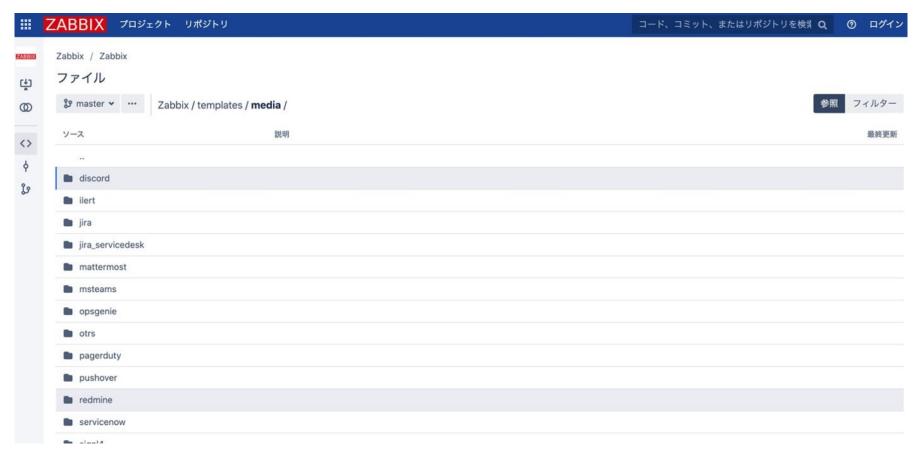
4.4からの機能追加:

ZBXNEXT-5372 メディアタイプのインポート・エクスポートをサポート

- XML形式でのインポートをサポート
- API経由ではJSONもサポート
- 旧バージョンからのアップグレードで も、随時追加されるメディアタイプを インポート

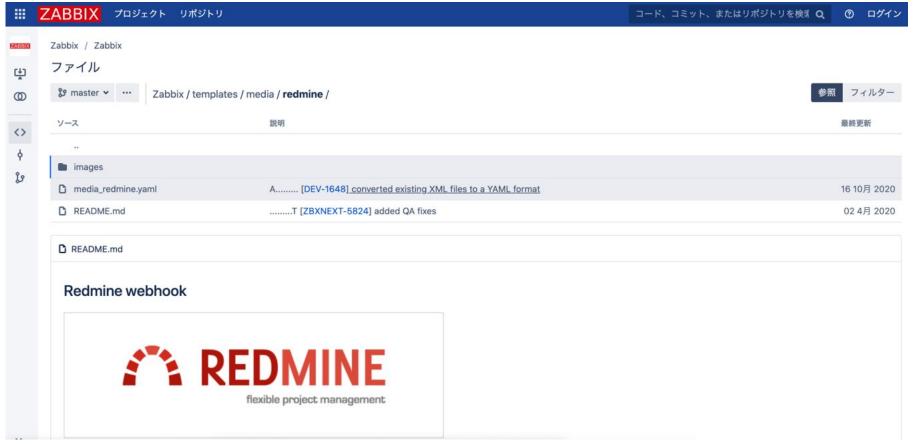


5.0での活用化: gitリポジトリから、テンプレートのダウンロードが可能



https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/templates/media

5.0での活用化: gitリポジトリから、テンプレートのダウンロードが可能



https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/templates/media/redmine

ね、大丈夫 (๑•`ㅂ•′) 💠



導入例:

例. 5.0でのRedmineチケット発行方法

- Redmineの設定
 - [管理]-[設定]-[API]でREST APIの許可を行う
 - 個人設定から、APIアクセスキーを用意する
 - Zabbixの「redmine_access_key」に必要
 - ・作成されたプロジェクト名と、トラッカーのIDを確認
 - Zabbixの「redmine_project」と「redmine_tracker_id」に必要。





導入例:

例. 5.0でのRedmineチケット発行方法

- Zabbixの設定
 - [ユーザ設定]-[メディアタブ]-[Redmine]のメディア作成
 - [管理]-[一般設定]のマクロに{\$ZABBIX.URL}を設定
 - ・メディアタイプのRedmineパラメータへRedmineの情報を設定

名前	值(例)
redmine_access_key	93d76e946ba4fdb4a54937cae0b71b13d471ca3c
redmine_project	Zabbix ticket
redmine_tracker_id	3
redmine_url	http://redmine.zabbix.lan/

導入例:

例. 5.0でのRedmineチケット発行方法

• フローのおさらい



さて、Zabbixのメディアタイプ 「Webhook」はどのように作られているのか 「Webhook」の中身はJavaScriptです。

∷ インベントリ

● 監視データ

山 レポート

設定

○ 管理

一般設定

プロキシ

認証

ユーザーグループ

ユーザー

メディアタイプ

```
JavaScript
```

```
1 try {
       Zabbix.Log(5, 'jira webhook script value='+value);
       var req = new CurlHttpRequest(),
       params = JSON.parse(value),
       fields = {},
       res = \{\},
       token;
       req.AddHeader('Content-Type: application/json-rpc');
       Zabbix.Log(4, 'webhook request value='+value);
10
11
       // user.login
12
13
       fields.method = 'user.login';
14
       fields.id = 1;
       fields.params = {"user": params.user, "password": params.password);
15
16
       fields.jsonrpc = '2.0';
                                                                                                                  適用
                                                                                                                          キャンセル
```

イベントメニュー項目を含める

JavaScriptの内容 1

cURLをカプセル化し、httpリクエストでのGet/Put/Post/Deleteを行う

```
メディアタイプへ設定した
パラメータ(value)をパース
{"redmine_access_key":"
93d76e946ba4fdb4a54937cae0b71b
13d471ca3c",
"redmine_project":" Zabbix ticket",
"redmine_tracker_id": "3",
"redmine_url":"http://192.168.0.125/
"}
```

```
var response,
       url = Redmine.params.url + query,
       request = new CurlHttpRequest();
Zabbix.Log(4, '[ Redmine Webhook ] Sending request: ' +
       url + ((typeof data === 'string') ? (' ' + data) : "));
     switch (method) {
       case 'get':
          response = request.Get(url, data);
         break;
try {
  var params = JSON.parse(value),
     params redmine = \{\},
     params_fields = \{\},
```

JavaScriptの内容 2

作成したメディアタイプの「タグとして処理」にチェックが入っている場合、結果からタグを設定する事も可能

Zabbix.Logを使用する事で zabbix_server.logへの出力が出来る 1-5までの指定はログのDebugLevelと 同様。

```
result.tags. zbx redmine issue id = issue id;
     result.tags. zbx redmine issuelink = params.redmine url +
       (params.redmine url.endsWith('/')?":'/') + 'issues/' + issue id;
  // Update created issue for trigger-based event.
  else {
    Redmine.updateIssue(params.alert_subject + '\forall n' + params.alert_message,
params fields);
  return JSON.stringify(result);
catch (error) {
  Zabbix.Log(3, '[ Redmine Webhook ] ERROR: ' + error);
  throw 'Sending failed: ' + error;
```

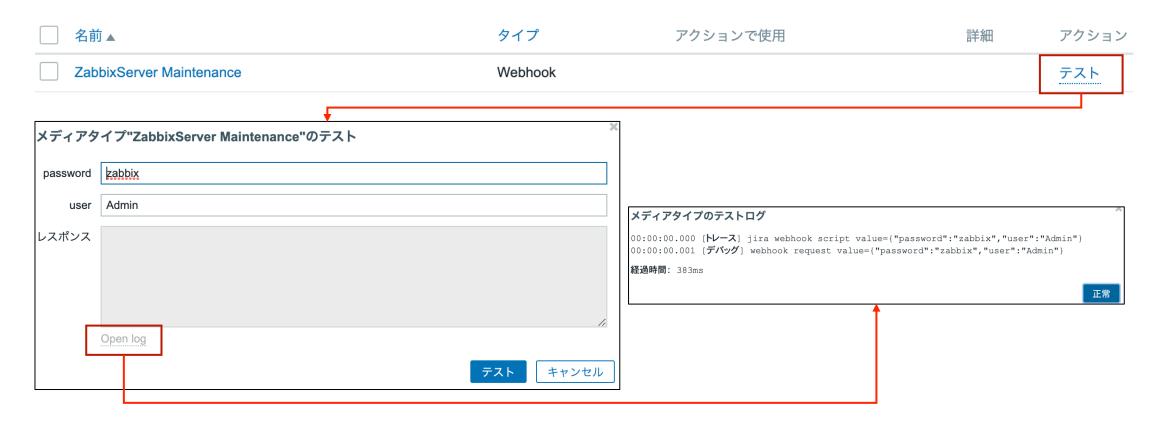
```
タグとして処理 ✓

イベントメニュー項目を含める ✓

* メニュー項目名 {EVENT.TAGS.rescode}

* メニュー項目URL {EVENT.TAGS.url}
```

動作試験はメディアスクリプトの「テスト」ボタンで可能



• テストボタンでも、ログへの出力設定を行っている場合はZabbixサーバログへ 記録されるので注意

WebUIでのTry & Errorは手間がかかる

- zabbix_js コマンドの提案
 - Zabbix 4.4 から用意されたコマンド
 - JavaScriptの動作試験をコマンドから行う
 - zabbix_jsコマンドでのログはzabbix_server.logへ出力されない
 - -pでパラメータの指定が必要 (引数を求めない場合は -p)

```
# zabbix_js -s ./zabbix_api.js -p '{"user":"Admin","password":"zabbix"}'
zabbix_js [22404]: Warning: jira webhook script value={"user":"Admin","password":"zabbix"}
zabbix_js [22404]: Warning: send:
{"method":"user.login","id":1,"params":{"user":"Admin","password":"zabbix"},"jsonrpc":"2.0"}
zabbix_js [22404]: Warning: send:
{"method":"user.logout","auth":"8e4ad8773d1aebcced024487c6d8b156","id":1,"params":{},"jsonrpc":"2.0"}
{"tags":{"rescode":"200", "url": "http://127.0.0.1/zabbix/api_jsonrpc.php"}}
```

メディアスクリプトで作成したタグから外部URLを参照



• 外部アプリケーション側のリンクをタグに載せる事で、作成された 情報の参照が容易に



不安を抱えていませんか?

 そういえば、2.2 -> 3.0 -> 4.0 -> 5.0とアップグレードはしているけど 設定の見直しはしていない。

今日の話を聞いてはみたけど、自分でやるには難しそうだしコストがかかりそう。

大丈夫 (๑•`ㅂ•′) 卖

技術的な面で不安を抱えている=相談

プレミアムパートナー

























