

ZABBIX マニュアル v1.6

レビューと承認

	名前	署名	日付
ZABBIX SIA:			

目次

目次.....	2
このマニュアルについて	11
序論.....	12
ドキュメントの目的.....	12
前提知識.....	12
対象読者.....	12
連絡先	13
用語集.....	14
リファレンス.....	17
関連ドキュメント.....	17
参考資料.....	17
1. はじめに.....	18
1.1. 変更履歴.....	18
1.2. 規約.....	18
1.3. 作成者.....	19
1.4. ZABBIX の概要.....	20
1.4.1. ZABBIX とは.....	20
1.4.2. ZABBIX の機能.....	20
1.4.3. ZABBIX の特長.....	20
1.4.4. ZABBIX ユーザ.....	21
1.5. 目標と原則.....	21
1.5.1. ZABBIX の開発における主な目標.....	21
1.5.2. ZABBIX の開発における主な原則.....	21
1.6. ZABBIX 1.6 の新機能.....	21
1.6.1. エスカレーションと繰り返し通知.....	21
1.6.2. パフォーマンスの改善.....	22
1.6.3. IPv6 のサポート.....	22
1.6.4. IPMI のサポート.....	22
1.6.5. 分散監視の改善.....	22
1.6.6. ZABBIX プロキシプロセス.....	22
1.6.7. ダッシュボード.....	22
1.6.8. 動的なスクリーン.....	22
1.6.9. グラフの詳細な拡大.....	22
1.6.10. 円グラフ.....	22

1.6.11.	基本的な管理機能	23
1.6.12.	より効率的なエージェントとの通信	23
1.6.13.	より効率的な ZABBIX Sender	23
1.6.14.	トリガーのステータスの表示を改善	23
1.6.15.	SNMP データの動的インデックスに対応	23
1.6.16.	既知の SNMP OID の特別な処理	23
1.6.17.	全てのスクリーンに印刷用表示機能を追加	23
1.6.18.	ユーザグループのログイン権限無効設定	23
1.6.19.	UTF-8 のサポートを追加	23
1.6.20.	翻訳の管理スクリーンを追加	24
1.6.21.	メンテナンスモードの追加	24
1.6.22.	制限のないマップリンク形式	24
1.6.23.	ユーザパーミッションスキーマの改善	24
1.6.24.	その他の改善点	24
1.6.24.1.	キューを管理に移動	24
1.6.24.2.	マップ、スクリーン、グラフへのリンクをダッシュボードに移動	24
1.6.24.3.	自動ログインオプション	24
1.6.24.4.	新しい通信プロトコル	24
1.6.24.5.	ZABBIX フロントエンドのテーマ機能	24
1.6.24.6.	ゲストユーザの無効化機能	24
1.6.24.7.	ユーザグループの無効化機能	25
1.6.24.8.	データベースダウンスクリーン	25
1.6.24.9.	二重ログインの拒否	25
1.6.24.10.	全てのスクリーンのソート機能	25
1.6.24.11.	インフォメーションメッセージの改善	25
1.6.24.12.	インポート/エクスポート機能がホストテンプレートのリンク情報をサポート	25
1.6.24.13.	グラフがマイナス値をサポート	25
1.6.24.14.	Include パラメータがディレクトリをサポート	25
1.6.24.15.	新しいマクロのサポート	25
1.6.24.16.	新しいログイン後のグリーティングメッセージ	25
1.6.24.17.	ICMP ping によるオートディスカバリ	25
1.6.24.18.	1 秒あたりに送信できるログエントリ数の増加	25
1.6.24.19.	ホストとトリガーに一括更新機能を追加	25
1.6.24.20.	フルスクリーンアイコンの追加	25
1.6.24.21.	ZABBIX エージェントのアクティブのみのモードを追加	26
1.6.24.22.	プロキシの可用性監視機能を追加	26
1.6.24.23.	ブルートフォースアタックに対抗するプロテクション機能を追加	26
1.6.24.24.	イベントの表示を改善	26
1.6.24.25.	より正確な ICMP ping	26
1.6.24.26.	より大量の認知をサポート	26
1.6.24.27.	可用性レポートの時間によるフィルター機能を追加	26
1.6.24.28.	アクションの履歴を管理の下に移動	26
1.6.24.29.	要求されるサーバのパフォーマンス値を取得可能	26
1.6.24.30.	自動ログインのサポートを追加	26
1.6.24.31.	ドロップダウンコントロールの 1 つ目の項目が自動選択されるように修正	26
1.6.24.32.	ユーザの最終アクセス時刻を表示	26
1.6.24.33.	より柔軟な[トリガーのステータス]スクリーン	26
1.6.24.34.	ホストプロファイルの拡張	27

1.7.	インストールとアップグレード	27
1.7.1.	インストール.....	27
1.7.2.	バージョンの互換性.....	27
1.7.3.	重要.....	27
1.7.4.	アップグレード手順.....	27
1.7.4.1.	ZABBIX サーバの停止.....	27
1.7.4.2.	現在の ZABBIX データベースのバックアップ.....	27
1.7.4.3.	設定ファイル、PHP ファイル、ZABBIX バイナリのバックアップ.....	27
1.7.4.4.	新しいサーババイナリのインストール.....	27
1.7.4.5.	サーバ設定パラメータのレビュー.....	28
1.7.4.6.	データベースのアップグレード.....	28
1.7.4.7.	新しい ZABBIX GUI のインストール.....	28
1.7.4.8.	新しい ZABBIX バイナリの起動.....	28
1.8.	有料サポート	28
2.	インストール	29
2.1.	ZABBIX の入手方法	29
2.2.	要件	29
2.2.1.	ハードウェア要件.....	29
2.2.1.1.	メモリ要件.....	29
2.2.1.2.	CPU 要件.....	29
2.2.1.3.	その他のハードウェア.....	29
2.2.1.4.	ハードウェア設定の例.....	29
2.2.2.	サポートされるプラットフォーム.....	30
2.2.3.	ソフトウェア要件.....	30
2.2.4.	データベースエンジンの選択.....	31
2.2.5.	データベースサイズ.....	31
2.2.6.	時刻の同期.....	33
2.3.	コンポーネント	34
2.3.1.	ZABBIX コンポーネント.....	34
2.3.2.	ZABBIX サーバ.....	34
2.3.3.	ZABBIX プロキシ.....	34
2.3.4.	ZABBIX エージェント.....	34
2.3.5.	ウェブインタフェース.....	34
2.4.	ソースからのインストール	35
2.4.1.	ソフトウェア要件.....	35
2.4.2.	ZABBIX の構造.....	36
2.4.3.	ZABBIX サーバ.....	37
2.4.4.	ZABBIX プロキシ.....	40
2.4.5.	ZABBIX エージェント.....	44
2.4.6.	ZABBIX ウェブインタフェース.....	46
2.5.	アップグレード	53

2.5.1.	データベースのアップグレード	53
3.	ZABBIX のプロセス	54
3.1.	ZABBIX サーバ.....	54
3.2.	ZABBIX プロキシ	57
3.3.	ZABBIX エージェント(UNIX、スタンドアロンデーモン).....	60
3.4.	ZABBIX エージェント(UNIX、INETD パージョン).....	63
3.5.	ZABBIX エージェント(WINDOWS)	64
3.5.1.	インストール	64
3.5.2.	使用方法	65
3.6.	ZABBIX SENDER(UNIX)	67
3.7.	ZABBIX GET(UNIX).....	68
4.	設定	70
4.1.	開発環境.....	70
4.2.	アクション	70
4.2.1.	アクションコンディション	71
4.2.2.	オペレーション	74
4.2.3.	メッセージとリモートコマンドでのマクロの使用	75
4.3.	マクロ.....	76
4.3.1.	サポートされているマクロのリスト	76
4.4.	アプリケーション	78
4.5.	グラフ.....	78
4.6.	メディア	79
4.6.1.	メール	79
4.6.2.	Jabber.....	79
4.6.3.	スクリプト.....	79
4.6.4.	GSM モデム	79
4.7.	ホストテンプレート	80
4.8.	ホストグループ	80
4.9.	ホストの依存関係とトリガーの依存関係.....	80
4.10.	アイテム.....	81
4.10.1.	アイテムキー	81
4.10.2.	プラットフォーム別のサポート	81
4.10.3.	ZABBIX エージェント.....	87
4.10.4.	SNMP エージェント	95
4.10.5.	シンプルチェック.....	97
4.10.5.1.	タイムアウト処理	99
4.10.5.2.	ICMP ping.....	99

4.10.6.	内部チェック	99
4.10.7.	累計チェック	101
4.10.8.	外部チェック	102
4.11.	ユーザパラメータ.....	102
4.11.1.	簡単なユーザパラメータ	102
4.11.2.	引数付きユーザパラメータ	103
4.12.	WINDOWS のパフォーマンスカウンタ.....	104
4.12.1.	簡単なユーザパラメータ	105
4.13.	トリガー.....	105
4.13.1.	トリガーの条件式	106
4.13.2.	トリガーの依存関係.....	112
4.13.3.	トリガーの深刻度	112
4.13.4.	履歴.....	113
4.14.	スクリーンとスライドショー.....	113
4.15.	IT サービス.....	115
4.16.	ユーザパーミッション	116
4.16.1.	概要	116
4.16.2.	ユーザの種類.....	116
4.17.	キュー.....	116
4.17.1.	概要	116
4.17.2.	キューの見方	117
4.18.	ユーティリティ.....	118
4.18.1.	スタートアップスクリプト	118
4.18.2.	snmptrap.sh	118
5.	クイックスタートガイド.....	120
5.1.	ログイン	120
5.1.1.	ブルートフォースアタックに対抗するプロテクション機能	121
5.2.	ユーザの追加.....	121
5.3.	メール設定	126
5.4.	エージェントを使用するホストの追加.....	127
5.5.	通知の設定	132
6.	XML のインポート/エクスポート.....	135
6.1.	目標.....	135
6.2.	概要.....	135
6.3.	データのエクスポート.....	135
6.4.	データのインポート.....	138

7.	チュートリアル	140
7.1.	ZABBIX エージェントの拡張	140
7.2.	ログファイルの監視	141
7.3.	リモートアクション	141
7.4.	WINDOWS サービスの監視	143
8.	エスカレーションと繰り返し通知	144
8.1.	目標	144
8.2.	概要	144
9.	ウェブ監視	145
9.1.	目標	145
9.2.	概要	145
9.3.	ウェブシナリオ	145
9.4.	ウェブステップ	147
9.5.	実際に使用するシナリオ	148
10.	ログファイルの監視	153
10.1.	概要	153
10.2.	ログファイルを監視するメカニズム	153
11.	オートディスカバリ	154
11.1.	目標	154
11.2.	概要	154
11.3.	オートディスカバリのメカニズム	155
11.3.1.	ディスカバリ	155
11.3.2.	アクション	155
11.4.	オートディスカバリルール	155
11.5.	実際に使用するシナリオ	156
12.	高度な SNMP 監視	160
12.1.	特別な MIB	160
12.2.	動的インデックスの使用	161
13.	IPMI デバイスの監視	163
13.1.	目標	163
13.2.	IPMI パラメータ	163
13.3.	IPMI アクション	163

14.	プロキシの使用	164
14.1.	プロキシの特長	164
14.2.	プロキシとノードの違い	164
14.3.	設定	165
15.	分散監視	166
15.1.	目標	166
15.2.	概要	166
15.3.	設定	167
15.3.1.	ノードの設定	167
15.3.2.	単純な設定	168
15.3.3.	複雑な設定	173
15.4.	プラットフォームの独立性	174
15.5.	単一ノードの設定	174
15.6.	ノード間のスイッチ	175
15.7.	データフロー	175
15.7.1.	子からマスターに送信されるデータ	175
15.7.2.	マスターから子に送信されるデータ	175
15.7.3.	ファイアウォール設定	176
15.8.	パフォーマンスに関する検討事項	176
16.	ZABBIX GUI のメンテナンスモード	177
16.1.	目標	177
16.2.	設定	177
16.3.	メンテナンスモードのスクリーン	178
17.	ウェブインタフェース	179
17.1.	独自のテーマの作成	179
17.2.	設定	180
17.2.1.	一般設定	180
17.2.1.1.	イベント	180
17.2.1.2.	housekeeper	181
17.2.1.3.	イメージ	182
17.2.1.4.	テーマ	184
17.2.1.5.	値のマッピング	185
17.2.1.6.	ワーキングタイム	187
17.2.1.7.	その他	188
17.2.2.	ウェブ	189
17.2.3.	ホスト	193
17.2.3.1.	ホスト	193

17.2.3.2.	テンプレート	197
17.2.3.3.	プロキシ	199
17.2.3.4.	ホストグループ	201
17.2.3.5.	テンプレートのリンク	202
17.2.3.6.	アプリケーション	205
17.2.4.	アイテム	207
17.2.4.1.	アイテム	207
17.2.5.	トリガー	213
17.2.5.1.	トリガー	213
17.2.6.	アクション	217
17.2.6.1.	アクション	217
17.2.7.	グラフ	219
17.2.7.1.	グラフ	219
17.2.8.	スクリーン	222
17.2.8.1.	スクリーン	222
17.2.9.	マップ	227
17.2.9.1.	マップ	227
17.2.10.	IT サービス	232
17.2.10.1.	IT サービス	232
17.2.11.	ディスカバリ	234
17.2.11.1.	ディスカバリ	234
17.2.12.	エクスポート/インポート	236
17.2.12.1.	エクスポート	236
17.2.12.2.	インポート	239
17.3.	管理	240
17.3.1.	認証	240
17.3.1.1.	HTTP	240
17.3.1.2.	LDAP	241
17.3.2.	ユーザ	242
17.3.2.1.	ユーザ	242
17.3.2.2.	ユーザグループ	245
17.3.3.	メディアタイプ	248
17.3.3.1.	メディアタイプ	248
17.3.4.	スクリプト	250
17.3.5.	監査	252
17.3.6.	キュー	254
17.3.7.	通知	257
17.3.8.	ロケール	258
17.3.9.	インストール	260
18.	パフォーマンスチューニング	261
18.1.	実際の設定	261
18.2.	パフォーマンスチューニング	261
18.2.1.	ハードウェア	261

18.2.2.	オペレーティングシステム	261
18.2.3.	データベースエンジン	262
18.2.4.	一般的なアドバイス.....	262
19.	具体的な設定例.....	263
19.1.	一般的な手法.....	263
19.1.1.	サーバの可用性の監視.....	263
19.1.2.	WinPopUps によるアラートの送信	263
19.2.	具体的なアプリケーションの監視	263
19.2.1.	AS/400	263
19.2.2.	MySQL.....	263
19.2.3.	Mikrotik ルータ	264
19.2.4.	WIN32.....	264
19.2.5.	Novell.....	264
19.2.6.	Tuxedo.....	265
19.2.7.	Informix.....	265
19.2.8.	JMX	265
19.3.	統合	267
19.3.1.	HP OpenView.....	267
20.	トラブルシューティング	269
20.1.	エラーメッセージと警告メッセージ.....	269
21.	ライセンス.....	270
22.	ZABBIX への貢献	276
23.	クレジット	278
23.1.	ZABBIX の開発者.....	278
23.2.	ZABBIX に対するコントリビュータ.....	278

このマニュアルについて

このマニュアルは、ZABBIX を利用する人をサポートする目的で配布されますが、市場性または特定目的に対する適合性を含むいかなる保証も一切行うものではありません。このマニュアルは ZABBIX ソフトウェアの一部として配布されます。最新版は <http://www.zabbix.com> から入手できます。

ZABBIX リファレンスマニュアルは、GPL ライセンスではなく、

以下の条項に従って配布および利用するものとします。

- 他の形式に翻訳および変換することは許可されますが、いかなる方法によっても内容を変更または編集することは禁じられています。
- 個人で使用する場合は印刷物の作成が許可されます。
- その他の利用目的、たとえば印刷物を販売する場合または別の出版物(印刷物または電子的)でこのマニュアル(の一部)を引用する場合は、ZABBIX 社からの事前の書面による合意が必要です。

詳細については、sales@zabbix.com までメールでお問い合わせください。

序論

ドキュメントの目的

本書では、ZABBIX の序論と概要、そのアーキテクチャ、特長、および機能について説明します。
ZABBIX を適正に管理するために必要な情報は全て本書に記載されています。

前提知識

UNIX に対する理解は必要不可欠ですが、詳細な技術的知識は必要ありません。

対象読者

本書は、ZABBIX のインストールと管理に関わる人および ZABBIX の内部動作に興味がある人を対象として書かれています。

連絡先

ZABBIX SIA

住所: Neretas 2/1-109, LV-1004, Riga, Latvia

電話: +371 7743943

Fax: +371 7743944

E メール: sales@zabbix.com

ZABBIX SIA, Product Manager, Director Alexei Vladishev

E メール: alexei.vladishev@zabbix.com

ZABBIX SIA, Sales

E メール: sales@zabbix.com

ZABBIX SIA, Customer Support

E メール: support@zabbix.com

用語集

用語	説明
アクティブ	ZABBIX エージェントの動作モード。アクティブに動作するエージェントは、サーバに送信するアイテムの種類と間隔を追跡します。送信中のアイテムを追跡するために、設定された間隔でサーバをポーリングすることができます。
アクティブチェッカ	アクティブチェッカは、ZABBIXエージェントが動作するシステムの稼動情報を収集し、ZABBIXで処理するために送信します。
アクション	アクションは、トリガーの状態が変化したときに実行される動作です。アクションとして、メディアタイプの設定に基づいてZABBIXで定義されている特定のユーザグループにメッセージを送信したり、リモートコマンドを実行するように設定できます。
エージェント	エージェントは、監視対象のホストで実行するプログラムです。サービスとして動作し、アクティブチェックとパッシブチェックを同時に処理できます。
アラータ	アラータは、アクション(Eメール、Jabber、SMS、スクリプト)を実行するサーバプロセスです。
自動登録	自動登録は、ホストが自らをZABBIXサーバに自動的に登録できるZABBIXの機能です。管理者はウェブインタフェースでホスト名パターン(「*-Linux」など)とそのホストで実行するアイテムを定義します。実行するアイテムは、アイテムのテンプレートを使用して定義します。
オートディスカバリ	ZABBIXオートディスカバリモジュールは、ホストとサービスのオートディスカバリを実行し、イベントを生成するモジュールです。イベントには処理を関連付けることができます。
イベント	イベントは、トリガーの状態が変化したときに生成されます。
グラフ	グラフには、簡易グラフとカスタムグラフがあります。簡易グラフは、監視する数値アイテムごとに表示できます。カスタムグラフは、複数の数値アイテムを1つのグラフにまとめて表示できます。
ホスト	ホストは、監視されるマシンです。
housekeeper	housekeeperは、ZABBIXサーバ上でサービスとして動作し、ユーザの定義に従ってZABBIXデータベースから古くなったアクション、イベント、ヒストリ、およびトレンドデータを削除します。アクションとイベントの削除については、[一般設定]で定義します。ヒストリとトレンドデータの削除については、アイテムごとに定義します。
IPMI	Intelligent Platform Management Interface。
IT サービス	ITサービスはZABBIXが提供する機能の1つです。ITサービスを使用してSLAを定義し、計画したSLAと実際のSLAを追跡するようにZABBIXに指示することができます。ITサービスはトリガーのグループとして定義され、グループの最小値や最大値を計算するように設定できます。
アイテム	アイテムは、ロードアベレージや応答時間など、ホストで実際に監視する個々のアイテムです。ZABBIXエージェントやSNMPなど、様々な手段で収集した情報をアイテムとして扱うことができます。データタイプとして、浮動小数点、64ビット整数、文字列、テキスト、またはログ値を設定できます。

ロケーション	単一ノードが監視する環境。
マップ	マップはZABBIXが提供する機能の1つです。ウェブインタフェースでグラフィックをカスタマイズして、ネットワークマップを作成したり、マップ上でホスト間のリンクを定義することができます。リンクは、トリガーに応じて色やスタイルを変えるように設定できます。
マスターまたはマスターノード	マスターノード。マスターノードには1つまたは複数の子ノードを関連付けることができます。 マスターノードから子ノードの設定をコントロールできます。
メディアタイプ	メディアタイプは、アクションが発生したことをZABBIXユーザに通知する手段です。メディアタイプとして、メールまたはカスタムスクリプトを指定できます。
ノード	複数のホストを監視する分散設定におけるZABBIXサーバ。
ノード ID	ノードIDは、ノードを識別する一意な番号です。各ノードには一意なノードIDを付ける必要があります。
Node Watcher	ノード間通信を処理するZABBIXサーバプロセス。
キュー	キューは、ZABBIXサーバが監視しているアイテムの内部キューです。ZABBIXサーバは、アイテムごとに指定された間隔で、アイテムを追跡するためのキューおよびポーリングするタイミングを管理します。
パッシブ	ZABBIXエージェントの動作モード。 パッシブに動作するエージェントは、アイテムに対するサーバからの要求を待機し、要求を受信したらアイテムを返信します。通常はエージェントはアクティブとパッシブの両方のモードで動作しており、モードはアイテムを設定する際に個別に定義することに注意してください。
Pinger	ICMPのpingを処理するZABBIXサーバプロセス。
Poller	ZABBIXエージェントとSNMPエージェントからデータを収集し、リモート(シンプル)チェックを処理するZABBIXサーバプロセス。
プロキシ	サーバとネットワークデバイスからパフォーマンスおよび可用性データを収集し、それらのデータを処理するためにZABBIXサーバに送信するZABBIXプロキシプロセス。
ROI	投資収益率。
スクリーン	スクリーンは、ZABBIXのカスタマイズ可能な機能の1つで、様々な情報を表示するためのカスタムページをZABBIX内に作成できます。スクリーンには、グラフ(カスタム)、簡易グラフ、マップ、またはプレーンテキストを表示できます。プレーンテキストには、たとえば特定のアイテムの最新の5個の値を表示できます。
Sender	ZABBIXサーバで処理するためにデータを送信するZABBIXユーティリティ。通常は、ユーザスクリプトで使用します。
サーバ	サーバは、「監視ステーション」として位置付けられている中央のマシン上で動作するプログラムです。サーバはサービスとして動作し、設定されている全てのホスト、アイテム、アクション、アラートなどを追跡します。
SLA	サービスレベルアグリーメント。通常は、企業とクライアントの間の契約において「特定のホストの可用性が99.5%であること」というように一定のサービスレベルを明記するために使用します。

子または子ノード	子ノードはマスターノードに関連付けられています。子ノードはマスターノードに情報を通知します。
テンプレート	テンプレートにはアイテムやトリガーなどが定義されており、ホストを関連付けることができます。テンプレートを使用すると、複数のホストの設定や変更を一元管理でき、個々のホストを変更する必要はありません。ホストテンプレートは、設定時にそのステータスが[テンプレート]に設定されていて実際には監視されないことを除けば、他のホストと同じです。
Timer	トリガー条件式の日時に関する関数を処理するZABBIXサーバプロセス。
Trapper	ZABBIXエージェント(アクティブ)チェック、ログファイル、およびsenderから送信されるデータを処理するZABBIXサーバプロセス。
トリガー	トリガーは、アイテムに対する閾値を定義し、閾値を超過した場合に通知を送信するために使用します。たとえば、特定のホストのロードアベレージを監視して、その値が1.0を超えたときに通知を受け取ることができます。
ユーザ	ZABBIXのウェブフロントエンドは、複数のユーザが複数のアクセスレベルでアクセスできるように設定できます。たとえば、guestアカウントとして匿名でアクセスできるユーザ、あらゆるデータを参照できるけれども変更は一切禁じられているユーザ、またはZABBIXの特定のセクションに対する表示または変更だけが可能なユーザなどを設定できます。
ユーザパラメータ	ユーザパラメータ(UserParameter)は、エージェントの設定ファイルに定義するカスタムスクリプトです。ユーザパラメータは、キーとコマンドで定義します。キーは、ウェブインタフェースで定義するアイテムであり、サーバから送信される引数を受け取るように設定できます。
ZABBIX	ZABBIXソフトウェア。
ZABBIX SIA	ZABBIXを開発し、サポートを行うラトビアの企業。

リファレンス

ZABBIX の技術的な詳細情報については、以下のサイトを参照してください。

関連ドキュメント

現在のところ、関連ドキュメントはありません。

参考資料

- hdparm: <http://freshmeat.net/projects/hdparm/>
- Microsoft: <http://www.microsoft.com>
- MySQL: <http://www.mysql.com>
- Oracle: <http://www.oracle.com>
- PHP: <http://www.php.net>
- PostgreSQL: <http://www.postgresql.org>
- SQLite: <http://www.sqlite.org>
- Sqlora8: <http://www.poitschke.de>
- SuSE Linux: <http://www.suse.com>
- Ubuntu Linux: <http://www.ubuntu.com>
- ZABBIX: <http://www.zabbix.com>

1. はじめに

1.1. 変更履歴

リリース	日付	変更理由	変更者
13	10/04/2008	リリースノートの更新	Alexei Vladishev
15	18/09/2008	1.6 以前のバージョンの更新	Alexei Vladishev
16	04/11/08	追加: <ul style="list-style-type: none"> ■ メンテナンスモード ■ 設定(ホスト、ウェブ、トリガー、グラフ、マップなど全て) ■ パラメータ StartDBSyncers、 BufferSend、BufferSize ■ 新しいテーマの作成 ■ キーzabbix[proxy,...] 	Alexei Vladishev
17	04/11/08	追加: <ul style="list-style-type: none"> ■ 通知のパーミッション チェックに関する情報 ■ {TRIGGER.NSEVERITY }の説明 ■ last(回数)の説明 ■ セキュリティで保護された LDAP 接続に関する情報 ■ エラーコード(バージョン 1.8) ■ システムマップ用のマクロ (バージョン 1.8) ■ アイテムのデータタイプ (バージョン 1.8) 	Alexei Vladishev

1.2. 規約

表記規約

ZABBIX のマニュアルでは、以下に示す表記規約を使用します。

形式	定義
ファイル名	ファイルまたはディレクトリの名前
重要	注意事項、重要情報、強調

シェルコマンド	シェルコマンド、パス、設定ファイル
定数	定数、設定パラメータ
注: 注意事項	注意事項、コメント、追加情報

1.3. 作成者

作成者	変更点
Alexei Vladishev	作成およびメンテナンス
Charlie Collins	初版(LyX)ドキュメントの多くの改善
Shawn Marriott	ZABBIX マニュアル v1.0 の校正

1.4. ZABBIX の概要

1.4.1. ZABBIX とは

ZABBIX は Alexei Vladishev によって作成され、現在は ZABBIX SIA が積極的に開発およびサポートを進めているエンタープライズクラスの監視ソリューションであり、オープンソースで配布されています。

ZABBIX は、ネットワークの膨大な量のパラメータおよびサーバの稼動状態や整合性を監視します。柔軟性の高い通知メカニズムを備え、ほとんど全てのイベントをメールベースのアラートで通知するように設定できるので、サーバの障害に迅速に対応できます。収集したデータから、見やすいレポートを作成したり、グラフィカルに表示することができます。そのため、キャパシティプランニングに最適なソフトウェアです。

ZABBIX はポーリングとトラッピングの両方をサポートします。ZABBIX の全てのレポート、統計情報、および設定パラメータには、ウェブベースのフロントエンドからアクセスできます。

フロントエンドはウェブベースなので、ネットワークのステータスやサーバの状態をどこからでも確認できます。ZABBIX を適切に設定することで、IT インフラストラクチャを監視する重要なツールとして使用できます。その動作は規模の影響を受けないので、数台のサーバを運用する中小の組織から大量のサーバを保有する大企業まで等しく対応できます。

ZABBIX は無料です。GPL(General Public License)バージョン 2 の下で作成および配布されているので、ソースコードは無料で配布され、誰でも自由に利用できます。ZABBIX 社は無料および有料のサポートを提供します。

1.4.2. ZABBIX の機能

ZABBIX は以下の機能を提供します。

- サーバおよびネットワークデバイスのオートディスカバリ
- 中央のウェブ管理インタフェースからの分散監視
- ポーリングとトラッピングのサポート
- サーバソフトウェアは Linux、Solaris、HP-UX、AIX、FreeBSD、OpenBSD、OS X に対応
- ハイパフォーマンスな専用エージェント(クライアントソフトウェアは Linux、Solaris、HP-UX、AIX、FreeBSD、OpenBSD、OS X、Tru64/OSF1、Windows NT 4.0、Windows 2000、Windows 2003、Windows XP、Windows Vista に対応)
- エージェントレス監視
- セキュリティで保護されたユーザ認証
- 柔軟なユーザパーミッション管理
- ウェブインタフェース
- 事前定義されたイベントをメールベースの柔軟なアラート機能で通知
- 監視対象リソースのハイレベル(ビジネス向け)な表示機能
- 監査ログ

1.4.3. ZABBIX の特長

- オープンソースソリューション
- UNIX ベースと WIN32 ベースのプラットフォームに対応した高性能エージェント

- インタフェースが直観的ですぐに習熟できる
- ROI が高い(ダウンタイムは非常に高コストのため)
- 所有コストが低い
- 設定が単純でわかりやすい
- 集中化された監視システム(設定やパフォーマンスデータなど全ての情報をリレーショナルデータベースに格納)
- 高度なサービスツリー
- 簡単にセットアップ可能
- SNMP(v1 と v2)のポーリングとトラッピングをサポート
- 監視データをグラフィカルに表示
- 設定により古いデータを削除

1.4.4. ZABBIX ユーザ

世界中のあらゆる規模の企業や組織が、メイン監視プラットフォームとして ZABBIX を採用しています。

1.5. 目標と原則

1.5.1. ZABBIX の開発における主な目標

ZABBIX は、以下の目標を設定しています。

- オープンソースの監視ソフトウェアとして認識されるようになること
- ZABBIX ユーザグループを創設し、ZABBIX のさらなる改良に努めること
- 高品質な有料サポートを提供すること

1.5.2. ZABBIX の開発における主な原則

- ユーザの使い勝手を重視すること
- 何事も単純化すること
- 処理に使用するリソースをできるだけ少なくすること
- 迅速に対応すること
- ソフトウェアのあらゆる側面をドキュメント化すること

1.6. ZABBIX 1.6 の新機能

1.6.1. エスカレーションと繰り返し通知

エスカレーションと繰り返し通知の機能が実装されました。

エスカレーションは非常に柔軟に設定することができ、通知だけでなくリモートコマンドや IPMI コマンドを実行することもできます。

1.6.2. パフォーマンスの改善

StartDBSyncers パラメータで有効化できる ZABBIX データベースキャッシュモジュールは、4~8 重以上の依存関係を持った設定で動作速度を改善します。

1.6.3. IPv6 のサポート

全ての ZABBIX モジュールは IPv4 と IPv6 の両方をサポートします。ZABBIX は混在または IPv6 のみの環境で利用することができます。

1.6.4. IPMI のサポート

ZABBIX は IPMI パラメータの監視をサポートし、ZABBIX フロントエンドからリモートコマンドと同様に IPMI コマンドを手動で実行することができます。

1.6.5. 分散監視の改善

より効率的なノード間の同期プロトコルにより、ZABBIX 分散監視が改善されました。

詳細については、「ZABBIX プロキシ」も参照してください。

1.6.6. ZABBIX プロキシプロセス

ZABBIX プロキシは、ZABBIX サーバの代わりにデータ収集を行う軽量なプロセスです。プロキシを利用し、分散環境において中央のサーバまたは ZABBIX ノードの 1 つにデータを送信することによって、遠隔地の集中監視を行うことができます。

ZABBIX プロキシは、集中化された分散監視の展開とメンテナンスを非常に単純化します。

1.6.7. ダッシュボード

ZABBIX ダッシュボードは、監視している環境の詳細について高度なパーソナライズ機能を提供します。現在、この機能は ZABBIX フロントエンドの中心となります。

1.6.8. 動的なスクリーン

スクリーンの要素を動的に生成することができます。この場合、要素に表示される情報は ZABBIX ユーザによって選択された特定のホストに依存します。

1.6.9. グラフの詳細な拡大

ドリルダウン分析のためにマウスで拡大する期間を選択することができます。

1.6.10. 円グラフ

円グラフ(2D と 3D)を利用することができます。

1.6.11. 基本的な管理機能

複数のスクリーンから Traceroute と Ping を実行することができます。その他にもスクリプトを追加、設定することができます。

スクリプトは、1 つの ZABBIX サーバ、または分散環境では任意の ZABBIX ノードで実行されます。

1.6.12. より効率的なエージェントとの通信

新しい設定パラメータの BufferSize と BufferSend によって、ZABBIX エージェントのデータバッファリング機能を有効にすることができます。

通信プロトコルは、1 つの TCP 接続で複数のデータを送信するように改善されました。

1.6.13. より効率的な ZABBIX Sender

ZABBIX Sender は、1 つのネットワーク接続で複数のデータを送信するように改善されました。

1.6.14. トリガーのステータスの表示を改善

トリガーとそれに関連するイベントの情報をスクリーンに表示することができます。

1.6.15. SNMP データの動的インデックスに対応

新しい構文で動的インデックスを持った SNMP データを監視することができます。詳細については、「SNMP」を参照してください。

1.6.16. 既知の SNMP OID の特別な処理

ifDescr、ifInOctets などのようなシンプルな SNMP OID は、ZABBIX によって自動的に正しい数値表現に変換されます。

1.6.17. 全てのスクリーンに印刷用表示機能を追加

全てのスクリーンは印刷用リンクをクリックすることにより簡単に印刷することができます。

1.6.18. ユーザグループのログイン権限無効設定

ユーザグループを指定して ZABBIX フロントエンドにアクセスできないように設定することができます。

1.6.19. UTF-8 のサポートを追加

ZABBIX フロントエンドに UTF-8 を使用することができます。ただし、ZABBIX データベースと ZABBIX サーバ、およびエージェントはまだ UTF-8 のデータを正しく処理することができません。

1.6.20. 翻訳の管理スクリーンを追加

ZABBIX フロントエンドに新たに追加された翻訳のスクリーンを利用することができます。

1.6.21. メンテナンスモードの追加

ZABBIX メンテナンスモードを有効にすると、ZABBIX フロントエンドを一時的に無効にすることができます。

1.6.22. 制限のないマップリンク形式

マップリンクに複数のトリガーをリンクさせることができます。これらのトリガーはリンクの表示形式も定義します。

1.6.23. ユーザパーミッションスキーマの改善

1.6 のユーザパーミッションは 1.4 から多少変更されています。

1.6.24. その他の改善点

1.6.24.1. キューを管理に移動

現在、情報を表示できるのは ZABBIX 特権管理者のみです。

1.6.24.2. マップ、スクリーン、グラフへのリンクをダッシュボードに移動

メインメニューが単純化されています。現在、ダッシュボードからマップ、スクリーン、グラフにアクセスすることができます。

1.6.24.3. 自動ログインオプション

ユーザプロファイルオプションにより、1 ヶ月間、ZABBIX フロントエンドへの自動ログインが可能になります。

1.6.24.4. 新しい通信プロトコル

より効率的な新しい通信プロトコルにより、1 つの TCP 接続で複数のデータを送信することができます。

1.6.24.5. ZABBIX フロントエンドのテーマ機能

新しいフロントエンドには、デフォルトで 2 つのテーマが用意されています。また、他のテーマも追加することができます。

1.6.24.6. ゲストユーザの無効化機能

この場合、ZABBIX フロントエンドへのアクセスにユーザ認証が必要になります。

1.6.24.7. ユーザグループの無効化機能

ユーザグループを無効化することができます。

1.6.24.8. データベースダウンスクリーン

ZABBIX フロントエンドがデータベースと通信できない場合、便利なスクリーンが表示されます。

1.6.24.9. 二重ログインの拒否

混乱を避けるため、[ログイン]メニュー項目が削除されています。

1.6.24.10. 全てのスクリーンのソート機能

ZABBIX フロントエンドのほとんどのテーブルを選択した列でソートすることができます。

1.6.24.11. インフォメーションメッセージの改善

インフォメーションメッセージの色がステータスに応じて変わります。また、デフォルトでは非表示の詳細が含まれる場合もあります。

1.6.24.12. インポート/エクスポート機能がホストテンプレートのリンク情報をサポート

XML インポート/エクスポート機能でホストテンプレートのリンク情報がサポートされています。

1.6.24.13. グラフがマイナス値をサポート

グラフにマイナス値を表示することができます。

1.6.24.14. Include パラメータがディレクトリをサポート

Include パラメータを使用してディレクトリ内の全てのファイルを含めることができます。

1.6.24.15. 新しいマクロのサポート

新しいマクロが追加され、通知に活用することができます ({EVENT.DATE}、{EVENT.TIME}、{EVENT.AGE}、{ESC.HISTORY})。

1.6.24.16. 新しいログイン後のグリーティングメッセージ

最初に表示されるメッセージがわかりやすくなっています。

1.6.24.17. ICMP ping によるオートディスカバリ

オートディスカバリで ICMP ping によるディスカバリがサポートされています。

1.6.24.18. 1 秒あたりに送信できるログエントリ数の増加

デフォルトでは、各ログファイルで 1 秒あたりに 100 行しか送信されません。

1.6.24.19. ホストとトリガーに一括更新機能を追加

ホスト属性とトリガー属性の一部を一括で更新することができます。

1.6.24.20. フルスクリーンアイコンの追加

ほとんどのスクリーンでフルスクリーンモードがサポートされ、フルスクリーンアイコンによってコン

ロールすることができます。

1.6.24.21. ZABBIX エージェントのアクティブのみのモードを追加

エージェントに対してアクティブのみのモードを有効にすることができます。この場合、エージェントでは、セキュリティ面で重要となる可能性のある受信接続の待ち受けは行われなくなります。

1.6.24.22. プロキシの可用性監視機能を追加

新しい内部チェックを使用してプロキシの可用性を自動的に監視することができます。

1.6.24.23. ブルートフォースアタックに対抗するプロテクション機能を追加

ZABBIX フロントエンドは、ブルートフォースアタックから保護されます。

1.6.24.24. イベントの表示を改善

単一の全てのイベントにおいて、実行されたコマンドと通知に関する詳細情報が表示されます。

1.6.24.25. より正確な ICMP ping

ICMP ping の更新速度をアイテムごとに個別にコントロールすることができます。

1.6.24.26. より大量の認知をサポート

大量認知により、1回のクリックで複数のイベントを認知することができます。

1.6.24.27. 可用性レポートの時間によるフィルター機能を追加

可用性レポートで期間の選択がサポートされています。

1.6.24.28. アクションの履歴を管理の下に移動

アクションとリモートコマンドの履歴が[管理]->[監査]に移動されました。

1.6.24.29. 要求されるサーバのパフォーマンス値を取得可能

値は ZABBIX の最適なパフォーマンスインジケータとなり、ハードウェア要件に使用することができます。

1.6.24.30. 自動ログインのサポートを追加

ユーザレベルでの自動ログインがオプションで1ヶ月間サポートされています。

1.6.24.31. ドロップダウンコントロールの1つ目の項目が自動選択されるように修正

全てのドロップダウンコントロールの1つ目の項目がデフォルトで選択されます。

1.6.24.32. ユーザの最終アクセス時刻を表示

ユーザの最終アクセス時刻が表示されます。

1.6.24.33. より柔軟な[トリガーのステータス]スクリーン

[トリガーのステータス]スクリーンにトリガーとそれに対応するイベントに関する情報が表示されます。

1.6.24.34. ホストプロファイルの拡張

拡張されたホストプロファイルをオプションで使用することができます。

1.7. インストールとアップグレード

1.7.1. インストール

詳細については、「インストール」を参照してください。

1.7.2. バージョンの互換性

ZABBIX 1.6 では、ZABBIX 1.0、ZABBIX 1.1.x、および ZABBIX 1.4.x の旧エージェントを使用することができます。エージェントサイドで設定を変更する必要はありません。

1.7.3. 重要

ユーザパーミッションスキーマが変更されました。デフォルトでは、ZABBIX 管理者に対して全てのホストへの書き込み可能アクセス権は付与されません。

ZABBIX 1.6 では、空のユーザパスワードは使用できません。データベースをアップグレードすると、空のパスワードは全て「zabbix」に置き換えられます。ただし、ゲストユーザは例外です。

1.7.4. アップグレード手順

ZABBIX 1.4.x から 1.6 にアップグレードするには、以下の手順を実行する必要があります。

ZABBIX データベースのサイズによっては、全てのアップグレード手順を完了するまでに数時間かかる場合があります。

1.7.4.1. ZABBIX サーバの停止

ZABBIX サーバを停止して、データベースに新しいデータが追加されないようにします。

1.7.4.2. 現在の ZABBIX データベースのバックアップ

これは非常に重要な手順です。必ずデータベースのバックアップを作成してください。アップグレード手順が失敗した場合(ディスク容量不足、電源断、予期しない問題が発生した場合)に、このバックアップが必要になります。

1.7.4.3. 設定ファイル、PHP ファイル、ZABBIX バイナリのバックアップ

設定ファイル、PHP ファイル、および ZABBIX バイナリのバックアップコピーを作成します。

1.7.4.4. 新しいサーババイナリのインストール

そのまま使用できるコンパイル済みのバイナリが提供されていますが、ユーザが独自にコンパイルしたバイナリを使用することもできます。

1.7.4.5. サーバ設定パラメータのレビュー

1.6では `zabbix_server.conf` の一部のパラメータが変更され、新しいパラメータが追加されています。必要に応じて、新しいパラメータを確認してください。

1.7.4.6. データベースのアップグレード

ディレクトリ `upgrades/dbpatches/1.6/<DB エンジン>` に、データベースごとのアップグレードスクリプトが用意されています。

MySQL: `upgrades/dbpatches/1.6/mysql/patch.sql`

Oracle: `upgrades/dbpatches/1.6/oracle/patch.sql`

PostgreSQL: `upgrades/dbpatches/1.6/postgresql/patch.sql`

注: データベースのアップグレードには、数時間以上かかる可能性があります。
非本番環境でアップグレード手順をテストすることをお奨めします。

十分なパーミッション(`create table`、`drop table`、`create index`、`drop index`)があることを確認してください。また、ディスクに十分な空き容量があることを確認してください。

注: 上記のスクリプトは、ZABBIX 1.4.xを1.6にアップグレードする場合にのみ使用できます。

1.7.4.7. 新しい ZABBIX GUI のインストール

指示されるインストール手順に従って実行します。

1.7.4.8. 新しい ZABBIX バイナリの起動

新しいバイナリを起動します。ログファイルをチェックして、バイナリが正常に起動されていることを確認します。

1.8. 有料サポート

ZABBIX SIA は、顧客固有のニーズに応えられるように、あらゆるタイプのサポートを提供します。

ZABBIX サポートサービスは、専門知識を持つサポートエンジニアが、ZABBIX の開発、展開、および管理を行っているお客様と直接対話しながらサポートします。

詳細については、<http://www.zabbix.com/services.php> を参照するか、または sales@zabbix.com までメールでお問い合わせください。

2. インストール

2.1. ZABBIX の入手方法

最新バージョンの情報およびダウンロード手順については、ZABBIX ホームページ (<http://www.zabbix.com>)を参照してください。

2.2. 要件

2.2.1. ハードウェア要件

2.2.1.1. メモリ要件

ZABBIX には、十分な量の物理メモリとディスク容量が必要です。導入当初は、物理メモリ 128MB とディスク空き容量 256MB があれば十分です。ただし、必要なディスク容量は、監視するホストとパラメータの数によって異なります。監視パラメータの履歴を長期間保存することを計画している場合、データベースに履歴を保存するための十分な容量として少なくとも 2GB を確保することを検討する必要があります。

ZABBIX デモンプロセスは、それぞれがデータベースサーバと複数の接続を確立して動作します。接続に割り当てられるメモリ容量は、データベースエンジンの設定によって異なります。

注: 物理メモリの容量を増やすほど、データベースは高速に動作し、その結果ZABBIXも高速に動作します。

2.2.1.2. CPU 要件

ZABBIX、特に ZABBIX データベースは、監視パラメータの数と使用するデータベースエンジンによって程度は異なりますが、CPU リソースを大量に消費します。

2.2.1.3. その他のハードウェア

ZABBIX の SMS 通知機能を使用する場合、シリアルポートとシリアル GSM モデムが必要です。

2.2.1.4. ハードウェア設定の例

以下の表に、様々なハードウェア設定の例を示します。

名前	プラットフォーム	CPU/メモリ	データベース	監視するホスト
小規模	Ubuntu Linux	PII 350MHz 256MB	MySQL MyISAM	20
中規模	Ubuntu Linux 64 ビット	AMD Athlon 3200+2GB	MySQL InnoDB	500
大規模	Ubuntu Linux 64 ビット	Intel Dual Core 64004GBRAID10	MySQL InnoDB または PostgreSQL	>1000
非常に大規模	RedHat Enterprise	Intel Xeon 2 × CPU 8GB Fast RAID10	MySQL InnoDB または PostgreSQL	>10000

注: 実際の設定は、アクティブなアイテムの数と更新速度によって大きく異なります。大規模システムでは、データベースを専用のマシンで動作させることを強くお勧めします。

2.2.2. サポートされるプラットフォーム

監視サーバのセキュリティ要件とミッションクリティカルな性質を考慮した場合、必要とされるパフォーマンス、フォールトトレランス、および復旧のしやすさを安定して実現できるオペレーティングシステムは UNIX だけです。ZABBIX は、市販されている主なバージョンの UNIX で動作します。

以下のプラットフォームで動作確認済みです。

- AIX
- FreeBSD
- HP-UX
- Linux
- Mac OS/X
- NetBSD
- OpenBSD
- SCO Open Server
- Solaris
- Windows 2000、2003、XP、Vista(ZABBIX エージェントのみ)

注: 上記以外のUNIXライクなオペレーティングシステムでも同様に動作可能です。

2.2.3. ソフトウェア要件

ZABBIX は、最新の Apache ウェブサーバ、主なデータベースエンジン、および PHP スクリプト言語を組み合わせて動作します。

ZABBIX が動作するには、以下のソフトウェアが必要です。

ソフトウェア	バージョン	備考
Apache	1.3.12 以降	
PHP	4.3 以降	
PHPモジュール: php-gd php-bcmath	4.3以降	PHP GDモジュールは、PNGイメージ対応である必要があります。
MySQL php-mysql	3.22以降	MySQLをZABBIXのバックエンドデータベースとして使用する場合に必要です。
Oracle php-sqlora8	9.2.0.4以降	OracleをZABBIXのバックエンドデータベースとして使用する場合に必要です。
PostgreSQL php-pgsql	7.0.2以降	PostgreSQLをZABBIXのバックエンドデータベースとして使用する場合に必要です。 パフォーマンスを高める必要がある場合は PostgreSQL 8.x以降の使用を検討してください。
SQLite php-sqlite3	3.3.5以降	SQLiteをZABBIXのバックエンドデータベースとして使用する場合に必要です。

注: ZABBIXは、Apache、MySQL、Oracle、およびPostgreSQLの古いバージョンでも動作可能です。

クライアントサイドのウェブブラウザ

HTML と PNG イメージをサポートする必要があります。Internet Explorer(5.xx と 6.xx)および Mozilla 1.x は完全に動作します。設定で Cookie と JavaScript を有効にする必要があります。他のブラウザも同様に使用可能です。

2.2.4. データベースエンジンの選択

ZABBIX サーバおよびプロキシでは、以下の4つのデータベースエンジンがサポートされています。

- MySQL
- Oracle
- PostgreSQL
- SQLite

どのデータベースエンジンにも独自の長所があり、順位を付けてお奨めすることはできません。以下の点について検討して、状況に適したデータベースエンジンを選択してください。

- ハードウェアの性能
- 無料のデータベースエンジンを使用するか、市販のデータベースエンジンを使用するか
- ZABBIX サーバまたはプロキシの負荷

以下の表に、それぞれの状況に応じてお奨めするデータベースエンジンを示します。

推奨データベースエンジン	用途
MySQL InnoDB	高負荷ノード/スタンドアロンサーバ 高負荷プロキシ
MySQL MyISAM	低負荷ノード/スタンドアロン 低負荷プロキシ
PostgreSQL	高負荷ノード/スタンドアロンサーバ 高負荷プロキシ
Oracle	高負荷ノード/スタンドアロンサーバ
SQLite	低負荷プロキシ

2.2.5. データベースサイズ

ZABBIX 設定データを保存するために一定量のディスク容量が必要ですが、ほとんど増えることはありません。

ZABBIX データベースサイズは、主に格納された履歴データの量に関する以下の数量によって決まります。

- 1秒あたりに処理される値の数

ZABBIX サーバが1秒あたりに受信する新しい値の平均個数です。

たとえば、3000 個のアイテムを更新速度 60 秒で監視する場合、1 秒あたりの値の数は $3000/60=50$ 個になります。

これは、毎秒 50 個の新しい値が ZABBIX データベースに追加されることを意味します。

■ ヒストリデータ用の housekeeper 設定

収集した値は、一定期間(通常は数週間から数ヶ月)だけ保存されます。新しい値を追加するたびに、データおよびインデックス用に一定量のディスク容量が消費されます。

そのため、1 秒あたり 50 個の値を受信して、そのヒストリを 30 日間保存する場合、値の総数は $(30 \times 24 \times 3600) \times 50=129,600,000$ 、すなわち約 1.3 億個になります。

使用するデータベースエンジンと受信する値のデータ型(浮動小数点、整数、文字列、ログファイルなど)によって異なりますが、1 つの値を保存するために必要なディスク容量は 40 バイトから数百バイトまで様々です。通常は、1 つの値に約 50 バイトが必要です。

上記の例で考えると、1.3 億個の値を保存するために、 $1.3 \text{ 億} \times 50 \text{ バイト}=6.5\text{GB}$ のディスク容量が必要になります。

■ トレンドデータ用の housekeeper 設定

テーブルトレンドの各アイテムについて、1 時間の最大/最小/平均/個数の統計情報が保存されます。保存されたデータは、トレンドグラフや長期間グラフの表示に使用されます。

データベースのタイプにもよりますが、ZABBIX データベースに上記の統計情報を 1 回分保存するために必要なディスク容量は 128 バイトです。

たとえば、3000 個の値のトレンドデータを 5 年分保存しようとする、1 年分で $(3000/1800) \times (24 \times 3600 \times 365) \times 128=6.3\text{GB}$ なので、5 年分では **31.5GB** のディスク容量が必要になります。

■ イベントデータ用の housekeeper 設定

1 個のイベントにつき約 130 バイトのディスク容量が必要です。ZABBIX で毎日発生するイベントの数を算出することは非常に困難です。最悪のケースとして、1 秒に 1 個のイベントが発生すると仮定します。

イベントデータを 3 年分保存する場合、 $3 \times 365 \times 24 \times 3600 \times 130=11\text{GB}$ のディスク容量が必要になります。

以下の表に、ZABBIX システムに必要なディスク容量の算出に役立つ式を示します。

パラメータ	必要なディスク容量を算出するための式(バイト)
ZABBIX設定	一定サイズ。通常は10MB以下です。
ヒストリ	$\text{days} \times (\text{items}/\text{refresh rate}) \times 24 \times 3600 \times \text{bytes}$ items: アイテム数 days: ヒストリを保存する日数 refresh rate: アイテムの平均更新速度 bytes: 1個の値を保存するために必要なバイト数。データベースエンジンにもよりますが、通常は50バイトです。
トレンド	$\text{days} \times (\text{items}/1800) \times 24 \times 3600 \times \text{bytes}$ items: アイテム数 days: トレンド履歴を保存する日数 bytes: 1個のトレンドを保存するために必要なバイト数。データベースエンジンにもよりますが、通常は128バイトです。

パラメータ	必要なディスク容量を算出するための式(バイト)
イベント	$days \times events \times 24 \times 3600 \times bytes$ events :1秒あたりのイベント個数。 最悪のケースでは1秒に1個のイベントが発生すると想定しています。 days : イベント履歴を保存する日数 bytes : 1個のトレンドを保存するために必要なバイト数。データベースエンジンにもよりますが、通常は130バイトです。

以上より、必要な総ディスク容量は、以下の式で計算できます。

設定+ヒストリ+トレンド+イベント

ここで算出したディスク容量は、ZABBIX をインストールした直後から使用されるわけではありません。

データベースサイズは最初は増え続けますが、ある時点で一定の値を維持するようになります。どの時点で一定になるかは、housekeeper の設定によって異なります。

注: 分散設定のノードで必要なディスク容量も同様の方法で計算できますが、1つのノードにリンクされている子ノードの総数にも依存します。

2.2.6. 時刻の同期

ZABBIX が動作するサーバでは、正確なシステム日時を維持することが非常に重要です。

timed はよく知られているデーモンの1つで、ホストの時刻を他のマシンの時刻と同期します。

2.3. コンポーネント

2.3.1. ZABBIX コンポーネント

ZABBIX は、複数の有名なソフトウェアコンポーネントから構成されています。各コンポーネントの役割を以下に示します。

2.3.2. ZABBIX サーバ

ZABBIX ソフトウェアの中核です。簡単なサービスチェック機能を使用して、ネットワーク経由で利用するサービス(ウェブサーバやメールサーバなど)をリモートからチェックしたり、エージェントから可用性や整合性に関する情報や統計情報を受信します。設定データ、統計データ、および運用データは全てサーバに保存されます。監視対象システムで問題が発生したときに、能動的に管理者にアラートを送信するのもサーバの役割です。

ZABBIX は、エージェントレスで監視したり、SNMP エージェントでネットワークデバイスを監視することもできます。

2.3.3. ZABBIX プロキシ

プロキシは、ZABBIX の展開でオプションとして使用されます。プロキシでは、ZABBIX サーバの代わりにパフォーマンスおよび可用性データが収集されます。収集された全てのデータがローカルのバッファに格納され、プロキシが属する ZABBIX サーバに転送されます。

ZABBIX プロキシは、ローカル管理者のいないリモートの場所、支店、ネットワークを集中監視するための理想的なソリューションとなります。

また、ZABBIX プロキシを使用して、単一の ZABBIX サーバの負荷を分散することもできます。この場合、プロキシのみでデータが収集されるため、サーバの処理に対する CPU とディスク I/O の使用量が軽減されます。

2.3.4. ZABBIX エージェント

ネットワークに接続されたシステムのローカルのリソースやアプリケーション(ハードドライブ、メモリ、プロセッサの統計情報など)を能動的に監視するには、そのシステムで ZABBIX エージェントを実行する必要があります。エージェントは、動作するシステムの稼動情報を収集し、ZABBIX サーバで処理するためにそれらのデータを送信します。障害(ハードディスク満杯、サービスプロセスのクラッシュなど)が発生した場合、ZABBIX サーバは、障害が発生したマシンに関するアラートを能動的に管理者に送信します。

ZABBIX エージェントは統計情報を収集する際にネイティブのシステムコールを使用するので、非常に効率的に動作します。

2.3.5. ウェブインタフェース

監視データや ZABBIX の設定データに、プラットフォームを選ばずにどこからでも簡単にアクセスできるようにするために、ウェブインタフェースが用意されています。このインタフェースは ZABBIX サーバに組み込まれており、通常は ZABBIX サーバが動作するのと同じ物理マシン上で動作しますが、別マシン上で動作させることもできます。

注: SQLiteを使用する場合は、SQLiteが動作するのと同じ物理マシン上でZABBIXフロントエンドを実行する必要があります。

2.4. ソースからのインストール

2.4.1. ソフトウェア要件

ここでは、ソースから ZABBIX サーバまたはエージェントを作成するために必要なソフトウェアについて説明します。

ZABBIX をコンパイルするために必要なソフトウェアを以下に示します。

以下のデータベースエンジンのうちいずれか 1 つ:

MySQL のヘッダとライブラリ

バージョン 3.22 以降が必要です。

Oracle のヘッダとライブラリ

Sqlora8 のヘッダとライブラリが必要です。

PostgreSQL のヘッダとライブラリ

バージョン 7.0.2 以降が必要です。パフォーマンスを高める必要がある場合は PostgreSQL 8.x の使用を検討してください。

SQLite のヘッダとライブラリ

バージョン 3.3.5 以降が必要です。

注: 通常は、mysql-dev、postgresql-dev、sqlite3-devの各パッケージの一部として提供されます。

NET-SNMP(または UCD-SNMP)のライブラリファイルとヘッダファイル

SNMP をサポートする場合に必要です。オプション。

Iksemel のライブラリファイルとヘッダファイル

Jabber メッセージングをサポートする場合に必要です。オプション。

Libcurl のライブラリファイルとヘッダファイル

ウェブ監視モジュールでバージョン 7.13.1 以降が必要です。オプション。

C コンパイラ

C コンパイラが必要です。オープンプラットフォームでは、GNU C コンパイラが最適です。

それ以外の C コンパイラ(HP や IBM 製)も使用できます。

GNU Make

ZABBIX の Makefile を処理するために GNU Make が必要です。

2.4.2. ZABBIX の構造

docs

本書が PDF 形式で格納されています。

src

フロントエンドを除く全ての ZABBIX プロセスのソースが格納されています。

src/zabbix_server

zabbix_server の Makefile とソースが格納されています。

src/zabbix_agent

zabbix_agent と zabbix_agentd の Makefile とソースが格納されています。

src/zabbix_get

zabbix_get の Makefile とソースが格納されています。

src/zabbix_sender

zabbix_sender の Makefile とソースが格納されています。

include

ZABBIX のインクルードファイルが格納されています。

misc

misc/init.d

様々なプラットフォーム用のスタートアップスクリプトが格納されています。

frontends

frontends/php

PHP フロントエンドのファイルが格納されています。

create

データベースの初期作成用の SQL スクリプトが格納されています。

create/schema

データベース作成スキーマが格納されています。

create/data

データベース初期作成用データが格納されています。

upgrades

ZABBIX の様々なバージョン用のアップグレード手順が格納されています。

2.4.3. ZABBIX サーバ

サーバサイド

ステップ 1 ZABBIX スーパーユーザアカウントの作成

サーバを実行するユーザを作成します。本番環境用に、特権を持たない専用アカウント(通常は zabbix)を作成する必要があります。セキュリティ上のリスクが生じるので、root や bin など、特権を持つアカウントでは、決して ZABBIX を実行しないでください。

注: ZABBIXサーバプロセス(zabbix_server)は、rootアカウントで実行できないように保護されています。

ステップ 2 ZABBIX ソースの展開

```
shell> gunzip zabbix-1.6.tar.gz && tar -xvf zabbix-1.6.tar
```

ステップ 3 ZABBIX データベースの作成

ZABBIX には、必要なデータベーススキーマを作成し、デフォルト設定データを挿入するための SQL スクリプトが付属しています。スクリプトは、MySQL、Oracle、PostgreSQL、SQLite 用に別々に用意されています。

MySQL の場合

```
shell> mysql -u<username> -p<password>
mysql> create database zabbix;
mysql> quit;
shell> cd create/schema
shell> cat mysql.sql | mysql -u<username> -p<password> zabbix
shell> cd ../data
shell> cat data.sql | mysql -u<username> -p<password> zabbix
shell> cat images_mysql.sql | mysql -u<username> -p<password> zabbix
```

Oracle の場合(ユーザ zabbix がパスワード password で登録され、データベースオブジェクト作成権限が与えられているものとします)

```
shell> cd create
shell> sqlplus zabbix/password
sqlplus> set def off
sqlplus> @schema/oracle.sql
sqlplus> @data/data.sql
sqlplus> @data/images_oracle.sql
sqlplus> exit
```

PostgreSQL の場合

```
shell> psql -U <username>
```

```
psql> create database zabbix;
psql> ¥q
shell> cd create/schema
shell> cat postgresql.sql | psql -U <username> zabbix
shell> cd ../data
shell> cat data.sql | psql -U <username> zabbix
shell> cat images_pgsql.sql | psql -U <username> zabbix
```

SQLite の場合

```
shell> cd create/schema
shell> cat sqlite.sql | sqlite3 /var/lib/sqlite/zabbix.db
shell> cd ../data
shell> cat data.sql | sqlite3 /var/lib/sqlite/zabbix.db
shell> cat images_sqlite3.sql | sqlite3 /var/lib/sqlite/zabbix.db
```

注: データベースが存在しない場合は、自動的に作成されます。

ステップ 4 使用するシステムに対応するソースコードの設定とコンパイル

サーバ(マシンを監視する)とクライアント(監視されるマシン)は、どちらもソースをコンパイルして作成する必要があります。サーバのソースを設定するために、使用するデータベースを指定する必要があります。

```
shell> ./configure --enable-server --with-mysql --with-net-snmp --with-jabber --with-libcurl # for MySQL + Jabber + WEB monitoring
```

または

```
shell> ./configure --enable-server --with-pgsql --with-net-snmp --with-jabber --with-libcurl # for PostgreSQL + Jabber + WEB monitoring
```

または

```
shell> ./configure --enable-server --with-oracle=/home/zabbix/sqlora8 --with-netsnmp --with-jabber --with-libcurl # for Oracle + Jabber + WEB monitoring
```

注: sqlora8のライブラリの位置を指定する場合はフラグ--with-oracleを使用します。Oracleをサポートする場合はこのライブラリが必要であり、libsqlora8のホームページから入手できます。

注: ライブラリをスタティックリンクする場合はフラグ--enable-staticを使用します。コンパイルしたバイナリを複数のサーバに分散させる場合は、分散先のサーバに必要なライブラリが存在しなくてもバイナリが動作するように、このフラグを使用する必要があります。Solarisでは--enable-staticは使用できません。--with-net-snmpの代わりにフラグ--with-ucd-snmpを使用することができます。SNMPをサポートする必要がない場合は、--with-net-snmpと--with-ucd-snmpはどちらも指定する必要はありません。

クライアントバイナリとサーババイナリを両方ともコンパイルする場合は、以下のコマンドを実行します。

```
shell> ./configure --enable-server --enable-agent --with-mysql --with-net-snmp --with-jabber --with-libcurl
```

パラメータ--enable-static を使用すると、強制的にスタティックリンクすることができます。

ステップ 5 Make とインストール

```
shell> make install
```

デフォルトでは、

```
make install
```

を実行すると、全てのファイルが/usr/local/bin、/usr/local/lib などにインストールされます。--prefix を使用すると、/usr/local 以外のインストールプリフィクスを指定できます。

ステップ 6 /etc/services の設定

このステップは必ずしも必要ではありませんが、実行することをお奨めします。クライアント(監視される)マシンで、以下の行を/etc/services に追加します。

```
zabbix-agent 10050/tcp Zabbix Agent
zabbix-agent 10050/udp Zabbix Agent
zabbix-trapper 10051/tcp Zabbix Trapper
zabbix-trapper 10051/udp Zabbix Trapper
```

ステップ 7 /etc/inetd.conf の設定

推奨されている zabbix_agentd ではなく、zabbix_agent を使用する場合は、以下の行を追加する必要があります。

```
zabbix_agent stream tcp nowait.3600 zabbix /opt/zabbix/bin/zabbix_agent
```

inetd をリスタートします。

```
shell> killall -HUP inetd
```

設定ファイルのデフォルト設定を変更します。

ステップ 8 /etc/zabbix/zabbix_agent.conf の設定

zabbix_agent をインストールする全てのホストでこのファイルを設定する必要があります。このファイルには、ZABBIX サーバの IP アドレスを指定する必要があります。他のホストからの接続は拒否されます。サンプルとして misc/conf/zabbix_agent.conf が提供されているので、必要に応じて参照してください。

ステップ 9 /etc/zabbix/zabbix_agentd.conf の設定

zabbix_agentd をインストールする全てのホストでこのファイルを設定する必要があります。このファイルには、ZABBIX サーバの IP アドレスを指定する必要があります。他のホストからの接続は拒否されます。サンプルとして misc/conf/zabbix_agentd.conf が提供されているので、必要に応じて参照してください。

ステップ 10 /etc/zabbix/zabbix_server.conf の設定

小規模な環境(監視するホスト数が 10 以下)にインストールする場合、デフォルトパラメータを修正する必要はありません。ただし、ZABBIX のパフォーマンスを最大限に高めるには、デフォルトパラメータを変更する必要があります。詳細については、「パフォーマンスチューニング」を参照してください。

サンプルとして misc/conf/zabbix_server.conf が提供されているので、必要に応じて参照してください。

ステップ 11 サーバプロセスの実行

サーバサイドで zabbix_server を実行します。

```
shell> cd bin
shell> ./zabbix_server
```

ステップ 12 エージェントの実行

必要なマシン上で zabbix_agentd を実行します。

```
shell> cd bin
shell> ./zabbix_agentd
```

2.4.4. ZABBIX プロキシ

ZABBIX プロキシは特別なプロセスです。プロセスを実行する必要はありません。

ステップ 1 ZABBIX スーパーユーザアカウントの作成

プロキシを実行するユーザを作成します。本番環境用に、特権を持たない専用アカウント(通常は

zabbix)を作成する必要があります。セキュリティ上のリスクが生じるので、root や bin など、特権を持つアカウントでは、決して ZABBIX プロキシを実行しないでください。

注: ZABBIXプロキシプロセス(zabbix_proxy)は、rootアカウントで実行できないように保護されています。

ステップ 2 ZABBIX ソースの展開

```
shell> gunzip zabbix-1.6.tar.gz && tar -xvf zabbix-1.6.tar
```

ステップ 3 ZABBIX データベースの作成(オプション)

注: データベースが存在しない場合は、ZABBIXプロキシプロセスの初回実行時に自動的に作成されます。それ以外の場合は、既存のデータベースが使用されます。データベースの自動作成がサポートされているのはSQLiteのみです。

ZABBIX には、必要なデータベーススキーマを作成するための SQL スクリプトが付属しています。スクリプトは、MySQL、Oracle、PostgreSQL、SQLite 用に別々に用意されています。

MySQL の場合

```
shell> mysql -u<username> -p<password>
mysql> create database zabbix;
mysql> quit;
shell> cd create/schema
shell> cat mysql.sql | mysql -u<username> -p<password> zabbix
shell> cd ../data
shell> cat data.sql | mysql -u<username> -p<password> zabbix
shell> cat images_mysql.sql | mysql -u<username> -p<password> zabbix
```

Oracle の場合(ユーザ zabbix がパスワード password で登録され、データベースオブジェクト作成権限が与えられているものとします)

```
shell> cd create/schema
shell> cat oracle.sql | sqlplus zabbix/password >out.log
```

注: エラーメッセージが出力されていないか、out.logをチェックしてください。

```
shell> cd ../data
shell> cat data.sql | sqlplus zabbix/password >out.log
shell> cat images_oracle.sql | sqlplus zabbix/password >>out.log
```

PostgreSQL の場合

```
shell> psql -U <username>
psql> create database zabbix;
```

```
psql> ¥q
shell> cd create/schema
shell> cat postgresql.sql | psql -U <username> zabbix
shell> cd ../data

shell> cat data.sql | psql -U <username> zabbix
shell> cat images_pgsql.sql | psql -U <username> zabbix
```

For SQLite:

```
shell> cd create/schema
shell> cat sqlite.sql | sqlite3 /var/lib/sqlite/zabbix.db
shell> cd ../data
shell> cat data.sql | sqlite3 /var/lib/sqlite/zabbix.db
shell> cat images_sqlite3.sql | sqlite3 /var/lib/sqlite/zabbix.db
```

注: データベースが存在しない場合は、自動的に作成されます。

ステップ 4 使用するシステムに対応するソースコードの設定とコンパイル

ソースをコンパイルして、ZABBIX プロキシプロセスのコンパイルを有効にする必要があります。プロキシのソースを設定するために、使用するデータベースを指定する必要があります。

```
shell> ./configure --enable-proxy --with-mysql --with-net-snmp -with-libcurl # for MySQL + WEB monitoring
```

または

```
shell> ./configure --enable-proxy --with-pgsql --with-net-snmp -with-libcurl # for PostgreSQL + WEB monitoring
```

または

```
shell> ./configure --enable-proxy --with-oracle=/home/zabbix/sqlora8 --with-netsnmp -with-libcurl # for Oracle + WEB monitoring
```

注: sqlora8のライブラリの位置を指定する場合はフラグ--with-oracleを使用します。Oracleをサポートする場合はこのライブラリが必要であり、libsqlora8のホームページから入手できます。

注: ライブラリをスタティックリンクする場合はフラグ--enable-staticを使用します。コンパイルしたバイナリを複数のホストに分散させる場合は、分散先のホストに必要なライブラリが存在しなくてもバイナリが動作するように、このフラグを使用する必要があります。Solarisでは--enable-staticは使用できません。--with-net-snmpの代わりにフラグ--with-ucd-snmpを使用することができます。SNMPをサポートする必要がない場合は、--with-net-snmpと--with-ucd-snmpはどちらも指定する必要はありません。

クライアントバイナリとプロキシバイナリを両方ともコンパイルする場合は、以下のコマンドを実行します。

```
shell> ./configure --enable-proxy --enable-agent --with-mysql -with-net-snmp --with-libcurl
```

パラメータ`--enable-static`を使用すると、強制的にスタティックリンクすることができます。

ステップ 5 Make とインストール

```
shell> make install
```

デフォルトでは、

```
make install
```

を実行すると、全てのファイルが`/usr/local/bin`、`/usr/local/lib`などにインストールされます。`--prefix`を使用すると、`/usr/local`以外のインストールプリフィクスを指定できます。

ステップ 6 /etc/services の設定

このステップは必ずしも必要ではありませんが、実行することをお奨めします。クライアント(監視される)マシンで、以下の行を`/etc/services`に追加します。

```
zabbix_agent 10050/tcp
```

```
zabbix_trap 10051/tcp
```

ステップ 7 /etc/inetd.conf の設定

推奨されている `zabbix_agentd` ではなく、`zabbix_agent` を使用する場合は、以下の行を追加する必要があります。

```
zabbix_agent stream tcp nowait.3600 zabbix /opt/zabbix/bin/zabbix_agent
```

`inetd` をリスタートします。

```
shell> killall -HUP inetd
```

設定ファイルのデフォルト設定を変更します。

ステップ 8 /etc/zabbix/zabbix_proxy.conf の設定

小規模な環境(監視するホスト数が 10 以下)にインストールする場合、デフォルトパラメータを修正する必要はありません。ただし、ZABBIX プロキシのパフォーマンスを最大限に高めるには、デフォルトパラメータを変更する必要があります。

ホスト名とサーバのパラメータが正しく設定されていることを確認してください。

サンプルとして `misc/conf/zabbix_proxy.conf` が提供されているので、必要に応じて参照してください。

ステップ 9 プロキシプロセスの実行

zabbix_proxy を実行します。

```
shell> cd sbin
shell> ./zabbix_proxy
```

2.4.5. ZABBIX エージェント

クライアントサイド

ステップ 1 ZABBIX アカウントの作成

エージェントを実行するユーザを作成します。本番環境用に、特権を持たない専用アカウント(通常は zabbix)を作成する必要があります。ZABBIX エージェントは、root アカウントで実行できないように保護されています。

ステップ 2 ZABBIX ソースの展開

```
shell> gunzip zabbix-1.6.tar.gz && tar xvf zabbix-1.6.tar
```

ステップ 3 使用するシステムに対応するソースコードの設定とコンパイル

クライアント用のソースだけをコンパイルする必要があります。

クライアントのソースを設定するには、以下のコマンドを実行します。

```
shell> ./configure --enable-agent
```

注: ライブラリをスタティックリンクする場合はフラグ--enable-staticを使用します。コンパイルしたバイナリを複数のホストに分散させる場合は、分散先のホストに必要なライブラリが存在しなくてもバイナリが動作するように、このフラグを使用する必要があります。

ステップ 4 エージェントのビルド

```
shell> make
```

bin/から/opt/zabbix/bin、もしくは/usr/local/bin や/usr/local/zabbix/bin など他の一般的なディレクトリにコピーします。

ステップ 5 /etc/services の設定

このステップは必ずしも必要ではありませんが、実行することをお奨めします。

クライアント(監視される)マシンで、以下の行を/etc/services に追加します。

```
zabbix_agent 10050/tcp
```

```
zabbix_trap 10051/tcp
```

ステップ 6 /etc/inetd.conf の設定

推奨されている zabbix_agentd ではなく、zabbix_agent を使用する場合は、以下の行を追加する必要があります。

```
zabbix_agent stream tcp nowait.3600 zabbix /opt/zabbix/bin/zabbix_agent
```

inetd をリスタートします。

```
shell> killall -HUP inetd
```

ステップ 7 /etc/zabbix/zabbix_agent.conf の設定

zabbix_agent をインストールする全てのホストでこのファイルを設定する必要があります。このファイルには、ZABBIX サーバの IP アドレスを指定する必要があります。他のホストからの接続は拒否されます。このファイルで、EOL 文字は使用しないでください。

サンプルとして misc/conf/zabbix_agent.conf が提供されているので、必要に応じて参照してください。

ステップ 8 /etc/zabbix/zabbix_agentd.conf の設定

zabbix_agentd をインストールする全てのホストでこのファイルを設定する必要があります。このファイルには、ZABBIX サーバの IP アドレスを指定する必要があります。他のホストからの接続は拒否されます。サンプルとして misc/conf/zabbix_agentd.conf が提供されているので、必要に応じて参照してください。

ステップ 9 監視する全てのマシン上で zabbix_agentd を実行

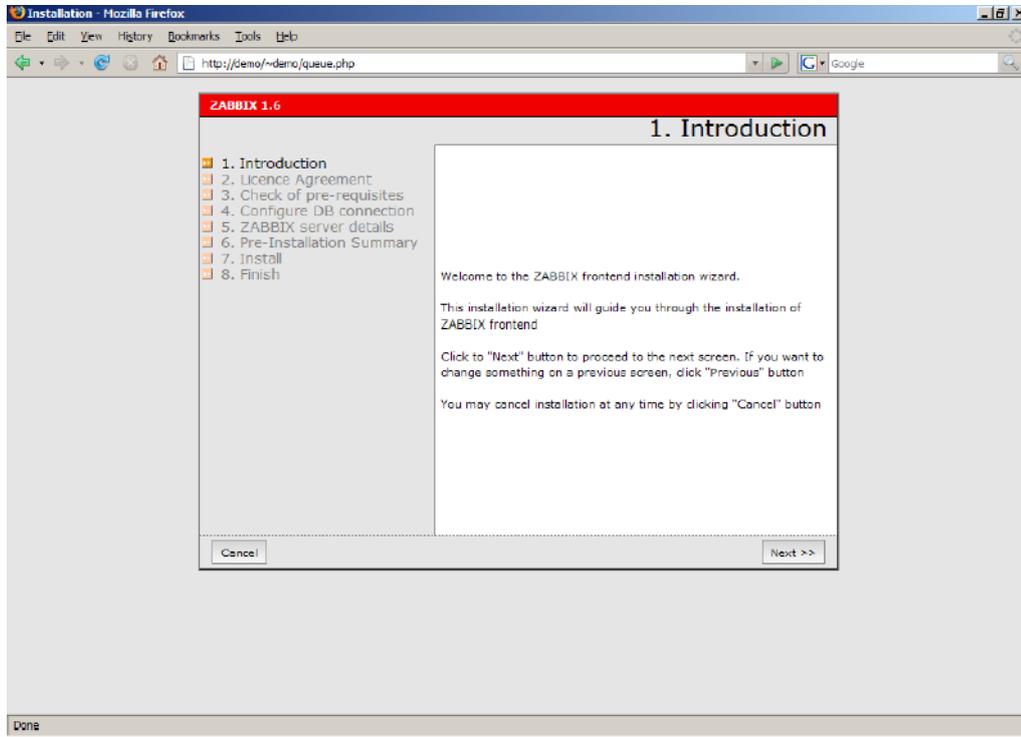
```
shell> /opt/zabbix/bin/zabbix_agentd
```

注: zabbix_agent を使用する場合は、zabbix_agentd は実行しないでください。

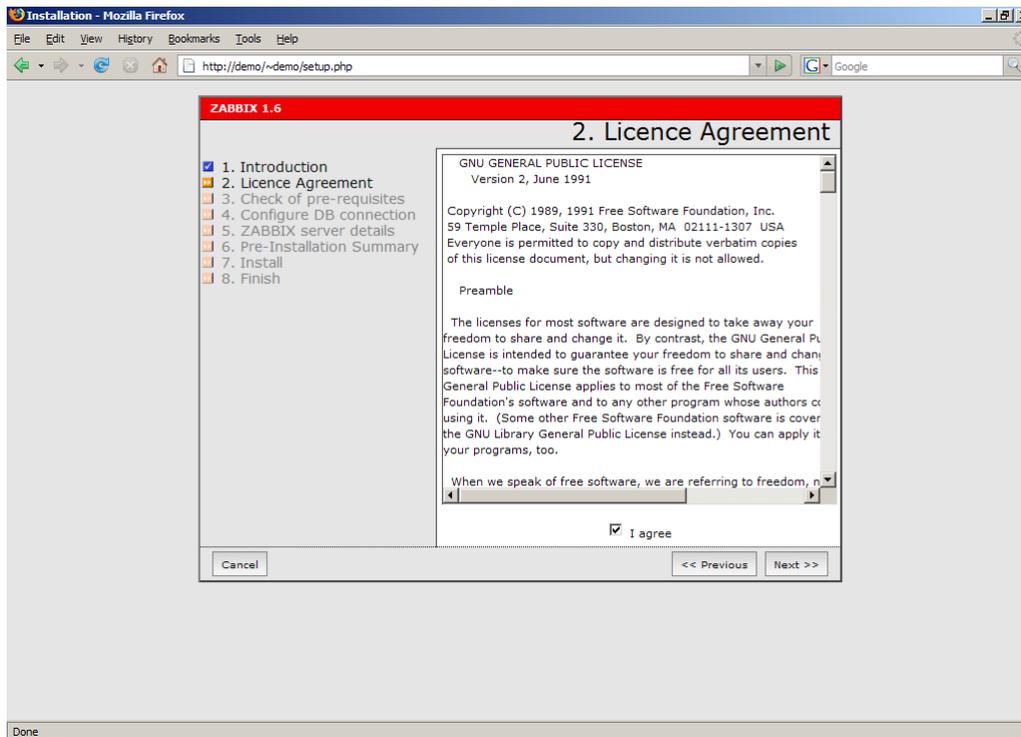
注: システムで共有メモリに2MBを割り当て可能であることを確認してください。この共有メモリを割り当てられない場合、エージェントの起動は失敗して、エージェントのログファイルに「Can't allocate shared memory for collector.」というメッセージが出力されます。この現象はSolaris 8で発生する可能性があります。

2.4.6. ZABBIX ウェブインタフェース

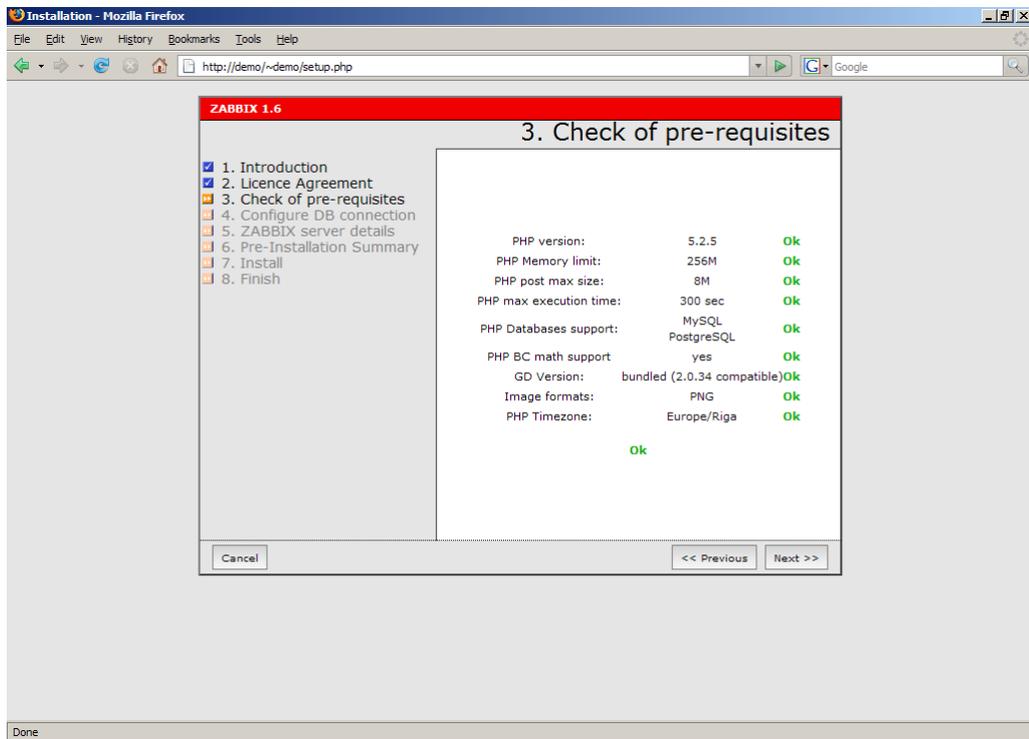
ステップ 1 ブラウザで ZABBIX URL に移動します。



ステップ 2 GPL v2 の内容を確認して合意します。



ステップ 3 ソフトウェアの前提条件を全て満たしていることを確認します。



前提条件	最小値	説明
PHPバージョン	4.3.0	
PHPメモリ制限	8MB	php.ini内での記述: memory_limit = 128M
PHP POSTの最大サイズ	8MB	php.ini内での記述: post_max_size = 8M
PHP最大実行時間	300秒	php.ini内での記述: max_execution_time = 300
PHPデータベースサポート	以下のいずれか1つ: MySQL、Oracle、PostgreSQL、SQLite	以下のモジュールのいずれか1つがインストールされている必要があります。 php-mysql php-sqlora8 php-pgsql php-sqlite3
PHP BC math	任意	PHP5でコンパイルされていること
GDバージョン	2.0以降	モジュールphp-gd
イメージ形式	少なくともPNGに対応していること	モジュールphp-gd

ステップ 4

データベースを設定します。ZABBIX データベースは作成済みである必要があります。

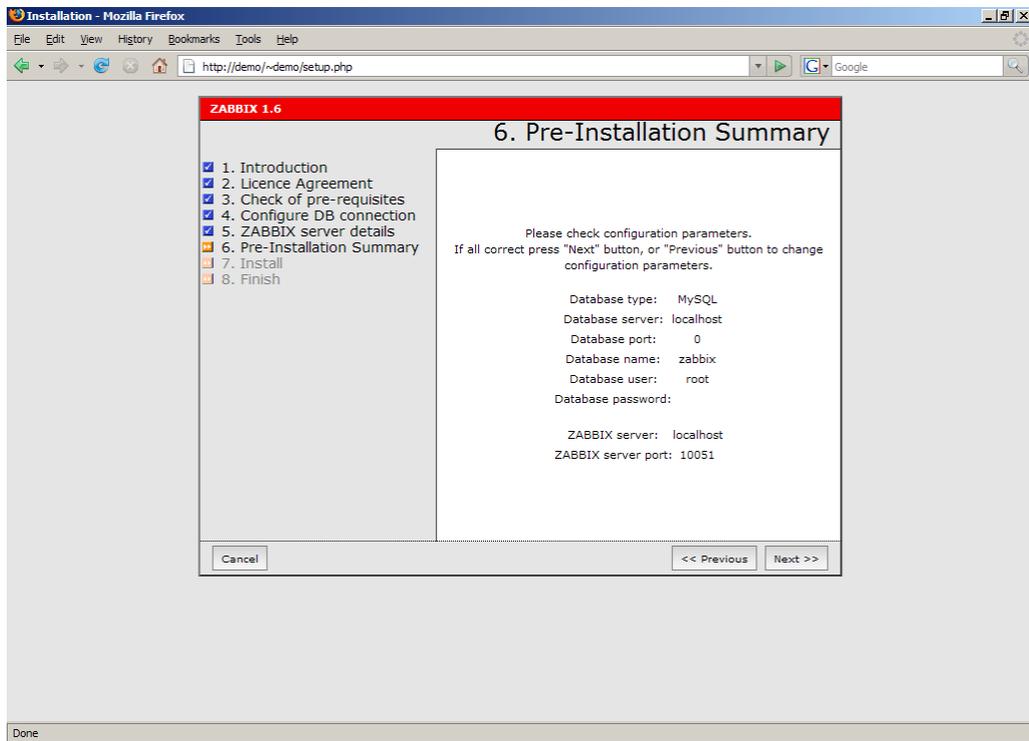
The screenshot shows the ZABBIX 1.6 installation wizard in Mozilla Firefox. The browser address bar shows `http://demo/~demo/setup.php`. The page title is "ZABBIX 1.6" and the main heading is "4. Configure DB connection". On the left, a navigation menu lists steps 1 through 8, with step 4 highlighted. The main content area contains the following text: "Please create database manually. And set the configuration parameters of connection to this database. And press 'Test connection' button." Below this text are input fields for "Type" (MySQL), "Host" (localhost), "Port" (0 - use default port), "Name" (zabbix), "User" (root), and "Password". There is an "Ok" label and a "Test connection" button. At the bottom, there are "Cancel", "<< Previous", and "Next >>" buttons.

ステップ 5

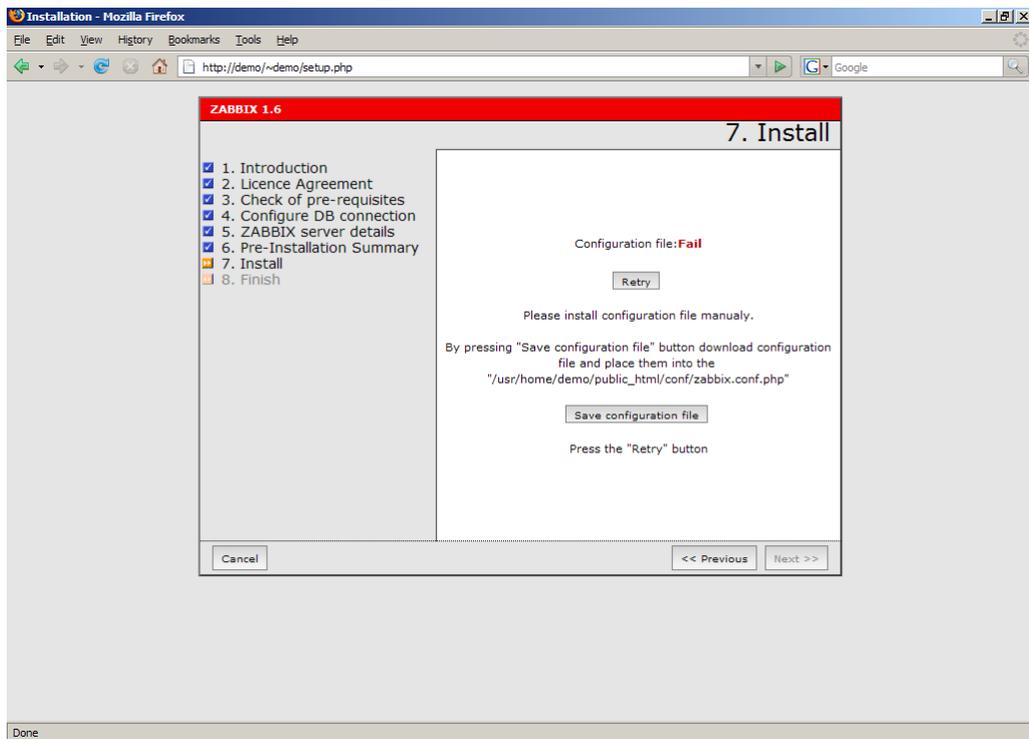
ZABBIX サーバの詳細を入力します。

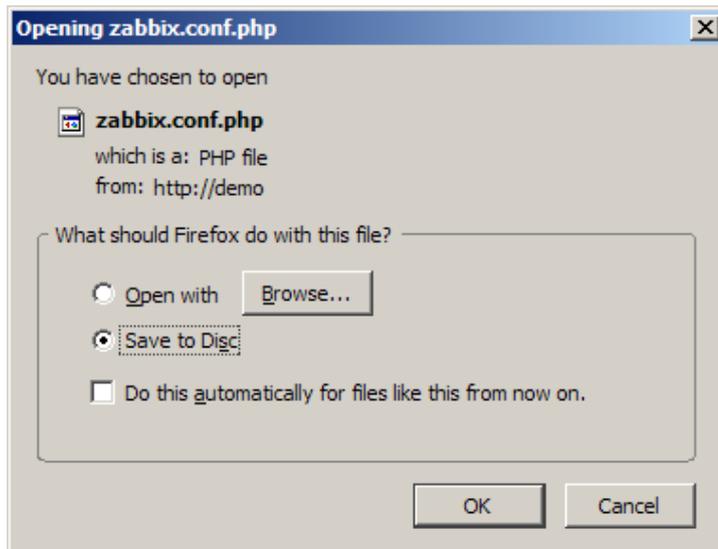
The screenshot shows the ZABBIX 1.6 installation wizard in Mozilla Firefox. The browser address bar shows `http://demo/~demo/setup.php`. The page title is "ZABBIX 1.6" and the main heading is "5. ZABBIX server details". On the left, a navigation menu lists steps 1 through 8, with step 5 highlighted. The main content area contains the following text: "Please enter host name or host IP address and port number of ZABBIX server". Below this text are input fields for "Host" (localhost) and "Port" (10051). At the bottom, there are "Cancel", "<< Previous", and "Next >>" buttons.

ステップ 6 これまでに設定した内容を確認します。

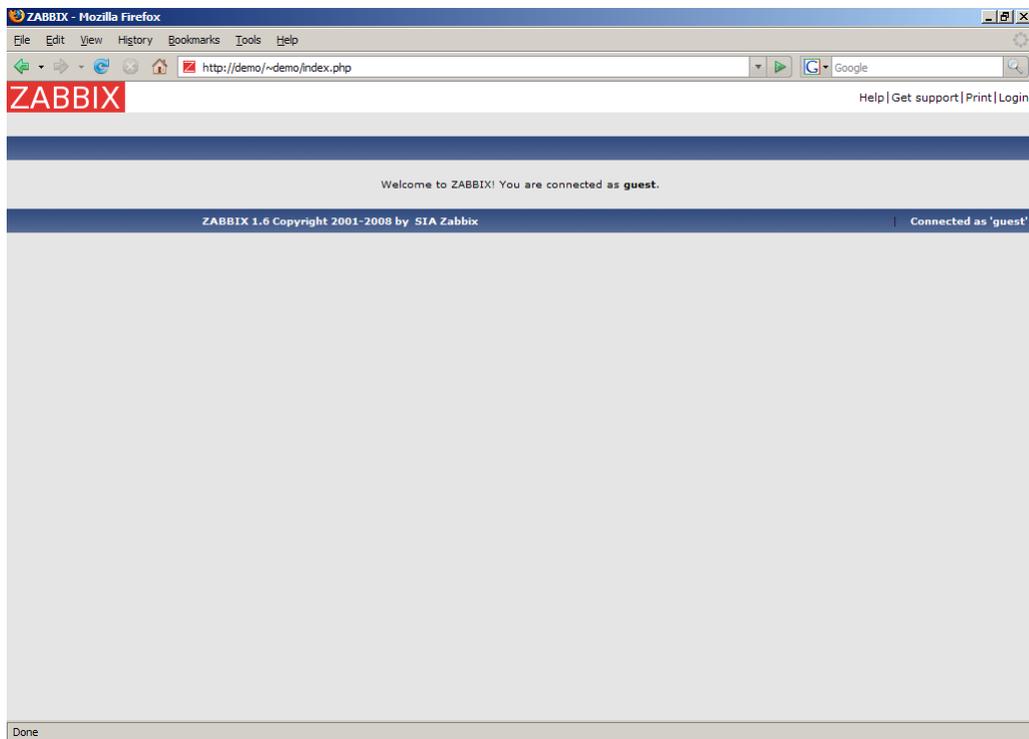
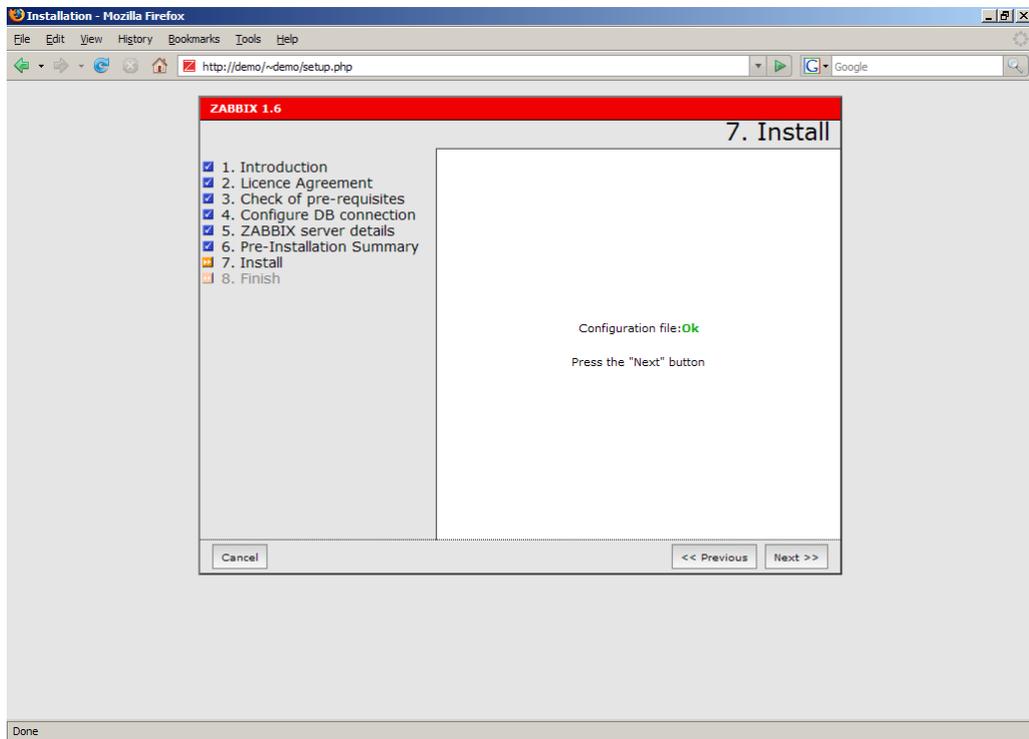


ステップ 7 設定ファイルをダウンロードして、conf/に配置します。





ステップ 8 インストール完了です。



ステップ 9 このステップは分散監視の場合にのみ実行します。

分散環境で ZABBIX を使用する場合、以下のコマンドを実行する必要があります。

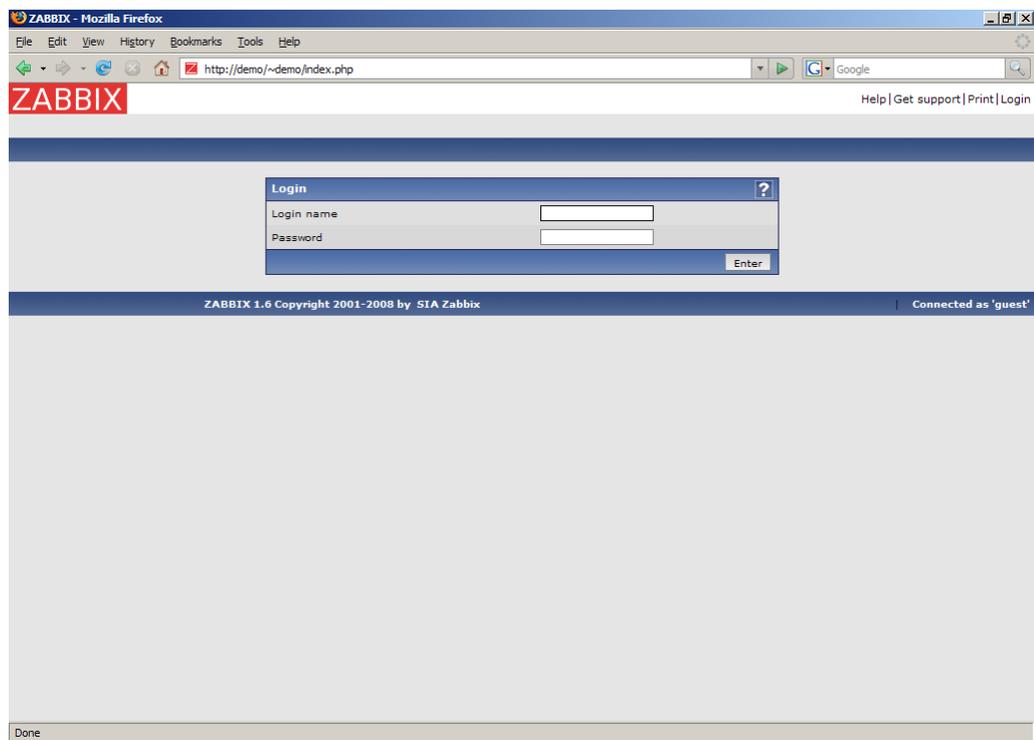
```
shell> ./zabbix_server -n <nodeid>
```

ここで、<nodeid>は一意的なノード ID です。以下に例を示します。

```
shell> ./zabbix_server -n 1
```

このコマンドは、データベースデータをノード ID「1」で使用できるように変換し、ローカルノードを追加します。

ステップ 10 ZABBIX フロントエンドの使用準備は完了しています。デフォルトのユーザ名は「Admin」、パスワードは「zabbix」です。



2.5. アップグレード

アップグレード手順は非常に簡単です。まず、新しいバイナリとフロントエンドを、最新のインストール手順に従ってインストールします。次に、以下のステップを実行してデータベース構造を更新します。

アップグレードは、瞬時に終わる場合(パッチを適用する必要がない場合)もあれば、数時間かかる場合もあります。データベースのパッチを適用する場合は、事前に全ての ZABBIX プロセスを停止する必要があります。

データベースをアップグレードする必要があるのは、安定したメジャーリリースから別のメジャーリリースにアップグレードする場合です。たとえば、1.4.x から 1.6.x へのアップグレードが該当します。

本番環境でアップグレードする場合は、必ずデータベースをバックアップしてください。

2.5.1. データベースのアップグレード

upgrades/dbpatches ディレクトリに移動します。このディレクトリには、バージョンのアップグレードを表す名前が付けられたサブディレクトリ(1.0beta3_to_1.0beta4 など)があります。実行しようとするアップグレードに対応するディレクトリに移動します。複数のバージョンを経由するアップグレードの場合は、1 段階ずつアップグレードする必要があります。使用するデータベースに応じて、以下のいずれかのコマンドを実行します。

```
shell> cd mysql; cat patch.sql |mysql zabbix -u<username> -p<password>
```

または

```
shell> cd postgresql; cat patch.sql|psql -U <username> zabbix
```

PHP フロントエンドファイルも忘れずにアップグレードしてください。

最後にバージョン固有のリリースノートを参照して、特別な手順や役に立つ情報が記述されていないか確認してください。

3. ZABBIX のプロセス

3.1. ZABBIX サーバ

ZABBIX サーバは、ZABBIX ソフトウェアの中核プロセスです。ZABBIX サーバを起動するには、以下のコマンドを実行します。

```
shell> cd bin
shell> ./zabbix_server
```

ZABBIX サーバは、デーモンプロセスとして動作します。
以下のコマンドラインパラメータを指定できます。

`-c --config <file>` 設定ファイルを指定します。デフォルトは以下のファイルです。
`/etc/zabbix/zabbix_server.conf`

`-h --help` ヘルプを表示します。

`-v --version` バージョン番号を表示します。

ヘルプを表示するには、以下のコマンドを実行します。

```
shell> zabbix_server -h
```

コマンドラインパラメータの使用例を示します。

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf
```

```
shell> zabbix_server --help
```

```
shell> zabbix_server -v
```

設定ファイルには、`zabbix_server` が使用する設定パラメータが格納されています。このファイルは必須であり、ユーザ `zabbix` にはこのファイルに対する読み取りパーミッションが必要です。以下のパラメータがサポートされています。

パラメータ	必須	デフォルト値	説明
AlertScriptsPath	×	/home/zabbix/bin	ユーザ定義メディアタイプスクリプトの保存ディレクトリ
DBHost	○	-	データベース名(通常は zabbix)
DBName	○	-	データベース名(通常は zabbix)
DBSocket	×	-	データベースソケット名。 MySQLデータベースにTCP以外の方法で接続する場合に使用します。 例: /tmp/mysql.sock
DBPassword	×	NULL	データベース接続ユーザのパスワード。パスワードを使用しない場合は、このパラメータをコメントアウトする必要があります

パラメータ	必須	デフォルト値	説明
			す。
DBUser	×	NULL	データベース接続ユーザ名
DebugLevel	×	3	デバッグレベル。以下のいずれか1つを指定します。 0 - なし 1 - クリティカル 2 - エラー 3 - 警告 4 - デバッグ
DisableHousekeeping	×	0	1に設定した場合、housekeeperが無効になります。
ExternalScripts	×	/etc/zabbix/externalscripts	外部チェックスクリプトの保存先
FpingLocation	×	/usr/sbin/fping	ICMP pingerの格納先。ICMP pingerにはsetuidフラグを設定する必要があります。
HousekeepingFrequency	×	1	housekeeperの実行間隔を時間単位で定義します。PostgreSQLを使用する場合、VACUUMコマンドが実行されるので、この値は24に設定します。
Include	×	-	設定ファイルにインクルードするファイルを指定します。 Include パラメータは何個でも指定できます。以下に例を示します。 Include=/etc/zabbix/db_conn.conf
ListenIP	×	-	trapperプロセスが待ち受けるインタフェース。設定しない場合、trapperは全インタフェースで待ち受けます。
ListenPort	×	10051	trapperプロセスが待ち受けるポート番号。
LogFile	×	-	ログファイル名。設定しない場合、syslogが使用されます。
LogFileSize	×	1	LogFileのログローテーション設定をコントロールします。デフォルトでは、ログファイルのサイズが1MBに達すると自動的にログローテーションが行われます。このパラメータはMB単位で指定します。

パラメータ	必須	デフォルト値	説明
			0に設定すると、ログローテーションは行われません。
NodeID	×	0	一意なノードID(0~999)を指定します。スタンドアロンZABBIXサーバの場合は0を設定するか、何も設定しないでください。
NodeNoEvents	×	0	1に設定すると、ローカルイベントはマスターノードに送信されません。
NodeNoHistory	×	0	1に設定すると、ローカルヒストリはマスターノードに送信されません。
PidFile	×	/tmp/zabbix_server.pid	PIDを保存するファイル名。
PingerFrequency	×	30	ZABBIXサーバが各サーバにpingする間隔(秒: 1~3600)を設定します。
SenderFrequency	×	30	デーモンがアラートの送信を試行する間隔(秒)を設定します。
SourceIP	×	-	プロセスによって確立される全ての接続のソースIPアドレスを設定します。
StartDBSyncers	×	0	データベースキャッシュを有効にします。 0 - 無効 1 - 有効
StartDiscoverers	×	1	起動するdiscovererの数(0~255)。
StartHTTPPollers	×	5	起動するHTTP pollerの数(0~255)。
StartPollers	×	5	起動するpollerの数(0~255)。
StartPollersUnreachable	×	1	監視不能ホスト用に起動するpollerの数(0~255)。
StartTrappers	×	5	起動するtrapperの数(0~255)。
Timeout	×	5	要求された値を取得する際のタイムアウト時間(1~30)を秒単位で設定します。 注: 設定例については、misc/conf/zabbix_server.confを参照してください。
TrapperTimeout	×	5	トラップを処理する際のタイムアウト時間(1~255)を秒単位で設定します。
UnavailableDelay	×	60	応答のないホストへの接続を試

パラメータ	必須	デフォルト値	説明
			る頻度(秒単位)を指定します。
DataSenderFrequency	×	10	プロキシで収集されたデータを送信する頻度(秒単位)を指定します。指定できる値は1~3600秒です。
DBHost	○	-	データベース名(通常はzabbix)
DBName	○	-	データベース名(通常はzabbix)
DBSocket	×	-	データベースsocket名。 MySQLデータベースにTCP以外の方法で接続する場合に使用します。 例: /tmp/mysql.sock
DebugLevel	×	3	デバッグレベル。以下のいずれか1つを指定します。 0 - なし 1 - クリティカル 2 - エラー 3 - 警告 4 - デバッグ
FpingLocation	×	/usr/sbin/fping	ICMP pingerの格納先。ICMP pingerにはsetuidフラグを設定する必要があります。
Fping6Location	×	/usr/sbin/fping6	TCP6に対するICMP pingerの格納先。ICMP pingerにはsetuidフラグを設定する必要があります。
Hostname	○	-	一意なプロキシ名。この名前はサーバサイドでプロキシを識別するために使用されます。
HeartbeatFrequency	×	60	ハートビートメッセージの頻度(秒単位)を指定します。0に設定すると、ハートビートメッセージは無効になります。
HousekeepingFrequency	×	1	housekeeperの実行間隔を時間単位で定義します。 PostgreSQLを使用する場合、VACUUMコマンドが実行されるので、この値は24に設定します。
ListenIP	×	-	trapperプロセスが待ち受けるインタフェース。設定しない場合、trapperは全インタフェースで待ち受けます。
ListenPort	×	10051	trapperプロセスが待ち受けるポート番号。

パラメータ	必須	デフォルト値	説明
LogFile	×	-	ログファイル名。設定しない場合、syslogが使用されます。
LogFileSize	×	1	LogFileのログローテーション設定をコントロールします。デフォルトでは、ログファイルのサイズが1MBに達すると自動的にログローテーションが行われます。このパラメータはMB単位で指定します。0に設定すると、ログローテーションは行われません。
PidFile	×	/tmp/zabbix_server.pid	PIDを保存するファイル名。
ProxyLocalBuffer	×	0	プロキシでデータをローカルに保持する期間(時間単位)を指定します。サードパーティのアプリケーションでローカルデータを使用する場合、このパラメータを使用できます。
ProxyOfflineBuffer	×	1	ZABBIXサーバに接続していない場合にプロキシでデータを保持する期間(時間単位)を指定します。古いデータは失われます。
Server	○	30	プロキシからデータを送信するZABBIXサーバのDNS名またはIPアドレスを指定します。
ServerPort	×	10051	プロキシの接続先のサーバポートを指定します。
SourceIP	×	-	プロセスによって確立される全ての接続のソースIPアドレスを設定します。
StartDBSyncers	×	0	データベースキャッシュを有効にします。 0 - 無効 1 - 有効
StartDiscoverers	×	1	起動するdiscovererの数(0~255)。
StartHTTPPollers	×	5	起動するHTTP pollerの数(0~255)。
StartPingers	×	1	起動するICMP pingerの数(0~255)。
StartPollers	×	5	起動するpollerの数(0~255)。
StartPollersUnreachable	×	1	監視不能ホスト用に起動するpollerの数(0~255)。

パラメータ	必須	デフォルト値	説明
StartTrappers	×	5	起動するtrapperの数(0~255)。
PingerFrequency	×	30	ZABBIXサーバが各サーバにpingする間隔(秒: 1~3600)を設定します。
Timeout	×	5	要求された値を取得する際のタイムアウト時間(1~255)を秒単位で設定します。
TrapperTimeout	×	5	トラップを処理する際のタイムアウト時間(1~255)を秒単位で設定します。
UnavailableDelay	×	60	応答のないホストへの接続を試行する間隔を設定します。
UnreachableDelay	×	15	応答のないホストへの接続を試行する間隔を設定します。
UnreachablePeriod	×	45	ホストが応答しない状態がこのパラメータで設定された秒数以上継続した場合、ホストは監視不能とみなされます。

3.3. ZABBIX エージェント(UNIX、スタンドアロンデーモン)

ZABBIX UNIX エージェントは、監視されるホスト上で動作します。エージェントは、ホストのパフォーマンスと可用性に関する情報を ZABBIX サーバに送信します。

エージェントは、タイプが[ZABBIX エージェント]または[ZABBIX エージェント(アクティブ)]であるようなアイテムを処理します。

ZABBIX エージェントを起動するには、以下のコマンドを実行します。

```
shell> cd bin
shell> ./zabbix_agentd
```

ZABBIX エージェントは、デーモンプロセスとして動作します。
以下のコマンドラインパラメータを指定できます。

- c --config <file> 設定ファイルを指定します。デフォルトは以下のファイルです。
 /etc/zabbix/zabbix_agentd.conf
- h --help ヘルプを表示します。
- v --version バージョン番号を表示します。
- p --print サポートされている測定基準を出力して終了します。
- t --test <metric> 指定した測定基準をテストして終了します。

ヘルプを表示するには、以下のコマンドを実行します。

```
shell> zabbix_agentd -h
```

コマンドラインパラメータの使用例を示します。

```
shell> zabbix_agentd -c /usr/local/etc/zabbix_agentd.conf
```

```
shell> zabbix_agentd --help
```

```
shell> zabbix_agentd --print
```

```
shell> zabbix_agentd -t "system.cpu.load[all,avg1]"
```

設定ファイルには、**zabbix_agentd** が使用する設定パラメータが格納されています。このファイルは必須であり、ユーザ **zabbix** にはこのファイルに対する読み取りパーミッションが必要です。以下のパラメータがサポートされています。

パラメータ	必須	デフォルト値	説明
BufferSend	×	5	バッファにデータを保持する期間(秒単位)を指定します。指定できる秒数は1~3600です。
BufferSize	×	100	バッファに格納する値の最大数を指定します。バッファが一杯になると、収集された全てのデータがZABBIXサーバまたはプロキシに送信されます。
DebugLevel	×	3	デバッグレベル: 0 - なし 1 - クリティカル 2 - エラー 3 - 警告 4 - デバッグ
DisableActive	×	0	アクティブチェック処理を無効にします。値を1に設定すると、エージェントは、ZABBIXサーバに接続してアクティブなアイテムのリストを取得するのを中止します。
DisablePassive	×	0	パッシブチェック処理を無効にします。エージェントはTCPポートの待ち受けを行いません。アクティブチェックのみを使用する場合は、このパラメータを1に設定します。
EnableRemoteCommands	×	0	リモートコマンドを有効にします。ZABBIXサーバから送信したコマンドを、エージェントに実行させることができます。

パラメータ	必須	デフォルト値	説明
Hostname	×	システムのホスト名	一意なホスト名。このホスト名は、アクティブチェックにのみ使用されます。 設定しない場合、システムのホスト名(system.hostname に格納)が使用されます。
Include	×	-	設定ファイルにインクルードするファイルを指定します。 Include パラメータは何個でも指定できます。 以下に例を示します。 Include=/etc/zabbix/user_parameters.conf
ListenIP	×	-	エージェントをバインドするIPアドレス。 ホストに複数のインターフェースが実装されている場合に使用します。
ListenPort	×	10050	待ち受けポート番号。
LogFile	×	-	ログファイル名。設定しない場合、syslogが使用されます。
LogFileSize	×	1	LogFileのログローテーション設定をコントロールします。 デフォルトでは、 ログファイル のサイズが1MBに達すると自動的にログローテーションが行われます。 このパラメータはMB単位で指定します。 0に設定すると、ログローテーションは行われません。
PidFile	×	/tmp/zabbix_agentd.pid	PIDファイル名。
RefreshActiveChecks	×	120	エージェントがアクティブチェックリストを更新する間隔を設定します。デフォルトは120秒です。
Server	○	-	ZABBIXサーバまたはプロキシのIPアドレスのカンマ区切りリスト。リストに含まれないIPアドレスからの接続は拒否されます。
ServerPort	×	10051	エージェントはこのリストのサーバのポートに接続してアクティブチェックを処理します。 ZABBIXサーバまたはプロキシのポートを使用できます。

パラメータ	必須	デフォルト値	説明
SourceIP	×	-	プロセスによって確立される全ての接続のソースIPアドレスを設定します。
StartAgents	×	5	起動するエージェントの数。
Timeout	×	3	要求された値を取得する際のタイムアウト時間(1~255)を秒単位で設定します。タイムアウトしたUserParameterプロセスをエージェントが強制終了(kill)することはありません。
UserParameter	×	-	監視するユーザ定義パラメータ。ユーザ定義パラメータは複数設定できます。値は構文に従って記述します。 例: UserParameter=users,who wc -l 注: 設定例については、misc/conf/zabbix_agentd.confを参照してください。

3.4. ZABBIX エージェント(UNIX、Inetd バージョン)

設定ファイルには、**zabbix_agent** が使用する設定パラメータが格納されています。このファイルは必須であり、ユーザ **zabbix** にはこのファイルに対する読み取りパーミッションが必要です。以下のパラメータがサポートされています。

パラメータ	必須	デフォルト値	説明
Server	○	-	ZABBIXサーバまたはプロキシのIPアドレスのカンマ区切りリスト。リストに含まれないIPアドレスからの接続は拒否されます。
Timeout	×	3	要求された値を取得する際のタイムアウト時間(1~255)を秒単位で設定します。タイムアウトしたUserParameterプロセスをエージェントが強制終了(kill)することはありません。
UserParameter	×	-	監視するユーザ定義パラメータ。ユーザ定義パラメータは複数設定できます。 例: UserParameter=users,who wc -l

注: 設定例については、misc/conf/zabbix_agent.confを参照してください。

3.5. ZABBIX エージェント(Windows)

Zabbix_agentd は、Win32/64 システム用の ZABBIX エージェントです。Windows NT 4.0、Windows 2000、Windows XP、および Windows Vista で動作します。

3.5.1. インストール

インストールは非常に簡単で、3つのステップで完了します。

ステップ 1 設定ファイルの作成

設定ファイル c:/zabbix_agentd.conf を作成します (UNIX エージェントと同じ構文を使用)。

ステップ 2 エージェントを Windows サービスとしてインストール

```
zabbix_agentd.exe --install
```

c:¥zabbix_agentd.conf 以外の設定ファイルを使用する場合は、以下のコマンドを実行してサービスをインストールする必要があります。

```
zabbix_agentd.exe --config <your_configuration_file> install
```

設定ファイルはフルパスで指定する必要があります。

ステップ 3 エージェントの実行

エージェントのサービスを開始するには、[コントロール パネル]を使用するか、または以下のコマンドを実行します。

```
zabbix_agentd.exe --start
```

注: Windows NT 4.0では、Zabbix_agentd.exeはPDH(Performance Data Helper) APIを使用して様々なシステム情報を収集するので、PDH.DLLが必要です。このDLLはWindows NT 4.0に標準で付属していないので、ダウンロードしてインストールする必要があります。この件の詳細については、Microsoft Knowledge Baseの記事番号284996を参照してください。ダウンロード用リンクもそこにあります。この記事のURLは <http://support.microsoft.com/default.aspx?scid=kb;en-us;284996>です。

3.5.2. 使用方法

コマンドラインの構文:

```
zabbix_agentd.exe [-Vhp] [-idsx] [-c <file>] [-t <metric>]
```

ZABBIX Windows エージェントには、以下のコマンドラインパラメータを指定できます。

オプション:

- c --config <file> デフォルト以外の設定ファイルを指定します (デフォルトは c:¥zabbix_agentd.conf)。
- h --help ヘルプ情報を表示します。
- V --version バージョン番号を表示します。
- p --print サポートされているチェック(測定基準)のリストを出力して終了します。
- t -test <metric> チェック(測定基準)を 1 つテストして終了します。

機能:

- i --install ZABBIX エージェントをサービスとしてインストールします。
- d --uninstall ZABBIX エージェントサービスをアンインストールします。
- s --start ZABBIX エージェントサービスを開始します。
- x --stop ZABBIX エージェントサービスを停止します。

設定ファイル(c:/zabbix_agentd.conf)には、Zabbix_agentd.exe が使用する設定パラメータが格納されています。以下のパラメータがサポートされています。

パラメータ	必須	デフォルト値	説明
Alias	×	-	パラメータのエイリアスを設定します。 長くて覚えにくいパラメータ名の代わりに短くて覚えやすい名前を使用できると便利です。たとえば、サーバからページングファイル使用率(%)を取得する場合、「perf_counter[¥Paging File(_Total)¥% Usage]」と指定しますが、設定ファイルに以下の行を追加してエイリアスを定義することもできます。Alias = pg_usage:perf_counter[¥Paging File(_Total)¥% Usage] この場合、パラメータ名として「pg_usage」を指定するだけで同じ情報を取得できます。エイリアスパラメータは必要な数だけ

パラメータ	必須	デフォルト値	説明
			け指定できます。ただし、PerfCounter設定ファイルレコードで定義されているパラメータにはエイリアスを定義できないので、注意してください。
DebugLevel	×	3	デバッグレベル。以下のいずれか1つを指定します。 0 - なし 1 - クリティカル 2 - エラー 3 - 警告 4 - デバッグ
Include	×	-	設定ファイルにインクルードするファイルを指定します。 Include パラメータは何個でも指定できます。 以下に例を示します。 Include=c:¥user_parameters.conf
ListenPort	×	10050	待ち受けポート番号。
LogFile	×	-	ログファイル名。設定しない場合、syslogが使用されます。
LogUnresolvedSy mbols	×	-	エージェントを起動する際の未解決の記号のロギングをコントロールします。指定できる値は「yes」または「no」です。
MaxCollectorProc essingTime	×	100	コレクタスレッドがデータサンプルを1個処理する場合のタイムアウト値(ミリ秒)を設定します。処理時間が指定された値を超えた場合、警告メッセージがログファイルに書き込まれます。
NoTimeWait	×	-	何の影響も及ぼしません。
PerfCounter	×	-	「<parameter_name>,<perf_counter_path>,<period>」という形式で指定します。これは、新しいパラメータ<parameter_name>を、システムパフォーマンスカウンタ<perf_counter_path>の指定した時間<period>(秒)の平均値として定義することを意味します。 たとえば、直近1分間の1秒当たりのプロセッサ割り込み回数の平均値を受信する場合、新しいパラメータ「interrupts」を以下

パラメータ	必須	デフォルト値	説明
			のように定義します。 PerfCounter = interrupts,"¥Processor(0)¥Int errupts/sec",60 パフォーマンスカウンタパスは 二重引用符で囲むことに注意し てください。平均値を計算する サンプルは、毎秒収集されま す。 typeperf -qx を実行すると、 Windowsで収集可能な全ての パフォーマンスカウンタのリスト を取得できます。
PidFile	×	-	何の影響も及ぼしません。
Server	○	-	ZABBIXサーバのIPアドレスの カンマ区切りリスト。リストに含 まれないIPアドレスからの接続 は拒否されます。
SourceIP	×	-	プロセスによって確立される全 体の接続のソースIPアドレスを 設定します。
StartAgents	×	-	何の影響も及ぼしません。
UserParameter	×	-	監視するユーザ定義パラメー タ。ユーザ定義パラメータは複 数設定できます。値は 「<key>,<shell command>」と いう形式で指定します。パイプ 文字「 」の前後に空白文字を使 用しないでください。例: UserParameter=test,echo 1

3.6. ZABBIX Sender(UNIX)

ZABBIX UNIX Sender は、パフォーマンスデータを ZABBIX サーバで処理するために送信することが
 できるコマンドラインユーティリティです。

通常は、可用性とパフォーマンスに関するデータを定期的に送信するために長期間動作するユー
 ザスクリプトで使用します。

ZABBIX Sender を起動するには、以下のコマンドを実行します。

```
shell> cd bin
```

```
shell> ./zabbix_sender -z zabbix -p 10051 -s LinuxDB3 -k db.connections -o 43
```

以下のコマンドラインパラメータを指定できます。

<code>-z --zabbix-server <zabbix server></code>	ZABBIX サーバのホスト名または IP アドレス。
<code>-p --port <zabbix server port></code>	サーバ上で動作するサーバトラッパのポート番号を指定します。デフォルトは 10051 です。
<code>-s --host <host name or IP></code>	ホスト名を指定します。ホスト IP アドレスと DNS 名では正常に動作しません。
<code>-l --sourceaddress <ip address></code>	ソース IP アドレスを指定します。
<code>-k --key <key of metric></code>	送信する測定基準名(キー)を指定します。
<code>-o --value <value></code>	キーの値を指定します。
<code>-i --input-file <input file></code>	入力ファイルから値をロードします。
<code>-h --help</code>	ヘルプを表示します。
<code>-v --version</code>	バージョン番号を表示します。

ヘルプを表示するには、以下のコマンドを実行します。

```
shell> zabbix_sender -h
```

3.7. ZABBIX Get(UNIX)

ZABBIX UNIX Get は、ZABBIX エージェントと通信して必要な情報を取得するプロセスです。通常は、ZABBIX エージェントのトラブルシューティングの際に使用します。

ZABBIX Get を起動するには、以下のコマンドを実行します。

```
shell> cd bin
shell> ./zabbix_get -s121.0.0.1 -p10050 -k"system.cpu.load[all,avg1]"
```

以下のコマンドラインパラメータを指定できます。

<code>-p --port <port number></code>	ホスト上で動作するエージェントのポート番号を指定します。デフォルトは 10050 です。
<code>-s --host <host name or IP></code>	ホストのホスト名または IP アドレスを指定します。
<code>-l --sourceaddress <ip address></code>	ソース IP アドレスを指定します。
<code>-k --key <key of metric></code>	取得する測定基準名(キー)を指定します。

- | | |
|---------------------------|----------------|
| <code>-h --help</code> | ヘルプを表示します。 |
| <code>-v --version</code> | バージョン番号を表示します。 |

ヘルプを表示するには、以下のコマンドを実行します。

```
shell> zabbix_get -h
```

4. 設定

4.1. 開発環境

ZABBIX の主な開発環境は、Ubuntu Linux です。

テストには、以下の 4 つのサーバを使用します。

- Debain Linux 2.1、Intel PII/350Mhz、192MB、IDE
- SuSE 8.1、Intel P4/1.6MHz、512MB、IDE
- Ubuntu 6.06、AMD Athlon 64 3200+、2GB、SATA
- Ubuntu 6.10、Intel Core2 6400 2.13GHz、2GB、SATA

Linux にするか、それとも他の OS にするか確定していない場合は、以下の Linux ディストリビューションを試すことをお奨めします。

- Debian Linux
- RedHat Linux
- SuSE Linux
- Ubuntu Linux

4.2. アクション

ZABBIX は、イベントに対応するために様々なオペレーションを実行します。ZABBIX で生成される 1 つまたは複数のイベントに対してアクションを 1 つ定義できます。

アクションの属性を以下に示します。

パラメータ	説明
[名前]	一意なアクション名。
[イベントソース]	イベントの発生元。現在は以下の2つのソースがサポートされています。 [トリガー] - トリガーのステータスの変化によって生成されるイベント [ディスカバリ] - オートディスカバリモジュールによって生成されるイベント
[エスカレーションを有効]	エスカレーションを有効にします。有効にすると、オペレーションに対して定義されているオペレーションステップに従ってアクションがエスカレートされます。
[期間](秒)	エスカレーションステップの増分の期間。
[イベントソース]	イベントの発生元。 [トリガー] - トリガーによって生成されたイベントに対してアクションが実行されます。 [ディスカバリ] - ディスカバリイベントに対してアクションが実行

パラメータ	説明
	されます。
[デフォルトの件名]	通知のデフォルトの件名。マクロを使用できます。
[デフォルトのメッセージ]	通知のデフォルトのメッセージ。マクロを使用できます。
[リカバリメッセージ]	有効にすると、元の問題が解決された後でリカバリメッセージが送信されます。リカバリメッセージは、以前にこの問題に関する任意のメッセージを受け取ったユーザに送信されます。
[リカバリの件名]	リカバリメッセージの件名。マクロを使用できます。
[リカバリメッセージ]	リカバリメッセージ。マクロを使用できます。
[ステータス]	アクションのステータス。 [有効] - アクションは有効です。 [無効] - アクションは無効です。

4.2.1. アクションコンディション

アクションは、定義されているコンディションの組み合わせを満たす場合にのみ実行されます。

トリガーをソースとするイベントには、以下のコンディションを定義できます。

コンディションタイプ	サポートされている演算子	説明
[アプリケーション]	[=]、[含まれる]、[含まれない]	[=] - アプリケーションの一部であるトリガーで生成されたイベント [含まれる] - アプリケーションの一部であり、特定の文字列を含むトリガーで生成されたイベント [含まれない] - アプリケーションの一部であり、特定の文字列を含まないトリガーで生成されたイベント
[ホストグループ]	[=]、[<>]	イベントを生成したトリガーが登録されているホストグループと比較します。 [=] - このホストグループで生成されたイベント [<>] - 別のホストグループで生成されたイベント
[ホストテンプレート]	[=]、[<>]	トリガーが属するホストテンプレートと比較します。 [=] - このホストテンプレートから継承されたトリガーで生成されたイベント [<>] - 別のホストテンプレートから継承されたトリガーで生成されたイベント
[ホスト]	[=]、[<>]	イベントを生成したトリガーが登録されているホストと比較します。

コンディションタイプ	サポートされている演算子	説明
		[=] - このホストで生成されたイベント [<>] - 別のホストで生成されたイベント
[トリガー]	[=]、[<>]	イベントを生成したトリガーと比較します。 [=] - このトリガーで生成されたイベント [<>] - 別のトリガーで生成されたイベント
[トリガーの名前](名前)	[含まれる]、[含まれない]	イベントを生成したトリガー名と比較します。 [含まれる] - 文字列がトリガー名に含まれる(大文字と小文字は区別される) [含まれない] - 文字列がトリガー名に含まれない(大文字と小文字は区別される)
[トリガーの深刻度]	[=]、[<>]、[>=]、[<=]	トリガーの深刻度と比較します。 [=] - 指定されたトリガーの深刻度と等しい [<>] - 指定されたトリガーの深刻度と等しくない [>=] - 指定されたトリガーの深刻度以上 [<=] - 指定されたトリガーの深刻度以下
[トリガーの値]	[=]	トリガーの値と比較します。 [=] - 指定されたトリガーの値([OK]または[障害])と等しい
[期間]	[期間中]	特定の期間内に生成されたイベントです。 [期間中] - イベント生成時刻が期間内 期間は以下の形式で指定します。 dd-dd, hh:mm-hh:mm; dddd, hh:mm:hh:mm; ...

トリガーの値:

- トリガーのステータスが正常から障害に変化した場合、トリガーの値は障害
- トリガーのステータスが障害から正常に変化した場合、トリガーの値は正常
-

注: ステータスが正常->不明->障害と遷移した場合は正常->障害と同様に、障害->不明->正常と遷移した場合は障害->正常と同様に、それぞれ処理されます。

ディスカバリーをソースとするイベントには、以下のコンディションを定義できます。

コンディションタイプ	サポートされている演算子	説明
[ホストのIPアドレス]	[=]、[<>]	検出されたホストのIPアドレスが指定されたIPアドレスの範囲に含まれているかどうかを比較します。 [=] - ホストのIPアドレスが範囲に含まれる [<>] - ホストのIPアドレスが範囲に含まれない
[サービスのタイプ]	[=]、[<>]	検出されたサービスをチェックします。 [=] - 検出されたサービスからのイベント

コンディションタイプ	サポートされている演算子	説明
		[<>] - 別のサービスからのイベント
[サービスのポート]	[=]、[<>]	検出されたサービスのTCPポート番号が指定されたポート番号の範囲に含まれているかどうかを比較します。 [=] - サービスのポートが範囲に含まれる [<>] - サービスのポートが範囲に含まれない
[ディスクバリのステータス]	[=]	[UP] - ホストアップイベントおよびサービスアップイベントと一致 [DOWN] - ホストダウンイベントおよびサービスダウンイベントと一致
[アップタイム/ダウンタイム]	[>=]、[<=]	ホストダウンイベントおよびサービスダウンイベントの場合はダウンタイムを比較します。 ホストアップイベントおよびサービスアップイベントの場合はアップタイムを比較します。 [>=] - アップタイム/ダウンタイムが指定された値以上 [<=] - アップタイム/ダウンタイムが指定された値以下 パラメータは秒単位で指定します。
[受信した値]	[=] [<>] [>=] [<=] [含まれる] [含まれない]	エージェント(ZABBIX、SNMP)から受信した値と比較します。文字列の比較で判定します。 [=] - 受信した値と等しい [<>] - 受信した値と等しくない [>=] - 受信した値以上 [<=] - 受信した値以下 [含まれる] - 受信した値に含まれる [含まれない] - 受信した値に含まれない パラメータは文字列で指定します。

以下のコンディションが計算タイプ[AND/OR]で設定されている場合、

Host group = Oracle servers

Host group = MySQL servers

Trigger name like 'Database is down'

Trigger name like 'Database is unavailable'

以下のように評価されます。

(Host group = Oracle servers **または**

Host group = MySQL servers) **かつ**

(Trigger name like 'Database is down' **または**

Trigger name like 'Database is unavailable')

4.2.2. オペレーション

イベントがコンディションを満たすときに1つまたは複数のオペレーションが実行されます。

ZABBIX では、以下のオペレーションがサポートされています。

- メッセージの送信
- リモートコマンド(IPMI を含む)

ディスカバリをソースとするイベントでは、以下のオペレーションもサポートされています。

- ホストの追加
- ホストの削除
- グループへの追加
- グループからの削除
- テンプレートへのリンク
- テンプレートからリンクを削除

パラメータ	説明
[ステップ]	このアクションに対してエスカレーションを有効にする場合、以下のエスカレーション設定を使用します。 [開始] - このステップから始めて各ステップに対して実行します。 [To] - このステップまで実行します([開始]から始めて全てのステップに対して実行する場合は0)。 [期間] - この期間の後、ステップ数を増分します。0を指定すると、デフォルトの期間が使用されます。
[オペレーションのタイプ]	アクションのタイプ。 [メッセージの送信] - ユーザにメッセージを送信します。 [コマンドの実行(Execute command)] - リモートコマンドを実行します。
[イベントソース]	
[メッセージの送信先]	メッセージの送信先。 [シングルユーザ] - シングルユーザ [ユーザグループ] - グループの全てのメンバー
[デフォルトのメッセージ]	選択すると、デフォルトのメッセージが使用されます。
[件名]	メッセージの件名。マクロを使用できます。
[メッセージ]	メッセージ本体。マクロを使用できます。
[リモートコマンド]	リモートコマンドのリスト。

注: 1.6.2以降では、イベントを生成したホスト(トリガー)に対する読み取りパーミッションを持つユーザのみに通知が送信されます。トリガー条件式の少なくとも1つのホストにアクセスできる必要があ

ります。

4.2.3. メッセージとリモートコマンドでのマクロの使用

マクロを使用すると、読みやすいレポートを作成できます。

例 1 Subject: {TRIGGER.NAME}: {TRIGGER.STATUS}

マクロが置き換えられて、以下のようなメッセージ件名が生成されます。

'Processor load is too high on server zabbix.zabbix.com: ON'

例 2 Message:
Processor load is:
{zabbix.zabbix.com:system.cpu.load[,avg1].last(0)}

マクロが置き換えられて、以下のようなメッセージが生成されます。

'Processor load is: 1.45'

例 3 Message:
Latest value:
{{HOSTNAME}}:{TRIGGER.KEY}.last(0)}
MAX for 15 minutes:
{{HOSTNAME}}:{TRIGGER.KEY}.max(900)}
MIN for 15 minutes:
{{HOSTNAME}}:{TRIGGER.KEY}.min(900)}

マクロが置き換えられて、以下のようなメッセージが生成されます。

Latest value: 1.45

MAX for 15 minutes: 2.33

MIN for 15 minutes: 1.01

4.3. マクロ

ZABBIXにはマクロが豊富に用意されており、様々な状況で使用できます。
マクロを活用することで、ZABBIXの設定にかかる時間を短縮し、透過性を高めることができます。

4.3.1. サポートされているマクロのリスト

ZABBIXでサポートされている全マクロのリストを以下に示します。

マクロ	通知とコマンド	@スクリプト	アイテムキーのパラメータ	マップラベル	トリガー条件式	トリガー名	説明
{DATE}	X						本日の日付(yyyy.mm.dd.形式)
{ESC.HISTORY}	X						エスカレーション履歴(以前に送信されたメッセージのログ)
{EVENT.AGE}	X						イベントの時期(エスカレートされたメッセージに役立つ)
{EVENT.DATE}	X						イベントの日付
{EVENT.ID}	X						アクションをトリガーしたイベントのID(数値)
{EVENT.TIME}	X						イベントの時刻
{HOSTNAME}	X		X	X		X	通知を生成したトリガーの最初のアイテムのホスト名
{HOST.CONN}		X	X	X			IPとホストDNS名(ホスト設定に依存)
{HOST.DNS}		X	X	X			ホストDNS名
{IPADDRESS}	X	X	X	X			通知を生成したトリガーの最初のアイテムのIPアドレス
{ITEM.LASTVALUE}	X					X	通知を生成したトリガー条件式の最初のアイテムの最新の値。ZABBIX 1.4.3以降でサポートされています。 {{(HOSTNAME):{TRIGGER}.KEY}.last(0)}のエイリアスでもあります。
{ITEM.NAME}	X						通知を生成したトリガーの最初のアイテムの名前

{ITEM.VALUE}		X	トリガーの表示に使用する場合は、トリガー条件式のN番目のアイテムの最新の値
{ITEM.VALUE1}			イベントの表示に使用する場合は、トリガー条件式のN番目のアイテムのイベント発生時の履歴値
...			ZABBIX 1.4.3以降でサポートされています。
{ITEM.VALUE9}			
{PROFILE.CONTACT}	X		ホストプロファイルに設定されている連絡先
{PROFILE.DEVICETYPE}	X		ホストプロファイルに設定されているデバイス名
{PROFILE.HARDWARE}	X		ホストプロファイルに設定されているハードウェア
{PROFILE.NAME}	X		ホストプロファイルに設定されている名前
{PROFILE.LOCATION}	X		ホストプロファイルに設定されている場所
{PROFILE.MACADDRESS}	X		ホストプロファイルに設定されているMACアドレス
{PROFILE.NOTES}	X		ホストプロファイルに設定されている備考
{PROFILE.OS}	X		ホストプロファイルに設定されているOS
{PROFILE.SERIALNO}	X		ホストプロファイルに設定されているシリアルNo
{PROFILE.SOFTWARE}	X		ホストプロファイルに設定されているソフトウェア
{PROFILE.TAG}	X		ホストプロファイルに設定されているタグ
{STATUS}	X		{TRIGGER.STATUS}のエイリアス
{TIME}	X		現在時刻(hh:mm:ss)
{TRIGGER.COMMENT}	X		トリガーのコメント
{TRIGGER.ID}	X		アクションをトリガーしたトリガーのID(数値)
{TRIGGER.KEY}	X		通知を生成したトリガーの最初のアイテムのキー
{TRIGGER.NAME}	X		トリガーの名前(説明)
{TRIGGER.NSEVERITY}	X		トリガーの深刻度(数値)。指定できる値は以下のとおりです。 0 - 未分類 1 - 情報 2 - 警告 3 - 軽度の障害

				4 - 重度の障害 5 - 致命的な障害 ZABBIX 1.6.2以降でサポートされています。 トリガーの深刻度。指定できる値は以下のとおりです。 [未分類] [情報] [警告] [軽度の障害] [重度の障害] [致命的な障害] [不明]
{TRIGGER.SEVERITY}	X			
{TRIGGER.STATUS}	X			トリガー状態。トリガーが障害状態の場合は「ON」、正常状態の場合は「OFF」に置き換えられます。
{TRIGGER.URL}	X			トリガーのURL
{TRIGGER.VALUE}	X		X	現在のトリガー値: 0 - トリガーがOFF状態 1 - トリガーがON状態 2 - トリガーが不明状態 このマクロは、トリガー条件式でも使用できます。
{host:key.func(param)}	X		X	トリガー条件式で使用する単純なマクロ

注: ホストラベルのマクロは1.8以降でサポートされています。

4.4. アプリケーション

アプリケーションは、ホストアイテムの集合です。たとえば、アプリケーション「MySQL Server」には、MySQL の稼働率、ディスク容量、プロセッサロードアベレージ、1 秒あたりのトランザクション数、時間のかかるクエリの数など、MySQL Server に関連する全てのアイテムが含まれます。

アイテムは 1 つまたは複数のアプリケーションにリンクされる可能性があります。

ZABBIX のフロントエンドでは、アプリケーションを使用してアイテムをグループ化します。

4.5. グラフ

ユーザ定義グラフを使用して、複雑なグラフを作成できます。作成したグラフには、メニュー項目[グラフ]から簡単にアクセスできます。

4.6. メディア

メディアは、ZABBIX アラートの配信チャネルです。ユーザには、1 つまたは複数のメディアタイプを割り当てることができます。また、メディアタイプを割り当てないこともできます。

4.6.1. メール

メールによる通知

4.6.2. Jabber

Jabber メッセージングによる通知

4.6.3. スクリプト

カスタムスクリプト。ZABBIX からスクリプトに、送信先、件名、およびメッセージの 3 つのコマンドラインパラメータが渡されます。

4.6.4. GSM モデム

ZABBIX サーバのシリアルポートに接続されたシリアル GSM モデムを使用した SMS メッセージの送信がサポートされています。

使用する前に以下のことを確認してください。

- シリアルデバイス(Linux では通常/dev/ttyS0)の速度が GSM モデムと一致していること。ZABBIX にはシリアルリンクの速度を設定する項目はなく、デフォルト設定が使用されます。
- ユーザ **zabbix** がシリアルデバイスに対する読み書き可能アクセス権を持っていること。シリアルデバイスの現在のパーミッションを調べるには、コマンド「**ls -l /dev/ttyS0**」を実行します。
- GSM モデムに PIN が登録され、電源をリセットした後も保持されること。それができない場合は、SIM カードの PIN を無効にする方法もあります。

PIN を登録するには、UNIX の minicom や Windows のハイパーターミナルなどの端末ソフトウェアでコマンド「**AT+CPIN="NNNN"**」(NNNN は PIN 番号で、二重引用符で囲むこと)を発行します。

以下の GSM モデムはテスト済みです。

- Siemens MC35
- Teltonika ModemCOM/G10

4.7. ホストテンプレート

テンプレートを使用することによって、ZABBIX の管理効率を大幅に高めることができます。

1 つのテンプレートを複数のホストにリンクできます。そのテンプレートに定義されているアイテム、トリガー、およびグラフは、自動的にリンクされているホストに追加されます。テンプレートのアイテム (トリガー、グラフ) の定義を変更すると、その変更は自動的にホストにも適用されます。

ホストテンプレートの属性を以下に示します。

パラメータ	説明
[名前]	一意なテンプレート(ホスト)名。ZABBIXノード内で一意な名前である必要があります。
[グループ]	テンプレートが属するホストグループのリスト。
[新規グループ作成]	新しいホストグループをテンプレートに割り当てます。
[テンプレートとのリンク]	階層型テンプレートの作成に使用します。

4.8. ホストグループ

ホストグループには、1 つまたは複数のホストが含まれます。1 つもホストが含まれないホストグループもあります。

ホストグループの属性を以下に示します。

パラメータ	説明
[グループ名]	一意なホストグループ名。ZABBIXノード内で一意な名前である必要があります。
[ホスト]	このグループに属するホストのリスト。

4.9. ホストの依存関係とトリガーの依存関係

ZABBIX ではホストの依存関係はサポートされていませんが、柔軟性に優れたトリガーの依存関係を使用することによって、ホストの依存関係を定義できます。

ここではその仕組みについて説明します。

トリガーには、依存するトリガーを 1 つまたは複数定義できます。このとき、トリガーの状態は依存するトリガーの状態に関係なく変化しますが、依存するトリガーが 1 つでも障害状態にあれば、通知が送信されたりアクションが実行されることはありません。

例 1 ホストの依存関係

2 つのホストとして、ルータとサーバが存在し、サーバはルータの後ろに存在する環境を想定してください。この構成でルータが停止した場合、「The router is down」と「The host is down」の 2 つの通知を受信する必要はないので、「The router is down」という通知を 1 つだけ受信するように設定します。

この場合は、「The host is down」の依存先として「The router is down」を設定します。

こうすることで、ルータとサーバが両方とも停止した場合、トリガー「The host is down」に設定されているアクションが実行されなくなります。

4.10. アイテム

アイテムとは、1 つのパフォーマンスチェック項目または可用性チェック項目です。

4.10.1. アイテムキー

引数付きパラメータと引数なしパラメータ

引数付きパラメータには、引数を渡すことができます。たとえば、`vfs.fs.free[*]`は引数付きパラメータです。*は、パラメータの引数として渡される任意の文字列です。`vfs.fs.free[]`、`vfs.fs.free[/opt]`は、どちらも正しい定義です。

使用できる文字

以下の文字を使用できます。

`0-9a-zA-Z_.,:-$<空白>`

注: 「,」と「:」を使用することはお奨めできません。将来のリリースで使用できなくなる可能性があります。Novellパラメータは引き続きサポートされる予定です。

4.10.2. プラットフォーム別のサポート

Windows パラメータについては、ZABBIX マニュアルを参照してください。以下の表は、ZABBIX 1.1beta3 以降で有効です。

パラメータシステム		Windows	Linux 2.4	Linux 2.6	FreeBSD	Solaris	HP-UX	AIX	Tru64	Mac OS/X
agent.ping		X	X	X	X	X	X	X	X	X
agent.version		X	X	X	X	X	X	X	X	X
kernel.maxfiles		-	X	X	X	-	-	-	-	-
kernel.maxproc		-	-	-	X	X	-	-	-	-
net.if.collisions[if]		-	X	X	X	X	-	-	-	-
net.if.in[if<,mode>]		-	X	X	X	X	-	-	-	-
mode	bytes	-	X	X	X	X	-	-	-	-
	packets	-	X	X	X	X	-	-	-	-
	errors	-	X	X	X	X	-	-	-	-
	dropped	-	X	X	-	-	-	-	-	-
net.if.out[if<,mode>]		-	X	X	X		-	-	-	-
mode	bytes	-	X	X	X		-	-	-	-
	packets	-	X	X	X		-	-	-	-
	errors	-	X	X	-	X	-	-	-	-
	dropped	-	X	X	-	-	-	-	-	-
net.tcp.dns[ip,zone]		-	X	X	X	X	X	X	X	-
net.tcp.listen[port]		-	-	-	X	X	-	-	-	-
net.tcp.port[<ip,>port]		X	X	X	X	X	X	X	X	X
net.tcp.service.perf[service<,ip><,port>]		-	X	X	X	X	X	X	X	-
net.tcp.services[service<,ip><,port>]		-	X	X	X	X	X	X	X	-

パラメータシステム		Windows	Linux 2.4	Linux 2.6	FreeBSD	Solaris	HP-UX	AIX	Tru64	Mac OS/X
proc.mem[<name><,user> <,mode><,cmdline>]		-	X	X	X	X	-	X	X	-
mode	sum	-	X	X	X	X	-	X	X	-
	avg	-	X	X	X	X	-	X	X	-
	max	-	X	X	X	X	-	X	X	-
	min	-	X	X	X	X	-	X	X	-
proc.num[<name><,user> <,state><,cmdline>]		-	X	X	X	X	-	X	X	-
state	all	-	X	X	X	X	-	X	X	-
	sleep	-	X	X	X	X	-	X	X	-
	zomb	-	X	X	X	X	-	X	X	-
	run	-	X	X	X	X	-	X	X	-
system.boottime		-	X	X	X	-	-	-	-	-
system.cpu.intr		-	X	X	X	X	-	-	-	-
system.cpu.load[<cpu> <,mode>]		X	X	X	X	X	X	-	-	-
mode	avg1	-	X	X	X	X	X	-	-	-
	avg5	-	X	X	X	X	X	-	-	-
	avg15	-	X	X	X	X	X	-	-	-
system.cpu.num		X	X	X	X	X	X	-	-	-
system.cpu.switches		-	-	-	X	X	-	-	-	-
system.cpu.util[<cpu><,type> <,mode>]		X	-	X	X	X	-	-	-	-
type	user	-	-	X	X	X	X	-	-	-

パラメータシステム		Windows	Linux 2.4	Linux 2.6	FreeBSD	Solaris	HP-UX	AIX	Tru64	Mac OS/X
	nice	-	-	X	X	-	X	-	-	-
	idle	-	-	X	X	X	X	-	-	-
	system	-	-	X	X	-	X	-	-	-
	kernel	-	-	-	-	X	X	-	-	-
	wait	-	-	-	-	X	X	-	-	-
	interrupt	-	-	-	X	-	-	-	-	-
mode	avg1	-	X	X	X	-	X	-	-	-
	avg5	-	X	X	X	-	X	-	-	-
	avg15	-	X	X	X	-	X	-	-	-
system.run[command<,mode>]		X	X	X	X	X	X	X	X	X
mode	wait	X	X	X	X	X	X	X	X	X
	nowait	X	X	X	X	X	X	X	X	X
system.hostname		X	X	X	X	X	X	X	X	X
system.localtime		X	X	X	X	X	X	X	X	X
system.swap.in[<swap><,type>]		-	-	X	-	X	-	-	-	-
type	count	-	-	-	-	X	-	-	-	-
	pages	-	-	-	-	X	-	-	-	-
system.swap.out[<swap><,type>]		-	-	X	-	X	-	-	-	-
type	count	-	-	-	-	X	-	-	-	-
	pages	-	-	-	-	X	-	-	-	-
system.swap.size[<swap><,type>]		X	X	X	X	X	-	-	X	-
mode	free	-	X	X	X	X	-	-	X	-

パラメータシステム		Windows	Linux 2.4	Linux 2.6	FreeBSD	Solaris	HP-UX	AIX	Tru64	Mac OS/X
	total	-	X	X	X	X	-	-	X	-
system.uname		X	X	X	X	X	X	X	X	-
system.uptime		X	X	X	X	X	-	-	-	-
system.users.num		-	X	X	X	X	X	X	X	-
vfs.dev.read[device<,type> <,mode>]		-	X	X	X	X	-	-	-	-
type	sectors	-	X	X	-	-	-	-	-	-
	operations	-	X	X	-	X	-	-	-	-
	bytes	-	-	-	-	X	-	-	-	-
	ops	-	-	-	X	-	-	-	-	-
	bps	-	-	-	X	-	-	-	-	-
mode	avg1	-	-	-	X	-	-	-	-	-
	avg5	-	-	-	X	-	-	-	-	-
	avg15	-	-	-	X	-	-	-	-	-
vfs.dev.write[device<,type> <,mode>]		-	X	X	X	X	-	-	-	-
type	sectors	-	X	X	-	-	-	-	-	-
	operations	-	X	X	-	X	-	-	-	-
	bytes	-	-	-	-	X	-	-	-	-
	ops	-	-	-	X	-	-	-	-	-
	bps	-	-	-	X	-	-	-	-	-
mode	avg1	-	-	-	X	-	-	-	-	-

パラメータシステム		Windows	Linux 2.4	Linux 2.6	FreeBSD	Solaris	HP-UX	AIX	Tru64	Mac OS/X
	avg5	-	-	-	X	-	-	-	-	-
	avg15	-	-	-	X	-	-	-	-	-
vfs.file.cksum[file]		X	X	X	X	X	X	X	X	-
vfs.file.exists[file]		X	X	X	X	X	X	X	X	X
vfs.file.md5sum[file]		X	X	X	X	X	X	X	X	-
vfs.file.regexp[file, user]		-	X	X	X	X	X	X	X	-
vfs.file.regmatch[file, user]		-	X	X	X	X	X	X	X	-
vfs.file.size[file]		X	X	X	X	X	X	X	X	-
vfs.file.time[file,<,mode>]		-	X	X	X	X	X	X	X	-
mode	modify	-	X	X	X	X	X	X	X	-
	access	-	X	X	X	X	X	X	X	-
	change	-	X	X	X	X	X	X	X	-
vfs.file.inode[fs,<,mode>]		-	X	X	X	X	X	X	X	-
mode	total	-	X	X	X	X	X	X	X	-
	free	-	X	X	X	X	X	X	X	-
	used	-	X	X	X	X	X	X	X	-
	pfree	-	X	X	X	X	X	X	X	-
	pusd	-	X	X	X	X	X	X	X	-
vfs.file.size[fs,<,mode>]		-	X	X	X	X	X	X	X	-
mode	total	-	X	X	X	X	X	X	X	-
	free	-	X	X	X	X	X	X	X	-
	used	-	X	X	X	X	X	X	X	-

パラメータシステム		Windows	Linux 2.4	Linux 2.6	FreeBSD	Solaris	HP-UX	AIX	Tru64	Mac OS/X
	pfree	-	X	X	X	X	X	X	X	-
	pusd	-	X	X	X	X	X	X	X	-
vm.memory.size[fs,<,mode>]		X	X	X	X	X	X	X	-	-
mode	total	-	X	X	X	X	X	X	X	-
	free	-	X	X	X	X	X	X	X	-
	shared	-	X	X	X	-	X	X	-	-
	buffers	-	X	X	X	-	X	X	-	-
	cached	-	X	X	X	-	X	X	-	-

4.10.3. ZABBIX エージェント

引数付きパラメータと引数なしパラメータ

引数付きパラメータには、引数を渡すことができます。たとえば、`vfs.fs.free[*]` は引数付きパラメータです。*は、パラメータの引数として渡される任意の文字列です。`vfs.fs.free[/]`、`vfs.fs.free[/opt]` は、どちらも正しい定義です。

[]内で使用できる文字:

0-9a-zA-Z.:,()_/[空白]

取得可能なパラメータのリスト

ZABBIX エージェント

キー	説明	戻り値	パラメータ	備考
agent.ping	エージェントの可用性のチェック	常に「1」	-	TCP pingとして使用可能
agent.version	ZABBIXエージェントのバージョン	文字列	-	戻り値の例: 1.3.2
kernel.maxfiles	オープンできる最大ファイル数(OS)	ファイルの数(整数)		

キー	説明	戻り値	パラメータ	備考
	で設定されている値)			
kernel.maxproc	最大プロセス数 (OSで設定されている値)	プロセスの数 (整数)		
log[file<,regexp>]	監視するログファイル	ログ	file - 詳細なファイル名 regexp - 正規表現	アクティブチェックを実行する必要があります。
net.if.collisions[if]	Out-of-windowコリジョン	衝突の数 (整数)	if - インタフェース	
net.if.in[if<,mode>]	ネットワークインタフェースの受信統計	整数	if - インタフェース mode - 以下のいずれか bytes - バイト数(デフォルト) packets - パケット数 errors - エラー数 dropped - 破棄されたパケット数	
net.if.out[if<,mode>]	ネットワークインタフェースの送信統計	整数	if - インタフェース mode - 以下のいずれか bytes - バイト数(デフォルト) packets - パケット数 errors - エラー数 dropped - 破棄されたパケット数	例: net.if.out[eth0,errors] net.if.out[eth0] [差分/時間]の設定とこのキーを組み合わせると、1秒あたりのバイト数を取得できます。
net.tcp.dns[ip,zone]	DNSサービスの動作チェック	0 - DNSは停止中 1 - DNSは動作中	ip - DNSサーバのIPアドレス(無視) zone - DNSをテストするゾーン	例: net.tcp.dns[127.0.0.1,zabbix.com]
net.tcp.listen[port]	ポートの待ち受け状態チェック	0 - 待ち受け状態ではない 1 - 待ち受け状態	port - ポート番号	例: net.tcp.listen[80]
net.tcp.port[<ip>,port]	portで指定されたポート番号に対するTCP接続可否チェック	0 - 接続不可 1 - 接続可	ip - IPアドレス(デフォルトは127.0.0.1) port - ポート番号	例: net.tcp.port[,80]は、ポート80で動作するウェブサーバの可用性をテストします。 旧記法: check_port[*]
net.tcp.service[service<,ip><,port>]	サービスが動作中でTCP接続を受け入れ可能かどうかをチェック	0 - サービスは停止中 1 - サービスは動作中 2 - サービスへの接続がタイムアウト	service - ssh、service.ntp、ldap、smtp、ftp、http、pop、nntp、imap、tcpのいずれか1つを指定 ip - IPアドレス(デフォルトは127.0.0.1)	例: net.tcp.service[ftp,,45]は、TCPポート45で動作するFTPサーバの可用性をテストします。 旧記法: check_service[*]

キー	説明	戻り値	パラメータ	備考
			port - ポート番号(デフォルトでは各サービスの標準ポート番号を使用)	
net.tcp.service.perf [service <,ip><,port>]	サービスのパフォーマンスチェック	0 - サービスは停止中 sec - サービスへの接続にかかる時間(秒単位)	service - ssh、service.ntp、ldap、smtp、ftp、http、pop、nntp、imap、tcpのいずれか1つを指定 ip - IPアドレス(デフォルトは127.0.0.1) port - ポート番号(デフォルトでは各サービスの標準ポート番号を使用)	例: net.tcp.service.perf[ssh]は、SSHサーバからの初期応答速度をテストします。 旧記法: check_service[*]
proc.mem [<name>><,user><,mode><,cmdline>]	ユーザuserが実行するプロセスnameのメモリ使用量	プロセスのメモリ使用量	name - プロセス名 user - ユーザ名(デフォルトは全ユーザ) mode - avg、max、min、sum(デフォルト)のいずれか1つを指定 cmdline - コマンドラインによるフィルター	例: proc.mem[,root] - ユーザrootが実行する全てのプロセスのメモリ使用量 proc.mem[zabbix_server,zabbix] - ユーザzabbixが実行する全てのプロセスzabbix_serverのメモリ使用量 proc.mem[,oracle,max,oracleZABBIX] - ユーザoracleが実行し、コマンドラインに「oracleZABBIX」を含むプロセスのうち、最もメモリを消費しているプロセスのメモリ使用量
proc.num [<name>><,user><,state><,cmdline>]	ユーザuserが実行し、状態がstateであるようなプロセスnameの数	プロセスの数	name - プロセス名 user - ユーザ名(デフォルトは全ユーザ) state - all(デフォルト)、run、sleep、zombのいずれか1つを指定 cmdline - コマンドラインによるフィルター	例: proc.num[,mysql] - ユーザmysqlが実行するプロセスの数 proc.num[apache2,www-data] - ユーザwww-dataが実行するプロセスapache2の数 proc.num[,oracle,sleep,oracleZABBIX] - ユーザoracleが実行し、コマンドラインに「oracleZABBIX」を含むスリープ状態のプロセスの数
system.cpu.intr	デバイスの割り込み回数	整数		
system.boottime	システム起動時のタイムスタンプ	整数		タイムスタンプを秒数で返します。

キー	説明	戻り値	パラメータ	備考
system.cpu.load[<cpu><,mode>]	CPUのロードアベレージ	プロセッサロードアベレージ (浮動小数点)	cpu - CPU番号(デフォルトは全CPU) mode - avg1(デフォルト)、avg5(直近5分間の平均値)、avg15のいずれか1つを指定	例: system.cpu.load[] 注: 戻り値はパーセント値ではありません。 旧記法: system.cpu.loadX
system.cpu.num	CPUの数	使用可能なプロセッサの数		例: system.cpu.num
system.cpu.switches	コンテキストスイッチの回数	スイッチ回数		旧記法: system[switches]
system.cpu.util[<cpu><,type><,mode>]	CPUの使用率	プロセッサロードアベレージ(パーセント値)	cpu - CPU番号(デフォルトは全CPU) type - idle、nice、user(デフォルト)、systemのいずれか1つを指定 mode - avg1(デフォルト)、avg5(直近5分間の平均値)、avg15のいずれか1つを指定	旧記法: system.cpu.idleX、 system.cpu.niceX、 system.cpu.systemX、 system.cpu.userX 例: system.cpu.util[0,user,avg5]
system.run[command<,mode>]	ホスト上で指定されたコマンドを実行	コマンドの実行結果(テキスト)	command - 実行するコマンド mode - wait(デフォルト、実行終了を待機)、nowait(待機しない)のいずれか1つを指定	例: system.run[ls -l /] - rootディレクトリの詳細なファイルリスト 注: この機能を有効にするには、エージェント設定ファイルで「EnableRemoteCommands=1」を設定する必要があります。
system.hostname	ホスト名	文字列		戻り値の例: www.zabbix.com
system.localtime	システムのローカル時刻	秒数		
system.swap.in[<device><,type>]	スワップインの回数	スワップ統計	device - スワップデバイス(デフォルトは全スワップデバイス)type - count(デフォルト、スワップイン回数)、pages(スワップインページ数)のいずれか1つを指定	例: system.swap.in[,bytes] 旧記法: swap[in]
system.swap.out[<device><,type>]	スワップインの回数	スワップ統計	device - スワップデバイス(デフォルトは全スワップデバイス)type - count(デフォルト、スワップアウト回数)、pages(スワップアウトページ数)の	例: system.swap.out[,pages] 旧記法: swap[out]

キー	説明	戻り値	パラメータ	備考
			いずれか1つを指定	
system.swap.size [<device><,mode>]	スワップ容量	バイト数またはパーセント値	device - スワップデバイス(デフォルトは全スワップデバイス)type - free(デフォルト、空きスワップ容量)、total(合計スワップ容量)、pfree(空きスワップ容量、パーセント値)、pused(使用中スワップ容量、パーセント値)のいずれか1つを指定	例: system.swap.size[,pfree] - 空きスワップ容量のパーセント値 旧記法: system.swap.free、system.swap.total
system.uname	詳細なホスト情報	文字列		戻り値の例: FreeBSD localhost 4.4-RELEASE FreeBSD 4.4-RELEASE #0: Tue Sep 18 11:57:08 PDT 2001 murray@builder.FreeBSD.org: /usr/src/sys/compile/GENERIC i386
system.uptime	システムのアップタイム(秒単位)	秒数		[単位]に「s」または「uptime」が設定されている場合、読みやすく変換された値を返します。
system.users.num	接続ユーザ数	ユーザ数		エージェントサイドでwhoコマンドが実行されます。
vfs.dev.read[device <,type>]	ディスク読み込み統計	数値	device - ディスクデバイス(デフォルトは全ディスクデバイス)type - sectors(デフォルト)またはoperationsのどちらかを指定	例: vfs.dev.read[,operations] 旧記法: io[*]
vfs.dev.write[device <,type>]	ディスク書き込み統計	数値	device - ディスクデバイス(デフォルトは全ディスクデバイス)type - sectors(デフォルト)またはoperationsのどちらかを指定	例: vfs.dev.write[,operations] 旧記法: io[*]
vfs.file.cksum[file]	ファイルのチェックサムの計算	UNIXのcksumと同じアルゴリズムで計算したチェックサム	file - ファイルのフルパス	戻り値の例: 1938292000 例: vfs.file.cksum[/etc/passwd]
vfs.file.exists[file]	ファイルが存在するかどうかのチェック	0 - ファイルは存在しない 1 - ファイルは存在	file - ファイルのフルパス	例: vfs.file.exists[/tmp/application.pid]

キー	説明	戻り値	パラメータ	備考
		する		
vfs.file.md5sum[file]	ファイルのMD5 チェックサム	ファイルのMD5 ハッシュ値。サポ ートされているのは サイズが64MB未 満のファイルのみ であり、それ以外 のファイルは未サ ポート		戻り値の例: <code>b5052decb577e0fffd 622d6ddc017e82</code> 例: <code>vfs.file.md5sum[/etc/zab bix/zabbix_agentd.conf]</code>
vfs.file.regexp[file , regexp]	ファイル内の文 字列検索	見つかった文字列	<code>file</code> - ファイルのフルパス <code>regexp</code> - GNU正規表現	例: <code>vfs.file.regexp[/etc/pass wd,zabbix]</code>
vfs.file.regmatch[file, regexp]	ファイル内の文 字列検索	0 - 式が見つから ない 1 - 式が見つか った	<code>file</code> - ファイルのフルパス <code>regexp</code> - GNU正規表現	例: <code>vfs.file.regexp[/var/log/a pp.log,error]</code>
vfs.file.size[file]	ファイルサイズ	サイズ(バイト単位)	<code>file</code> - ファイルのフルパス	ユーザzabbixにはこの ファイルに対する読み取 りパーミッションが必要で す。 例: <code>vfs.file.size[/var/log/sysl og]</code>
vfs.file.time[file< mode>]	ファイル時刻情 報	秒数	<code>file</code> - ファイルのフルパス <code>mode</code> - <code>modify</code> (デフォル ト、変更時刻)、 <code>access</code> (最終アクセス時 刻)、 <code>change</code> (最終変更時 刻)のいずれか1つを指定	例: <code>vfs.file.time[/etc/passwd ,modify]</code>
vfs.fs.inode[fs<,m ode>]	iノードの数	数値	<code>fs</code> - ファイルシステム <code>mode</code> - <code>total</code> (デフォル ト)、 <code>free</code> 、 <code>used</code> 、 <code>pfree</code> (空 きiノード、パーセント値)、 <code>puused</code> (使用中iノード、 パーセント値)のいずれか 1つを指定	例: <code>vfs.fs.inode[/,pfree]</code> 旧記法: <code>vfs.fs.inode.free[*], vfs.fs.inode.pfree[*], vfs.fs.inode.total[*]</code>
vfs.fs.size[fs<,mo de>]	ディスク容量	ディスク容量(KB単 位)	<code>fs</code> - ファイルシステム <code>mode</code> - <code>total</code> (デフォル ト)、 <code>free</code> 、 <code>used</code> 、 <code>pfree</code> (空 き容量、パーセント値)、 <code>puused</code> (使用中容量、 パーセント値)のいずれか 1つを指定	マウントされているボ リュームの場合、ローカ ルファイルシステムのディ スク容量を返します。 例: <code>vfs.fs.size[/tmp,free]</code> 旧記法: <code>vfs.fs.free[*], vfs.fs.total[*], vfs.fs.used[*], vfs.fs.pfree[*],</code>

キー	説明	戻り値	パラメータ	備考
				vfs.fs.pused[*]
vm.memory.size[<mode>]	メモリサイズ	メモリサイズ(バイト単位)	mode - total(デフォルト)、shared、free、buffers、cachedのいずれか1つを指定	旧記法: vm.memory.buffers、 vm.memory.cached、 vm.memory.free、 vm.memory.shared、 vm.memory.total
web.page.get[host,<path>,<port>]	ウェブページの内容の取得	host - ホスト名 path - HTMLドキュメントへのパス(デフォルトは/)port - ポート番号(デフォルトは80)	ウェブページのソース(テキスト)	異常終了の場合はEOFを返します。 例: web.page.get[www.zabbix.com,index.php,80]
web.page.perf[host,<path>,<port>]	ウェブページ全体のロードにかかる時間の取得	秒数	host - ホスト名path - HTMLドキュメントへのパス(デフォルトは/)port - ポート番号(デフォルトは80)	例: web.page.perf[www.zabbix.com,index.php,80]
web.page.regexp[host,<path>,<port>,<regexp>,<length>]	ウェブページで最初に正規表現に一致した文字列の取得	見つかった文字列	host - ホスト名path - HTMLドキュメントへのパス(デフォルトは/)port - ポート番号(デフォルトは80)regexp - GNU正規表現length - 返す文字数	異常終了の場合はEOFを返します。 例: web.page.get[www.zabbix.com,index.php,80,OK,2]

Linux 特有の注意事項: ZABBIX エージェントには、ファイルシステム/`proc`に対する読み取り専用アクセス権が必要です。 www.grsecurity.org で公開されているカーネルパッチは、非特権ユーザのアクセス権を制限します。

WIN32 固有のパラメータ

ここでは、ZABBIX WIN32 エージェントでのみサポートされているパラメータについて説明します。

キー	説明	戻り値	備考
agent[avg_collector_time]	直近1分間のコレクタスレッドによるサンプル処理時間の平均値	ミリ秒数	
agent[max_collector_time]	直近1分間のコレクタスレッドによるサンプル処理時間の最大値	ミリ秒数	
agent[accepted_requests]	エージェントが受け付けた処理要求の合計数	要求数	

キー	説明	戻り値	備考
agent[rejected_requests]	エージェントが拒否した処理要求の合計数	要求数	
agent[timed_out_requests]	処理がタイムアウトした要求の合計数	要求数	
agent[accept_errors]	システムコール accept()のエラー合計数	システムコールの数	
agent[processed_requests]	エージェントが正常に処理した要求の合計数	要求数	
agent[failed_requests]	処理中にエラーが発生した要求の合計数	要求数	リターンコードZBX_ERRORが生成されます。
agent[unsupported_requests]	サポートされていないパラメータに対する要求の合計数	要求数	リターンコードZBX_UNSUPPORTEDが生成されます。
perf_counter[*]	パラメータがカウンタパスの場合のパフォーマンスカウンタの値	カウンタの値	[パフォーマンス モニタ]で、使用可能なカウンタのリストを取得できます。このパラメータが正しい値を返すのは、サンプルを1つだけ必要とするカウンタ (¥System¥Threadsなど)が指定された場合だけです。CPU使用率のように複数のサンプルを必要とするカウンタが指定された場合は正しい値を返しません。
service_state[*]	パラメータで指定された名前のサービスの状態	0 - 実行中 1 - 一時停止中 2 - 開始待ち 3 - 一時停止待ち 4 - 続行待ち 5 - 停止待ち 6 - 停止 7 - 不明 255 - 該当サービスなし	パラメータには、サービスのプロパティの[名前]に表示される正確なサービス名またはEXEファイルの名前を指定する必要があります。
proc_info[<process>,<attribute>,<type>]	指定されたプロセス(複数可)の様々な情報	<process> - プロセス名(proc_cnt[]パラメータと同じ) <attribute> - 要求するプロセス属性	以下の属性が現在サポートされています。vmsize - プロセスの仮想メモリのサイズ(KB単位)wkset - プロセスのワーキングセットのサイズ(プロセスが使用する物理メモリ量、KB単位)pf - ページフォルトの数 ktime - プロセスのカーネル時間(ミリ秒単位)utime - プロセスのユーザ時間(ミリ秒単位)io_read_b - I/O処理中のプロセスによる読み取りバイト数io_read_op - プロセスが実行した読み取り処理の数io_write_b - I/O処理中のプロセスによる書き込みバイト数io_write_op - プロセスが実行した書き込み処理の数io_other_b - 読み取り/書き込み処理以外の処理中

キー	説明	戻り値	備考
			<p>にプロセスが転送したバイト数io_other_op - プロセスが実行した、読み取り/書き込み処理以外のI/O処理の数gdiobj - プロセスが使用したGDIオブジェクトの数userobj - プロセスが使用したUSERオブジェクトの数</p> <p><type> - 表示タイプ(同じ名前のプロセスが複数存在する場合に有効)。有効な値を以下に示します。min - <process>という名前の全てのプロセス中の最小値max - <process>という名前の全てのプロセス中の最大値avg - <process>という名前の全てのプロセスの平均値sum - <process>という名前の全てのプロセスの値の合計値</p> <p>例:1. 全てのInternet Explorerプロセスが使用している物理メモリ量を取得するには、以下のパラメータを指定します。 proc_info[iexplore.exe,wkset,sum] 2. Internet Explorerプロセスのページフォルトの平均数を取得するには、以下のパラメータを指定します。 proc_info[iexplore.exe,pf,avg] 注: 「io_」で始まる全ての属性、gdiobj、およびuserobjを取得できるのはWindows 2000以降のWindowsバージョンだけであり、Windows NT 4.0では取得できません。</p>

4.10.4. SNMP エージェント

SNMP エージェントが収集したデータを取得するには、SNMP をサポートするように ZABBIX を設定する必要があります。

SNMP パラメータを監視するために必要な手順を以下に示します。

ステップ 1 SNMP デバイスのホストの作成

IP アドレスとポート番号 161 を入力します。ホストの[ステータス]は[無効]に設定します。host.SNMP テンプレートを使用すると、必要なアイテムが自動的に追加されます。ただし、テンプレートとホストの間に互換性がない場合もあります。

ステップ 2 監視するアイテムの SNMP 文字列の特定

ホストを作成した後、snmpwalk(ZABBIX インストールに付属する ucd-snmp/net-snmp ソフトウェアに含まれる)または同等のツールを実行します。

```
shell> snmpwalk <host or host IP> public
```

SNMP 文字列とその最新値のリストが表示されます。リストが表示されない場合は、SNMP コミュニティが標準の public ではないことが考えられるので、それを確認する必要があります。リストが表示

されたら、監視する文字列を探します。たとえば、スイッチのポート 3 に着信するバイト数を監視する場合、以下の文字列を探します。

```
interfaces.ifTable.ifEntry.ifOctetsIn.3 = Counter 32: 614794138
```

次に、interfaces.ifTable.ifEntry.ifInOctets.3 の OID を調べるために、snmpget コマンドを実行します。

```
shell> snmpget -On 10.62.1.22 interfaces.ifTable.ifEntry.ifOctetsIn.3
```

文字列の最後の数値は、監視するポート番号です。このコマンドを実行すると、以下のような文字列が出力されます。

```
.1.3.6.1.2.1.2.2.1.10.3 = Counter32: 614794138
```

出力された OID の最後の数値もポート番号です。

3COM は、ポート 1=ポート 101、ポート 3=ポート 103 というように、ポート番号を 100 単位で使用するようです。一方 Cisco は、ポート 3=ポート 3 というようにそのまま使用します。

ステップ 3 監視するアイテムの作成

ZABBIX に戻り、アイテムをクリックして、先の手順で作成した SNMP ホストを選択します。ホストを作成する際にテンプレートを使用したかどうかによって、ホストに関連付けられている SNMP アイテムのリストが表示される場合と新しいアイテムを入力するボックスだけが表示される場合があります。ここでは snmpwalk と snmpget を使用して収集した情報に基づいてアイテムを作成するものと仮定して、新しいアイテムの[名前]フィールドに簡単な説明を英語で入力します。[ホスト]フィールドにはスイッチ/ルータを入力し、[タイプ]フィールドでは[SNMPv1 エージェント]を選択します([SNMPv2 エージェント]を選択すると正常に動作しなかったのでここでは使用しない)。コミュニティ(通常は public)を入力し、[SNMP OID]フィールドに先の手順で取得した OID を入力します。先頭にピリオドを付けることを忘れないでください(例: .1.3.6.1.2.1.2.2.1.10.3)。

[SNMP ポート]に 161 を入力し、[キー]に有効な値(「SNMP-InOctets-Bps」など)を入力します。必要に応じて[乗数]を選択し、[更新間隔]と[ヒストリの保存期間(日)]では必要に応じてデフォルト以外の値を選択します。[ステータス]に[有効]、[データ型]に[数値]、[保存時の計算]に[差分]を設定します。[差分]以外の値を設定すると、SNMP デバイスから直近の変化量ではなく累積値を取得することになるので、必ず[差分]を設定してください。

以上の内容でアイテムを追加し、ZABBIX の[ホスト]領域に戻ります。ここで SNMP デバイスを[有効]に設定し、必要な SNMP データについて[最新データ]を選択します。

例 1 一般的な例

パラメータ	説明
コミュニティ	public
OID	1.2.3.45.6.7.8.0(または.1.2.3.45.6.7.8.0)
キー	トリガーが参照する一意な文字列 例: 「my_param」

OIDは数値または文字列のどちらでも指定できます。ただし、文字列のOIDを数値表現に変換する必要がある場合があります。その場合は `snmpget` を使用します。

```
shell> snmpget -On localhost public enterprises.ucdavis.memory.memTotalSwap.0
```

ZABBIX ソースを設定する際に `--with-net-snmp` フラグまたは `--with-ucd-snmp` フラグを指定した場合は、SNMP パラメータを監視できます。

例 2 アップタイムの監視

パラメータ	説明
コミュニティ	public
OID	MIB::sysUpTime.0
キー	router.uptime
データ型	浮動小数点
単位	uptime
乗数	0.01

4.10.5. シンプルチェック

シンプルチェック

シンプルチェックは、主にエージェントレス監視またはサービスのリモートチェックに使用します。シンプルチェックを行う場合、ZABBIX エージェントは必要ありません。シンプルチェックを処理(外部接続の確立など)するのは、ZABBIX サーバの役割です。

シンプルチェックは省略可能な 2 つのパラメータを受け取ります。

`ip` - IP アドレス。デフォルト値は `127.0.0.1` です。

`port` - ポート番号。省略された場合、デフォルトの標準サービスポートを使用します。

シンプルチェックの使用例:

```
ftp,121.0.0.1,155
http,11.22.33.44
http_perf,11.22.33.44,8080
```

サポートされているシンプルチェックのリストを以下に示します。

キー	説明	戻り値
<code>icmping</code>	ICMP pingでサーバにアクセスでき	0 - ICMP pingが異常終了 1 - ICMP pingが正常終了

キー	説明	戻り値
	るかどうかのチェック	
icmppingsec	ICMP pingの応答時間	秒数
ftp,<ip>,<port>	FTPサーバが動作中で接続を受け付けるかどうかのチェック	0 - FTPサーバは停止中 1 - FTPサーバは動作中 2 - タイムアウト
http,<ip>,<port>	HTTPサーバが動作中で接続を受け付けるかどうかのチェック	0 - HTTPサーバは停止中 1 - HTTPサーバは動作中 2 - タイムアウト
imap,<ip>,<port>	IMAPサーバが動作中で接続を受け付けるかどうかのチェック	0 - IMAPサーバは停止中 1 - IMAPサーバは動作中 2 - タイムアウト
nntp,<ip>,<port>	NNTPサーバが動作中で接続を受け付けるかどうかのチェック	0 - NNTPサーバは停止中 1 - NNTPサーバは動作中 2 - タイムアウト
pop,<ip>,<port>	POPサーバが動作中で接続を受け付けるかどうかのチェック	0 - POPサーバは停止中 1 - POPサーバは動作中 2 - タイムアウト
smtp,<ip>,<port>	SMTPサーバが動作中で接続を受け付けるかどうかのチェック	0 - SMTPサーバは停止中 1 - SMTPサーバは動作中 2 - タイムアウト
ssh,<ip>,<port>	SSHサーバが動作中で接続を受け付けるかどうかのチェック	0 - SSHサーバは停止中 1 - SSHサーバは動作中 2 - タイムアウト
tcp,<ip>,<port>	TCPサービスが動作中で接続を受け付けるかどうかのチェック	0 - TCPサービスは停止中 1 - TCPサービスは動作中 2 - タイムアウト
ftp_perf,<ip>,<port>	FTPサーバが動作中で接続を受け付けるかどうかのチェック	0 - FTPサーバは停止中 それ以外の値はFTPサーバへの接続にかかった時間(ミリ秒単位)
http_perf,<ip>,<port>	HTTP(ウェブ)サーバが動作中で接続を受け付けるかどうかのチェック	0 - HTTP(ウェブ)サーバは停止中 それ以外の値はHTTPサーバへの接続にかかった時間(ミリ秒単位)
imap_perf,<ip>,<port>	IMAPサーバが動作中で接続を受け	0 - IMAPサーバは停止中 それ以外の値はIMAPサーバへの接続にかかっ

キー	説明	戻り値
	付けるかどうかのチェック	た時間(ミリ秒単位)
<code>nntp_perf,<ip>,<port></code>	NNTPサーバが動作中で接続を受け付けるかどうかのチェック	0 - NNTPサーバは停止中 それ以外の値はNNTPサーバへの接続にかかった時間(ミリ秒単位)
<code>pop_perf,<ip>,<port></code>	POPサーバが動作中で接続を受け付けるかどうかのチェック	0 - POPサーバは停止中 それ以外の値はPOPサーバへの接続にかかった時間(ミリ秒単位)
<code>smtp_perf,<ip>,<port></code>	SMTPサーバが動作中で接続を受け付けるかどうかのチェック	0 - SMTPサーバは停止中 それ以外の値はSMTPサーバへの接続にかかった時間(ミリ秒単位)
<code>ssh_perf,<ip>,<port></code>	SSHサーバが動作中で接続を受け付けるかどうかのチェック	0 - SSHサーバは停止中 それ以外の値はSSHサーバへの接続にかかった時間(ミリ秒単位)

4.10.5.1. タイムアウト処理

シンプルチェックにかかる時間が ZABBIX サーバ設定ファイルで定義されているタイムアウト時間(秒単位)を超えると、シンプルチェックは終了します。

その場合、戻り値は 2 です。

4.10.5.2. ICMP ping

ZABBIX は、外部ユーティリティ `fping` を使用して ICMP ping を処理します。このユーティリティは ZABBIX ディストリビューションに付属していないので、別途インストールする必要があります。このユーティリティが存在しない場合、正しいパーミッションが設定されていない場合、または設定ファイルの `FpingLocation` で定義されている場所と違う場所に存在する場合、ICMP ping(`icmpping` と `icmppingsec`)は処理されません。

正しいパーミッションを設定するには、ユーザ `root` で以下のコマンドを実行します。

```
shell> chown root:zabbix /usr/sbin/fping
```

```
shell> chmod 710 /usr/sbin/fping
```

```
shell> chmod ug+s /usr/sbin/fping
```

4.10.6. 内部チェック

内部チェックを使用すると、ZABBIX 内部を監視できます。内部チェックは、ZABBIX サーバが処理します。

キー	説明	戻り値
zabbix[boottime]	ZABBIXサーバプロセスの起動時間(秒単位)	エポックを起点とした経過秒数が返されます。
zabbix[history]	テーブル HISTORY に保存されている値の数	MySQL InnoDB、Oracle、または PostgreSQL を使用している場合は、このキーを使用しないでください。
zabbix[history_str]	テーブル HISTORY_STR に保存されている値の数	MySQL InnoDB、Oracle、または PostgreSQL を使用している場合は、このキーを使用しないでください。
zabbix[items]	ZABBIXデータベースに保存されているアイテムの数	
zabbix[items_unsupported]	ZABBIXデータベースに保存されている取得不可アイテムの数	
zabbix[log]	ZABBIXサーバで生成された警告メッセージとエラーメッセージを保存します。	文字。このキーを使用してアイテムを追加すると、ZABBIX内部メッセージが保存されます。
zabbix[proxy,<name>,<param>]	プロキシ関連の情報へのアクセス	<name> - プロキシ名 サポートされているパラメータのリスト(<param>): lastaccess - プロキシから受信した最後のハートビートメッセージのタイムスタンプ 以下に例を示します。 zabbix[proxy,"Germany",lastaccess]
zabbix[queue]	キュー内のアイテムの数	
zabbix[trends]	テーブル TRENDS に保存されている値の数	MySQL InnoDB、Oracle、または PostgreSQL を使用している場合は、このキーを使用しないでください。
zabbix[triggers]	ZABBIXデータベースに保存されているトリガーの数	
zabbix[uptime]	ZABBIXサーバプロセスのアップタイム(秒単位)	

4.10.7. 累計チェック

累計チェックを行う場合、監視するホスト上でエージェントを実行する必要はありません。累計情報は、ZABBIX サーバが直接データベースに対してクエリを発行して収集します。

累計アイテムのキーの構文を以下に示します。

```
groupfunc["Host group","Item key","item func","parameter"]
```

サポートされているグループ関数を以下に示します。

グループ関数	説明
grpavg	平均値
grpmax	最大値
grpmin	最小値
grpsum	合計値

サポートされているアイテム関数を以下に示します。

アイテム関数	説明
avg	平均値
count	値の個数
last	最新値
max	最大値
min	最小値
sum	合計値

累計アイテムのキーの例:

例 1 ホストグループ「MySQL Servers」の合計ディスク容量

```
grpsum["MySQL Servers","vfs.fs.size[/,total]","last","0"]
```

例 2 ホストグループ「MySQL Servers」のプロセッサロードアベレージ

```
grpavg["MySQL Servers","system.cpu.load[,avg1]","last","0"]
```

例 3 ホストグループ「MySQL Servers」の直近 5 分間における 1 秒あたりの平均クエリ数

```
grpavg["MySQL Servers","mysql.qps","avg","300"]
```

4.10.8. 外部チェック

外部チェックは、ZABBIX サーバがシェルスクリプトまたはバイナリを実行することによって実行するチェックです。

外部チェックを行う場合、監視するホスト上でエージェントを実行する必要はありません。

アイテムのキーの構文を以下に示します。

```
script[parameters]
```

script - スクリプトの名前

parameters - コマンドラインパラメータのリスト

ZABBIX サーバは、設定パラメータ **ExternalScripts** で定義されているディレクトリにあるスクリプトを実行します。1 番目のコマンドラインパラメータとしてホスト名が渡され、その後に **parameters** で指定されたパラメータが渡されます。

注: 外部チェックは適切な頻度で実行してください。過度に使用すると、ZABBIXシステムのパフォーマンスが大幅に低下する恐れがあります。

例 1

外部スクリプト `check_oracle.sh` をパラメータ「`-h 192.168.1.4`」で実行し、ホスト名が「`www1.company.com`」の場合:

```
check_oracle.sh[-h 192.168.1.4]
```

このとき、ZABBIX は以下のコマンドを実行します。

```
check_oracle.sh www1.company.com -h 192.168.1.4
```

4.11. ユーザパラメータ

エージェントの設定ファイルでユーザパラメータ(UserParameter)を定義することによって、ZABBIX エージェントの機能を拡張することができます。

4.11.1. 簡単なユーザパラメータ

監視するパラメータを新しく定義するには、ZABBIX エージェントの設定ファイルに 1 行追加して、エージェントをリスタートする必要があります。

ユーザパラメータの構文を以下に示します。

UserParameter=key,command

パラメータ	説明
キー	一意なアイテムキー

コマンド	キーの値を評価するために実行するコマンド
------	----------------------

例 1 単純なコマンド

UserParameter=ping,echo 1

キー「ping」のとき、エージェントは常に「1」を返します。

例 2 複雑なコマンドの例

UserParameter=mysql.ping,mysqladmin -uroot ping|grep alive|wc -l

エージェントは、MySQL サーバが動作中なら「1」、それ以外なら「0」を返します。

4.11.2. 引数付きユーザパラメータ

引数付きユーザパラメータを使用すると、細かいコントロールが可能になり、柔軟性が高まります。

引数付きユーザパラメータを使用する構文を以下に示します。

UserParameter=key[*],command

パラメータ	説明
キー	一意なアイテムキー。[*]はこのキーがパラメータを受け取ることを定義します。
コマンド	キーの値を評価するために実行するコマンド。[]の内容が解析され、コマンドの\$1、...、\$10 と置き換えられます。

例 1 単純な引数付きパラメータ

UserParameter=ping[*],echo \$1

ping[引数]という形式のアイテムを無制限に定義して監視することができます。

ping[0] - 常に「0」を返します。

ping[aaa] - 常に「aaa」を返します。

例 2 実用的な例

```
UserParameter=mysql.ping[*],mysqladmin -u$1 -p$2 ping|grep alive|wc -l
```

このパラメータを使用して、MySQL データベースの可用性を監視できます。引数として、ユーザ名とパスワードを渡します。

```
mysql.ping[zabbix,our_password]
```

例 3 ファイル内で正規表現と一致する行数を求める例

```
UserParameter=wc[*],grep "$2" $1|wc -l
```

このパラメータを使用して、ファイル内の行数を計算できます。

```
wc[/etc/passwd,root]  
wc[/etc/services|zabbix]
```

4.12. Windows のパフォーマンスカウンタ

perf_counter[]を使用して、Windows のパフォーマンスカウンタを効果的に監視できます。

以下に例を示します。

```
perf_counter["Processor(0)¥Interrupts/sec"]
```

監視可能なパフォーマンスカウンタの完全なリストを取得するには、以下のコマンドを実行します。

typeperf -qx

残念ながら、パフォーマンスカウンタの命名規則は各 Windows サーバのローカル設定によって異なる場合があります。この場合、ローカルの異なる複数の Windows マシンを監視するためのテンプレートの作成時に特定の問題が生じます。

全てのパフォーマンスカウンタを言語設定に関係なく一意で同一の数値形式に変換できます。

regedit を実行し、HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥WindowsNT¥CurrentVersion¥Perflib¥009 を探します。

レジストリエントリには、以下のような情報が含まれています。

```
1  
1847  
2  
System  
4
```

```

Memory
6
% Processor Time
10
File Read Operations/sec
12
File Write Operations/sec
14
File Control Operations/sec
16
File Read Bytes/sec
18
File Write Bytes/sec
....

```

したがって、パフォーマンスカウンタの文字列名を数値形式に変換するには、以下のようにパフォーマンスカウンタの各部分で対応する数値を探します。

```

System -> 2
% Processor Time -> 6
/System/% Processor Time

```

次に、これらの数値を使用して数値形式を作成します。

```
/2/6
```

4.12.1. 簡単なユーザパラメータ

監視するパラメータを新しく定義するには、ZABBIX エージェントの設定ファイルに 1 行追加して、エージェントをリスタートする必要があります。

4.13. トリガー

トリガーは論理条件式として定義され、システム状態を表します。

トリガーのステータス(条件式)は、ZABBIX サーバがこの条件式で使用されている値を新しく受信するたびに再計算されます。条件式に使用できる値を以下に示します。

値	説明
障害	通常は、何かが発生したことを意味します。たとえば、プロセスサロードアベレージが非常に高い場合です。
正常	通常のトリガー状態です。
不明	この場合、ZABBIXはトリガー条件式を評価できません。以下の原因が考えられます。

値	説明
	<ul style="list-style-type: none"> ■ サーバから応答がない ■ トリガー条件式を評価できない ■ トリガー条件式が最近変更された

4.13.1. トリガーの条件式

トリガーで使用する条件式は非常に細かい指定も可能であり、監視する統計情報に関して複雑な論理テストを作成できます。

トリガーでは以下の演算子がサポートされています(実行プライオリティの高い順)。

プライオリティ	演算子	定義
1	/	除算
2	*	乗算
3	-	減算
4	+	加算
5	<	より小さい
6	>	より大きい
7	#	等しくない。この演算子は以下の式で定義されています。 $A=B \Leftrightarrow (A < B - 0.000001) \mid (A > B + 0.000001)$
8	=	等しい。この演算子は以下の式で定義されています。 $A=B \Leftrightarrow (A > B - 0.000001) \& (A < B + 0.000001)$
9	&	論理和
10		論理積

以下の関数がサポートされています。

関数	引数	サポートされているデータ型	定義
abschange	無視	float, int, str, text	最新値と前回値の差を絶対値で返します。文字列の場合は以下の値を返します。 0 - 値が等しい 1 - 値が異なる
avg	秒数または回数	float, int	期間内の平均値を返します。パラメータには期間の長さ(秒単位)を指定します。
delta	秒数または回数	float, int	max()-min()と同じ値を返します。
change	無視	float, int, str, text	最新値と前回値の差を返します。文字列の場合は以下の値を返します。 0 - 値が等しい 1 - 値が異なる

関数	引数	サポートされているデータ型	定義
count	秒数または回数	float、int、log、str	<p>期間内(秒単位)の履歴値の数またはコンディションを満たすlast(回数)値の数を返します。</p> <p>この関数は、2番目のパラメータとしてパターン、3番目のパラメータとして演算子を受け取ります。</p> <p>例: count(600,12)は履歴に保存されている値のうち「12」の出現回数を返します。整数アイテムの場合は正確に一致する場合だけを数えます。浮動小数点アイテムの場合は値が±0.00001の許容範囲内であれば一致とみなします。文字列アイテム、テキストアイテム、およびログアイテムの場合は、値にパターンが含まれれば一致とみなします。3番目のパラメータは、整数値および浮動小数点値の場合のみ有効です。サポートされている演算子: eq - 等しい ne - 等しくない gt - より大きい ge - 以上 lt - より小さい le - 以下</p> <p>例: count(600,12,"gt")は直近600秒の間に履歴に保存されている値のうち「12」よりも大きい値の正確な出現回数を返します。</p> <p>別の例: count(#10,12,"gt")は履歴に保存されている直近10個の値のうち「12」よりも大きい値の正確な出現回数を返します。</p> <p>回数のパラメータはZABBIX 1.6.1以降でサポートされています。</p>
date	無視	any	<p>現在日付をYYYYMMDD形式で返します。 例: 20031025</p>
dayofweek	無視	any	<p>曜日を1(月曜日)から7(日曜日)までの範囲で返します。</p>
diff	無視	float、int、str、text	<p>以下の値を返します。</p> <ul style="list-style-type: none"> ■ 1 - 最新値と前回値が異なる ■ 0 - それ以外
fuzzytime	秒数	float、int	<p>タイムスタンプ(アイテムの値)とZABBIXサーバ時刻の差がN秒以内なら1、それ以外なら0を返します。</p>

関数	引数	サポートされているデータ型	定義
iregexp	1番目 - 文字列 2番目 - 秒数または回数	str, log, text	通常はsystem.localtimeと組み合わせて、ローカル時刻とZABBIXサーバのローカル時刻が同期しているかどうかをチェックします。この関数は regexp に類似していますが、大文字と小文字は区別されません。
last	秒数または回数	float, int, str, text	最新値を返します。以下のパラメータを使用します。 秒数 - 無視 回数 - N番目の値 例: last(0) - 常にlast(#1)と等しい last(#3) - 3番目に新しい値 履歴の1秒以内に2つ以上の値が存在する場合、値の正確な順序は保証されません。 回数のパラメータはZABBIX 1.6.2以降でサポートされています。
logseverity	無視	log	最終ログエントリの深刻度を返します。パラメータは無視されます。 <ul style="list-style-type: none"> ■ 0 - デフォルトの深刻度 ■ N - 深刻度(整数、Windows イベントログの場合に有効)。ZABBIX は Windows イベントログのフィールド情報からログの深刻度を取得します。
logsource	文字列	log	最終ログエントリの[ソース]がパラメータと一致するかどうかをチェックします。 <ul style="list-style-type: none"> ■ 0 - 一致しない ■ 1 - 一致する 通常は、Windowsイベントログのチェックに使用します。例: logsource("VMWare Server")
max	秒数または回数	float, int	期間内の最大値を返します。パラメータには期間の長さ(秒単位)を指定します。
min	秒数または回数	float, int	期間内の最小値を返します。パラメータには期間の長さ(秒単位)を指定します。
nodata	秒数	any	以下の値を返します。 <ul style="list-style-type: none"> ■ 1 - 期間内(秒単位)にデータを1つも受信しない場合。期間として30秒以上の値を指定する必要があります。 ■ 0 - それ以外

関数	引数	サポートされているデータ型	定義
now	無視	any	エポック(1970年1月1日00時00分00秒UTC)を起点とした経過秒数を返します。
prev	無視	float、int、str、text	前回値を返します。パラメータは無視されません。
regexp	1番目 - 文字列 2番目 - 秒数または回数	str、log、text	最新値が正規表現と一致するかどうかをチェックします。パラメータにはPosix形式の正規表現を指定します。省略可能な2番目のパラメータでは、分析する秒数または行数を指定します。この場合、複数の値が処理されます。大文字と小文字は区別されます。以下の値を返します。 <ul style="list-style-type: none"> ■ 1 - 一致する ■ 0 - それ以外
str	1番目 - 文字列 2番目 - 秒数または回数	str、log、text	最新値で文字列を検索します。パラメータには検索する文字列を指定します。大文字と小文字は区別されます。省略可能な2番目のパラメータでは、分析する秒数または行数を指定します。この場合、複数の値が処理されます。以下の値を返します。 <ul style="list-style-type: none"> ■ 1 - 見つかった ■ 0 - それ以外
sum	秒数または回数	float、int	期間内の値の合計値を返します。パラメータには期間の長さ(秒単位)を指定します。
time	無視	any	現在の時刻を返します。例: 123055

注: 一部の関数は数値以外のパラメータには使用できません。

数値関数のほとんどは、引数として秒数を受け取ります。プリフィクス「#」を使用して、引数に別の意味を持たせることもできます。

引数	定義
sum(600)	600秒間の全ての値の合計値
sum(#600)	直近600個の値の合計値

トリガーでは以下の定数がサポートされています。

定数	定義
<number>	正の浮動小数点。

定数	定義
	例: 0, 1, 0.15, 123.55
<number><K M G>	K - 1024 × N M - 1024 × 1024 × N G - 1024 × 1024 × 1024 × N 例: 2K, 4G, 0.5M

単純で有効な条件式は以下のような形式で指定します。

```
{<server>:<key>.<function>(<parameter>)}<operator><const>
```

パラメータを無視する関数にもパラメータを渡す必要があります。例: last(0)

例 1 `www.zabbix.com` のプロセッサロードアベレージが高すぎる状態を検出する条件式

```
{www.zabbix.com: system.cpu.load[all,avg1].last(0)}>5)
```

「`www.zabbix.com: system.cpu.load[all,avg1]`」は監視するパラメータの簡易名で、サーバが「`www.zabbix.com`」、監視するキーが「`system.cpu.load[all,avg1]`」であることを指定します。関数「`last()`」は、最新の値を参照することを意味します。「`>5`」は、`www.zabbix.com` の最新のプロセッサロードアベレージが 5 より大きい場合にトリガーが障害になることを意味します。

例 2 `www.zabbix.com` が過負荷であることを検出する条件式

```
(({www.zabbix.com: system.cpu.load[all,avg1].last(0)}>5) |
({www.zabbix.com: system.cpu.load[all,avg1].min(600)}>2))
```

この条件式は、現在のプロセッサロードアベレージが 5 より大きいか、または直近 10 分間のプロセッサロードアベレージが 2 より大きい場合に障害になります。

例 3 `/etc/passwd` が変更されたことを検出する条件式

関数 `diff()` を使用します。

```
(({www.zabbix.com: vfs.file.cksum[/etc/passwd].diff(0)})>0)
```

この条件式は、`/etc/passwd` のチェックサムの前回値が最新値と異なる場合に障害になります。

同様の式を使用して、`/etc/passwd`、`/etc/inetd.conf`、`/kernel` などの重要なファイルの変更を監視できます。

例 4 誰かがインターネットから大きいサイズのファイルをダウンロードしていることを検出する条件式

関数 `min()` を使用します。

```
{www.zabbix.com: net.if.in[eth0,bytes].min(300)}>100K
```

条件式は、直近 5 分間に eth0 で受信したバイト数が 100KB を超えた場合に障害になります。

例 5 クラスタ化された SMTP サーバのノードが 2 つとも停止したことを検出する条件式

1 つの条件式での 2 つのホストの使用について記述します。

```
{smtp1.zabbix.com:net.tcp.service[smtp].last(0)}=0 & {smtp2.zabbix.com:net.tcp.service[smtp].last(0)}=0
```

条件式は、smtp1.zabbix.com と smtp2.zabbix.com の両方で SMTP サーバが停止した場合に障害になります。

例 6 ZABBIX エージェントをアップグレードする必要があることを検出する条件式

関数 `str()` を使用します。

```
{zabbix.zabbix.com:agent.version.str(beta8)}=1
```

条件式は、ZABBIX エージェントのバージョンが beta8(おそらく 1.0beta8)である場合に障害になります。

例 7 サーバから応答がないことを検出する条件式

```
{zabbix.zabbix.com:status.last(0)}=2
```

注: 「status」は、指定されたホストに少なくとも1つの監視パラメータが存在する場合にのみ計算される特殊パラメータです。詳細については、「status」の説明を参照してください。

例 8 直近 3 分間で一度もハートビートを受信しなかったことを検出する条件式

関数 `nodata()` を使用します。

```
{zabbix.zabbix.com:tick.nodata(180)}=1
```

「tick」のタイプには[ZABBIX トラッパー]を選択します。このトリガーを使用するには、アイテム「tick」を定義する必要があります。ホストは zabbix_sender を使用して、このパラメータのデータを定期的に送信する必要があります。180 秒間データを受信しないと、トリガーの値が障害になります。

例 9 夜間の CPU 使用率を監視する条件式

関数 `time()` を使用します。

```
{zabbix:
system.cpu.load[all,avg1].min(300)}>2) & ({zabbix:system.cpu.load[all,
avg1].time(0)}>000000) &
({zabbix:system.cpu.load[all,avg1].time(0)}<060000)
```

このトリガーが障害になるのは、夜間(00:00~06:00)だけです。

4.13.2. トリガーの依存関係

トリガーの依存関係を使用して、トリガー間の関係を定義できます。

トリガーの依存関係は、複数のリソースをソースとするイベントが発生した場合に送信されるメッセージの数を制限できる便利な手段です。

たとえば、ホスト Host がルータ Router2 の背後にあり、Router2 が Router1 の背後にあるとします。

ZABBIX - Router1 - Router2 - Host

Router1 が停止すると、当然 Host と Router2 は応答なしになります。しかし、この場合に Host、Router1、Router2 のそれぞれに関する通知を全て受信する必要はありません。トリガーの依存関係を使用するとこの問題を解決できます。

この場合、以下の依存関係を定義します。

- トリガー「Host is down」はトリガー「Router2 is down」に依存
- トリガー「Router2 is down」はトリガー「Router1 is down」に依存

ZABBIX はトリガー「Host is down」のステータスを変更する前に、該当するトリガーの依存関係が定義されているかどうかをチェックします。定義されている場合、依存するトリガーの少なくとも 1 つが障害なら、変更しようとしたトリガーのステータスは変更されず、アクションの実行や通知の送信も行われません。

ZABBIX はこのチェックを再帰的に実行します。つまり、Router1 または Router2 のいずれかが応答なしの場合、Host のトリガーは更新されません。

4.13.3. トリガーの深刻度

トリガーの深刻度は、トリガーの重要性を定義します。サポートされているトリガーの深刻度を以下に示します。

深刻度	定義	色
[未分類]	不明な深刻度です。	灰色

深刻度	定義	色
[情報]	情報目的です。	明るい緑
[警告]	注意を促します。	明るい黄色
[軽度の障害]	軽度の障害が発生しています。	暗い赤
[重度の障害]	何らかの重大な障害が発生しています。	赤
[致命的な障害]	致命的な障害が発生しています。財務上の損失などが考えられます。	明るい赤

深刻度には以下の用途があります。

- トリガーを画面表示します。深刻度別に違う色で表示します。
- [トリガーのステータス]スクリーンで音声アラームを鳴らします。深刻度別に違う音声を出力します。
- ユーザメディアを選択します。深刻度別に違うメディア(通知チャンネル)を使用します。たとえば、深刻度が重度の障害の場合は SMS で通知し、それ以外はメールで通知します。

4.13.4. ヒストリ

トリガーの状態ごとに異なるコンディションを設定する必要がある場合があります。たとえば、サーバ室の温度が 20°Cを超えたときに障害になり、その後は 15°Cを下回るまで障害を維持するトリガーを定義する場合があります。

この場合、以下のトリガーを定義します。

例 1 サーバ室の温度が高すぎる状態を検出する条件式

```
{TRIGGER.VALUE}=0 & {server:temp.last(0)} > 20 |
{TRIGGER.VALUE}=1 & {server:temp.last(0)} > 15
```

{TRIGGER.VALUE}の使用法に注意してください。このマクロは現在のトリガー値を返します。

4.14. スクリーンとスライドショー

スクリーンに様々な情報をグループ化することによって、1枚のスクリーンで多くの情報に簡単にアクセスしたり、表示することができます。スクリーンビルダは使い勝手に優れており、簡単な操作で直観的にスクリーンを作成できます。

スクリーンは表形式で構成され、各セルに以下の項目を表示できます。

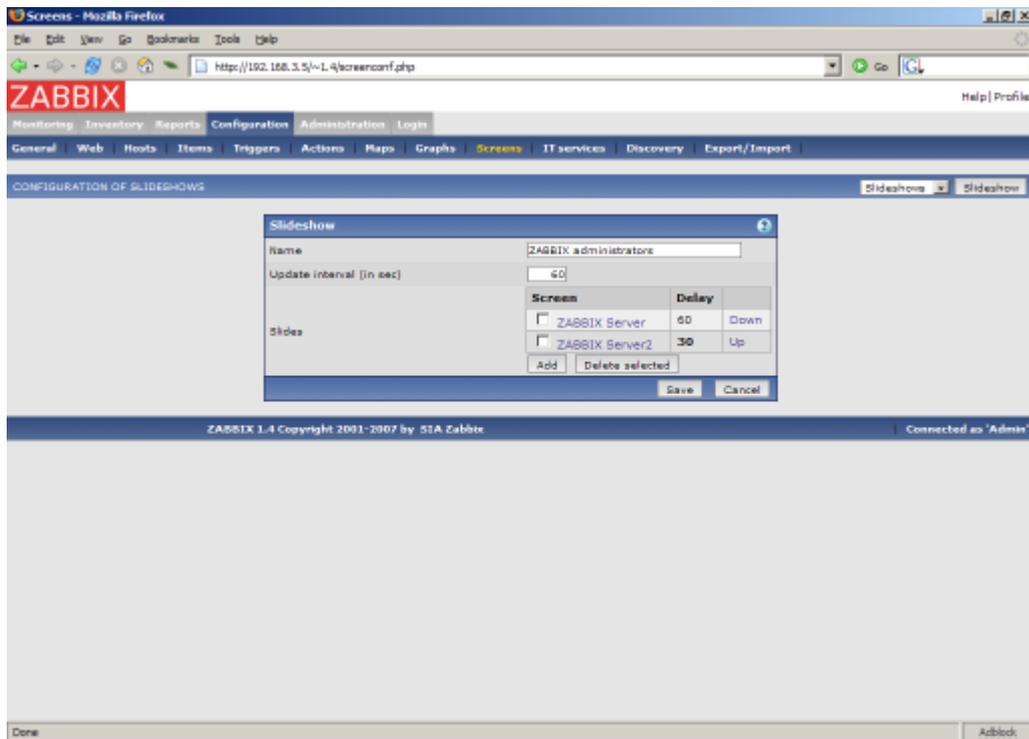
- 簡易グラフ
- ユーザ定義グラフ
- マップ
- 別のスクリーン
- プレーンテキスト情報
- サーバ情報(概要)
- トリガー情報(概要)

- データの概要
- 時刻
- イベント履歴
- アクション履歴
- URL(別の場所から取得したデータ)

1枚のスクリーンに定義できる項目の数に制限はありません。

スライドショーは複数のスクリーンの集合であり、設定されている更新間隔に従って自動的にスクリーンを切り替えます。

パラメータ	説明
[名前]	スライドショーの名前。
[更新間隔(秒)]	スクリーンを切り替えるデフォルトの間隔(秒単位)を指定します。
[スライド]	個々のスライド(スクリーン)のリスト。
[スクリーン]	スクリーンの名前。
[間隔]	対応するスクリーンを表示する時間(秒単位)を指定します。0を設定すると、スライドショーの 更新間隔 を使用します。



例 1 スライドショー「ZABBIX administrators」

上記のスライドショーは2枚のスクリーンで構成され、以下の順序で表示されます。

ZABBIX Server → 60 秒間一時停止 → ZABBIX Server2 → 30 秒間一時停止 → ZABBIX Server → 60 秒間一時停止 → ZABBIX Server2 → ...

4.15. IT サービス

IT サービスは、インフラストラクチャを上位レベル(ビジネス)の視点で監視する必要があるユーザを対象としています。このようなユーザが必要としているのは、ディスク容量不足やプロセッサロードアベレージが高すぎるといった下位レベルの細かい項目ではなく、自社の IT 部門が提供するサービスの可用性に関する情報です。その他にも、IT インフラストラクチャの弱点、各種の IT サービスの SLA、既存の IT インフラストラクチャの構造など、必要とされる上位レベルの情報は多岐に渡ります。

ZABBIX の IT サービスを使用すると、これらの情報を全て入手できます。

IT サービスは監視データを階層的に表現します。

非常に単純化した IT サービス構造の例を以下に示します。

IT サービス

|

| -ワークステーション

||

|| -ワークステーション 1

||

|| -ワークステーション 2

|

| -サーバ

階層構造の各ノードには、ステータスが割り当てられます。ステータスは、選択したアルゴリズムに基づいて計算され、上位レベルに伝達されます。トリガーは、IT サービスの最下位レベルに位置します。[仕上げ中...]

ユーザパーミッション

ZABBIX ユーザは、ウェブベースのフロントエンド経由で ZABBIX アプリケーションにアクセスします。各ユーザには一意なログイン名とパスワードが割り当てられます。ユーザパスワードは全て暗号化され、ZABBIX データベースに保存されます。ZABBIX のユーザ ID とパスワードを使用して UNIX に直接ログインすることはできません。そのためには、ZABBIX のユーザ ID とパスワードでログインできるように UNIX で設定しておく必要があります。ウェブサーバとユーザが使用するブラウザの間の通信は、SSL で保護することが可能です。

メニューからスクリーンにアクセスするためのパーミッションは、ユーザごとに設定できます。デフォルトでは、ZABBIX に登録した直後のユーザには、スクリーンに対するパーミッションは一切付与されていません。

ユーザが 30 分間何も操作しないと、自動的に切断されます。

[仕上げ中...]

4.16. ユーザパーミッション

4.16.1. 概要

ZABBIX で採用されているユーザパーミッションスキーマは柔軟性に優れ、単独の ZABBIX インストール内または分散環境におけるユーザパーミッションを効率的に管理できます。

パーミッションは、ホストグループレベルでユーザグループに対して付与されます。

ZABBIX では、様々な種類のユーザがサポートされ、種類別にユーザに付与する管理機能をコントロールします。

4.16.2. ユーザの種類

ユーザの種類を使用して、管理機能へのアクセスを定義したり、デフォルトパーミッションを指定します。

ユーザの種類	説明
ZABBIXユーザ	この種類のユーザは[監視データ]メニューにアクセスできます。デフォルトでは、リソースに対する アクセス権は付与されていません 。ホストグループに対するパーミッションは、明示的に指定する必要があります。
ZABBIX管理者	この種類のユーザは[監視データ]メニューと[設定]メニューにアクセスできます。デフォルトでは、ホストグループに対する アクセス権は付与されていません 。ホストグループに対するパーミッションは、明示的に指定する必要があります。
ZABBIX特権管理者	この種類のユーザは[監視データ]、[設定]、および[管理]の全てのメニューにアクセスできます。全てのホストグループに対して 読み書き可能なアクセス権が付与されています 。特定のホストグループへのアクセスを拒否することでパーミッションを 無効 にすることはできません。

4.17. キュー

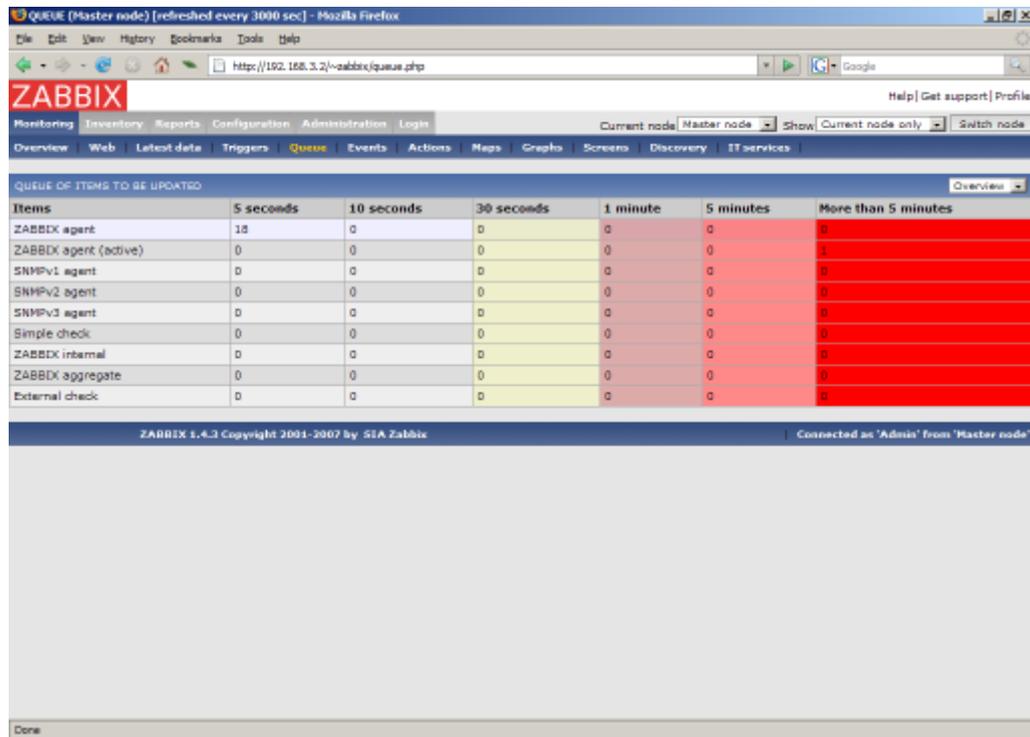
4.17.1. 概要

ZABBIX キューには、更新待ちのアイテムが表示されます。キューは、データベースのデータを論理的に表現しているだけです。ZABBIX には、IPC キューなどのキューメカニズムは存在しません。

キューで示される統計には、ZABBIX サーバのパフォーマンスが反映されます。

4.17.2. キューの見方

スタンドアロンアプリケーションまたはマスターノードのキューには、更新待ちのアイテムが表示されます。



Items	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 5 minutes
ZABBIX agent	10	0	0	0	0	0
ZABBIX agent (active)	0	0	0	0	0	1
SNMPv1 agent	0	0	0	0	0	0
SNMPv2 agent	0	0	0	0	0	0
SNMPv3 agent	0	0	0	0	0	0
Simple check	0	0	0	0	0	0
ZABBIX inbatal	0	0	0	0	0	0
ZABBIX aggregate	0	0	0	0	0	0
External check	0	0	0	0	0	0

この図からは、タイプが[ZABBIX エージェント]のアイテム 3 個が 0~5 秒間の更新待ちであること、および[ZABBIX エージェント(アクティブ)]のアイテム 1 個が 5 分以上待たされている(エージェントが停止?)ことがわかります。

子ノードについて表示されている情報は最新ではないことに注意してください。マスターノードがヒストリデータを受信する場合、一定の遅延(ノード間データ転送で通常は 10 秒間)が発生するので、その分古い情報が表示されます。

Items	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 5 minutes
ZABBIX agent	0	0	0	0	0	93
ZABBIX agent (active)	0	0	0	0	0	0
SNMPv1 agent	0	0	0	0	0	0
SNMPv2 agent	0	0	0	0	0	0
SNMPv3 agent	0	0	0	0	0	0
Simple check	0	0	0	0	0	0
ZABBIX internal	0	0	0	0	0	0
ZABBIX aggregate	0	0	0	0	0	0
External check	0	0	0	0	0	0

このスクリーンショットでは、子ノードで 93 個のアイテムが 5 分以上更新待ちになっていますが、これは以下の情報に依存しているので、信頼できる数字ではありません。

- 子ノードのパフォーマンス
- マスターノードと子ノードの間の通信
- マスターノードと子ノードのローカル時刻のずれ

注: 特殊アイテムキー `zabbix[queue]` を使用すると、ZABBIX のキューの状態を監視できます。

4.18. ユーティリティ

4.18.1. スタートアップスクリプト

システムのスタートアップ/シャットダウンの際に、スクリプトを使用して自動的に ZABBIX プロセスを開始/停止することができます。

スクリプトは、ディレクトリ `misc/init.d` に置かれています。

4.18.2. `snmptrapd.sh`

SNMP トラップ受信用のスクリプトです。このスクリプトは、`net-snmp` パッケージに付属する `snmptrapd` と一緒に使用する必要があります。

設定手順:

- `snmptrapd` (`net-snmp` または `ucd-snmp` に付属) をインストールする
- `snmptrapd.conf` を編集する

以下の行を追加

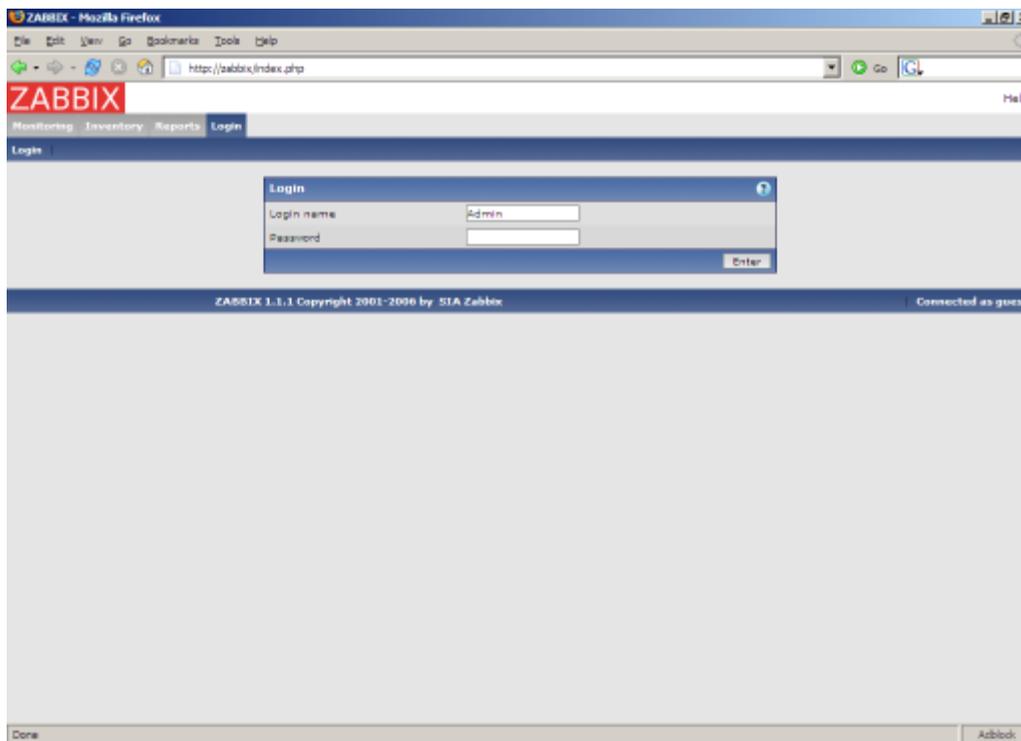
```
traphandle default /bin/bash /home/zabbix/bin/snmptrap.sh
```

- misc/snmptrap/snmptrap.sh を~zabbix/bin にコピーする
- snmptrap.sh を編集して基本設定をする
- 専用のホストとトラッパー(データ型は文字列)アイテムを ZABBIX に追加する(アイテムのキーについては snmptrap.sh を参照)
- snmptrapd を実行する

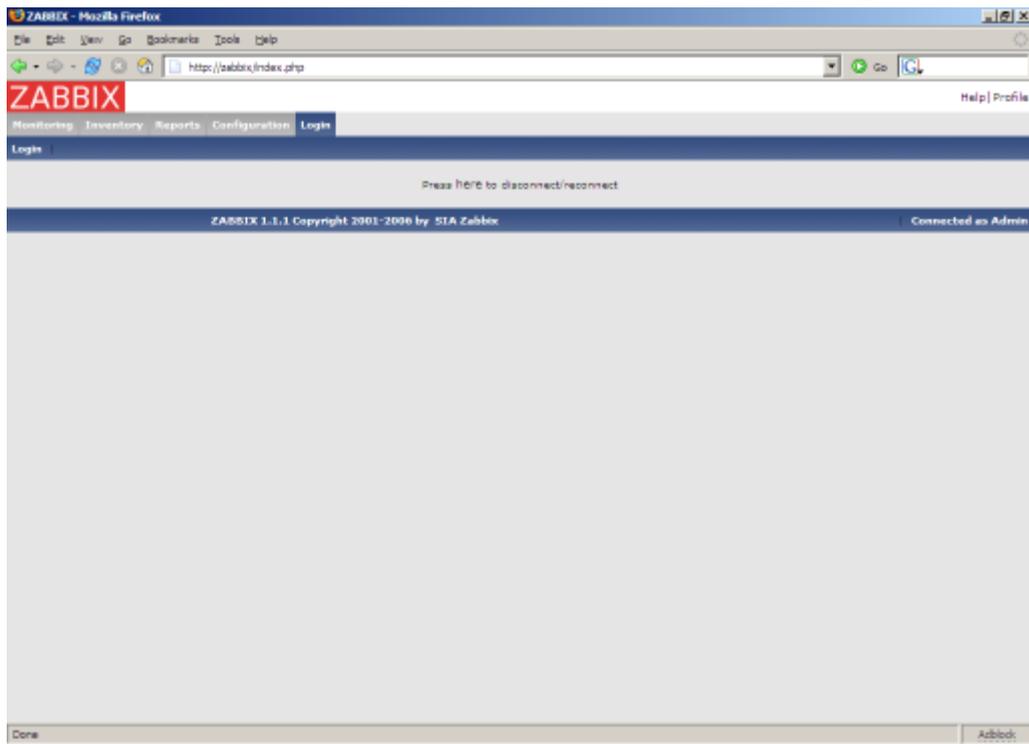
5. クイックスタートガイド

5.1. ログイン

ZABBIX で最初に表示される画面を以下に示します。インストール後は、ユーザ名「Admin」、パスワードなしで ZABBIX スーパーユーザとしてログインします。



ログインすると、[管理者として接続(Connected as Admin)]と表示され、[設定]メニューにアクセスできるようになります。



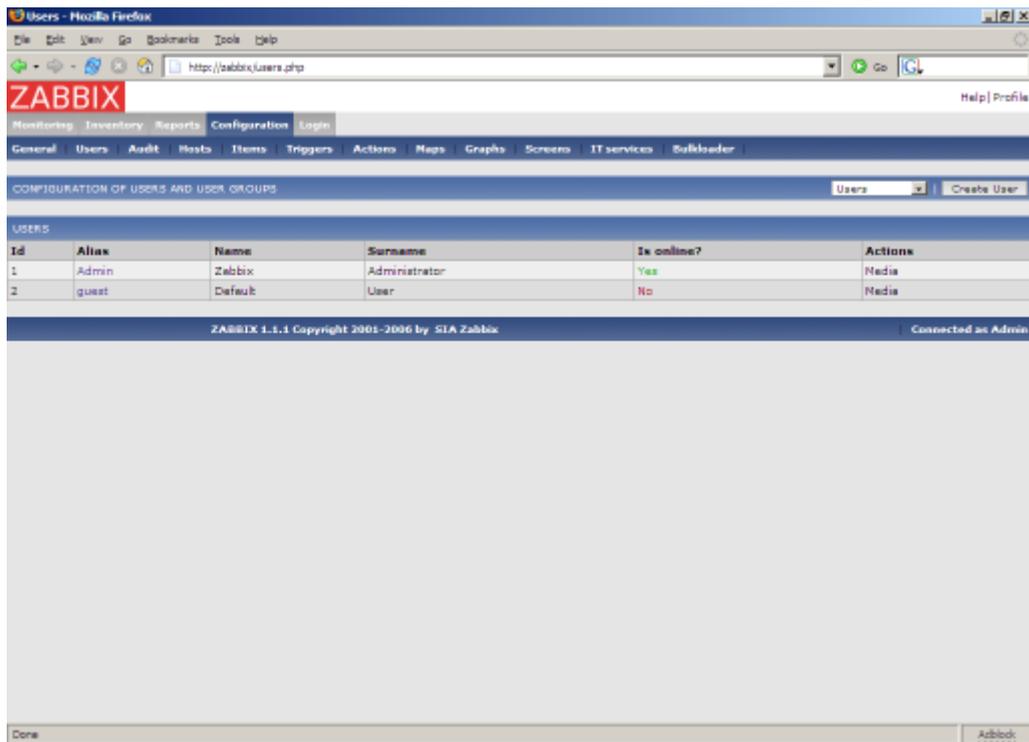
5.1.1. ブルートフォースアタックに対抗するプロテクション機能

5 回連続してログイン試行に失敗すると、ブルートフォースアタックおよびディクショナリアタックを防ぐために、ZABBIX インタフェースは次の 15 分以内に 60 秒間一時停止します。

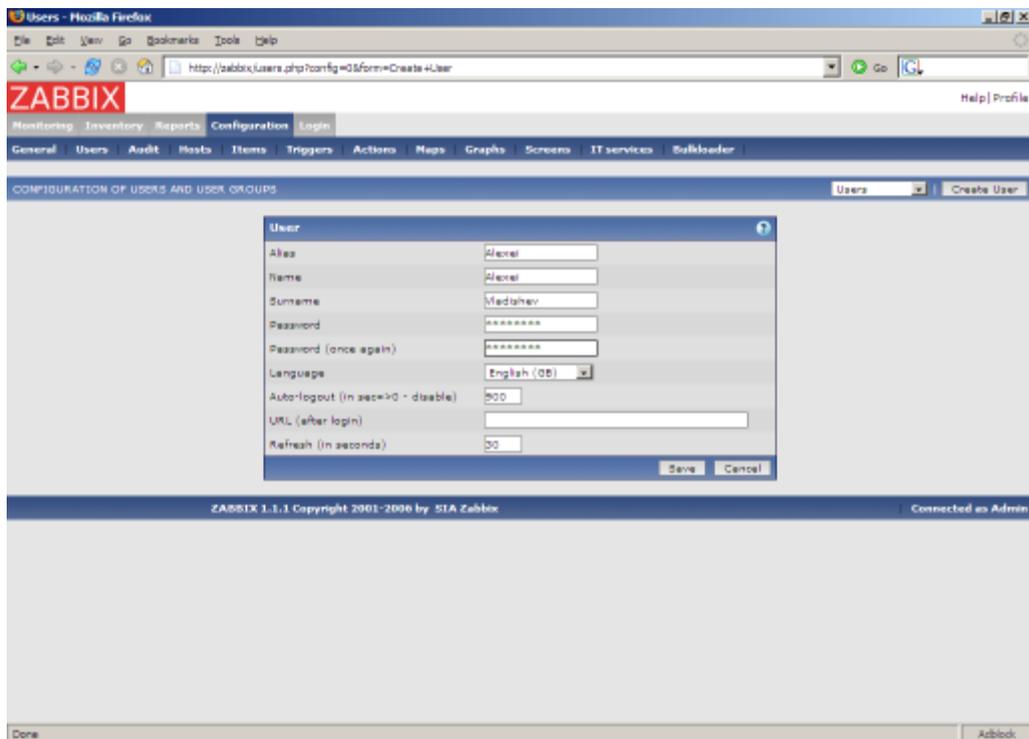
ログインに成功すると、失敗したログイン試行の IP アドレスが表示されます。

5.2. ユーザの追加

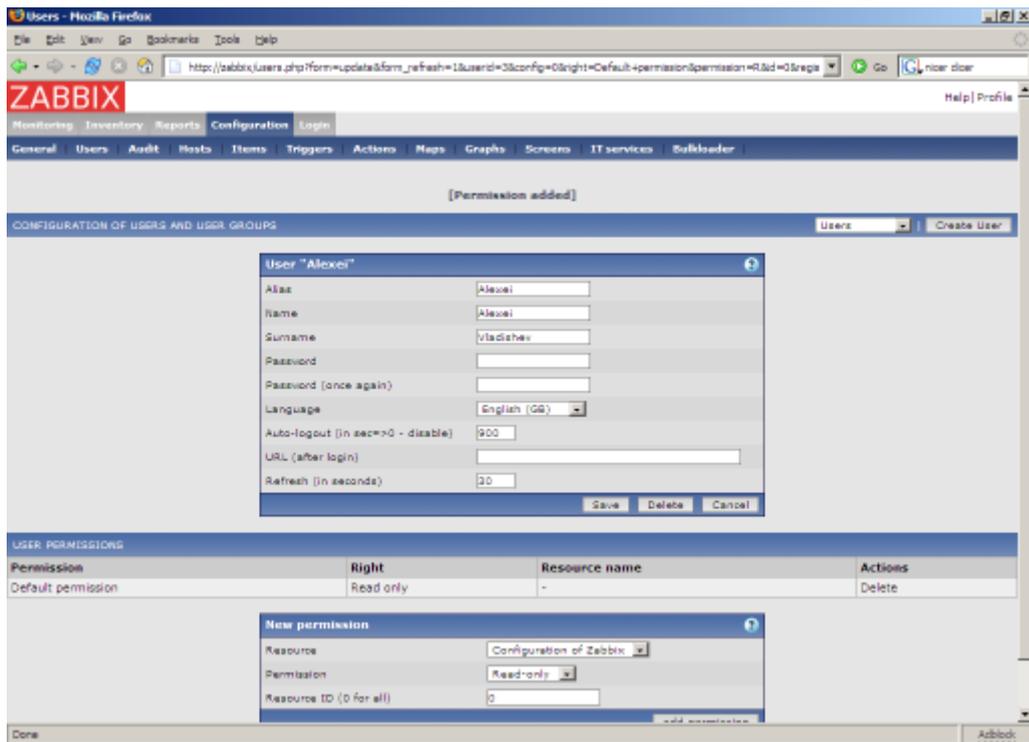
インストール直後の ZABBIX には、ZABBIX のスーパーユーザであらゆるパーミッションを付与されている「Admin」と、特殊なデフォルトユーザである「guest」の 2 つだけが定義されています。ログインしていないユーザには「guest」の持つパーミッションが付与されます。デフォルトでは、「guest」に付与されているのは読み取り専用パーミッションだけです。



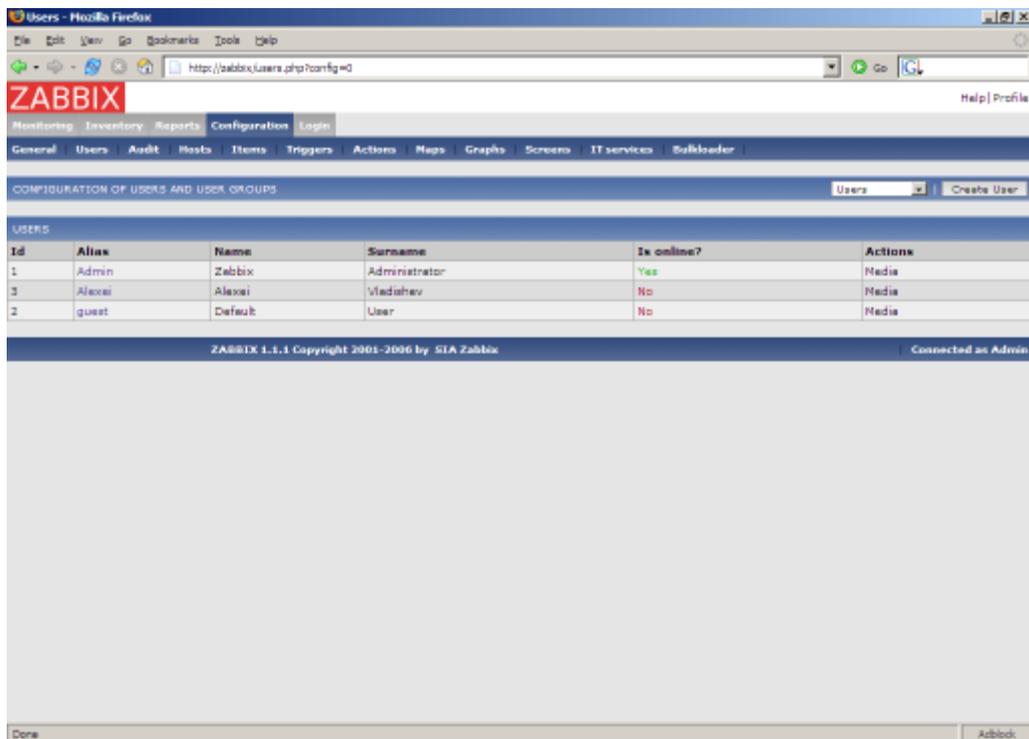
新しいユーザを追加するには、[ユーザの作成]をクリックします。



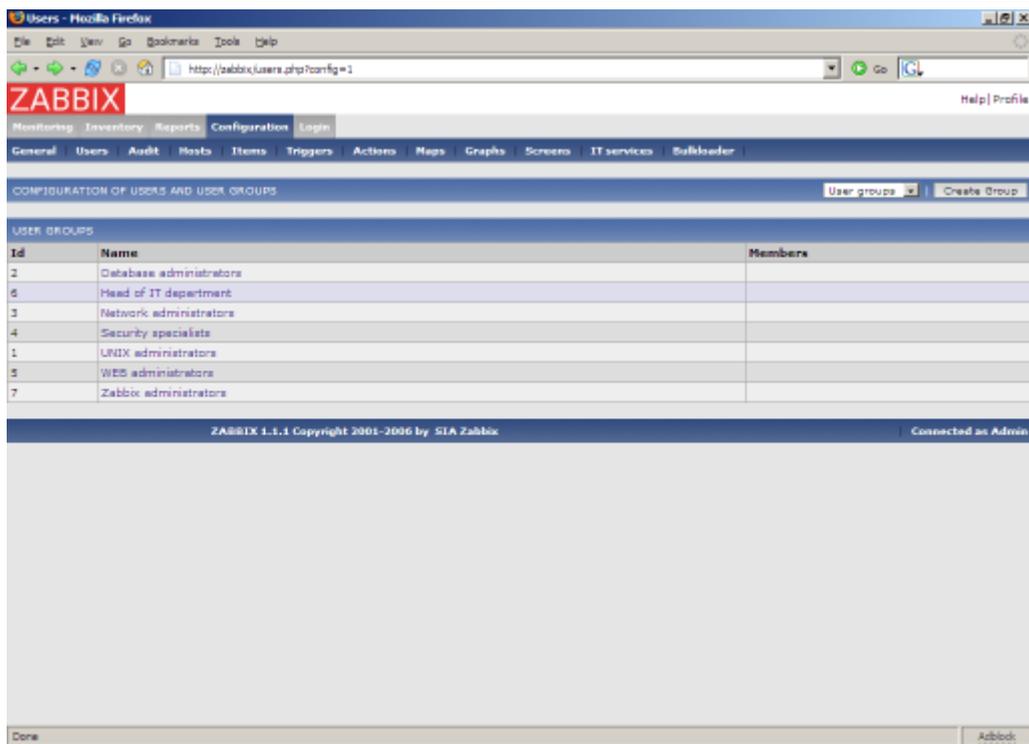
デフォルトでは新しいユーザには何もパーミッションが付与されていないので、必要に応じて付与します。



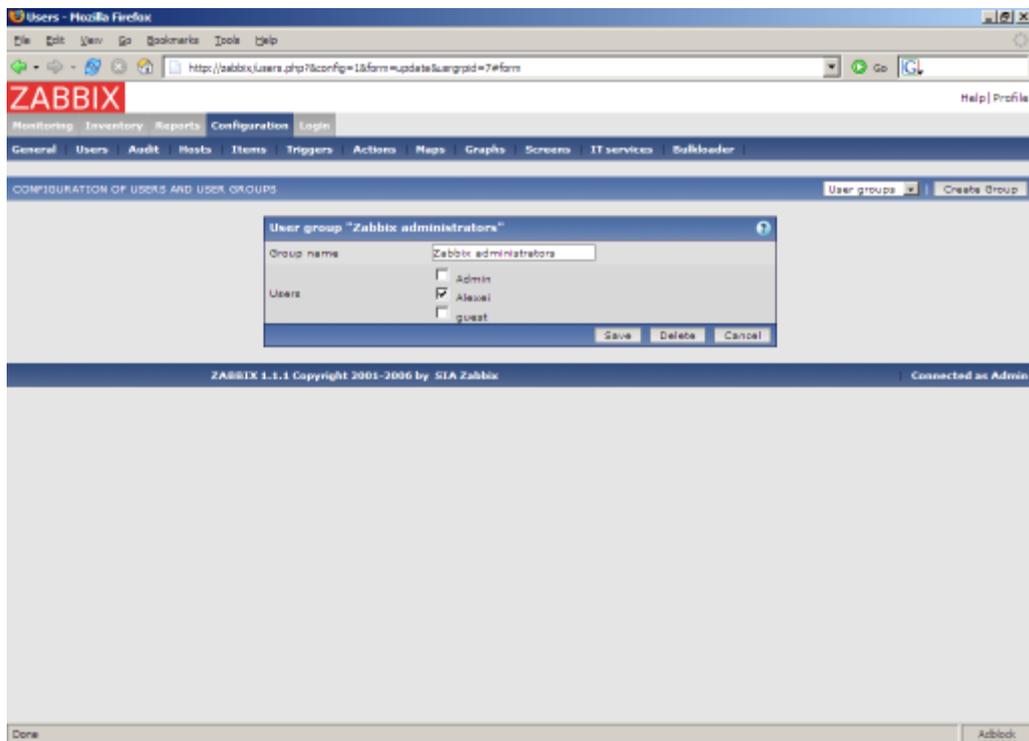
ユーザが追加されます。



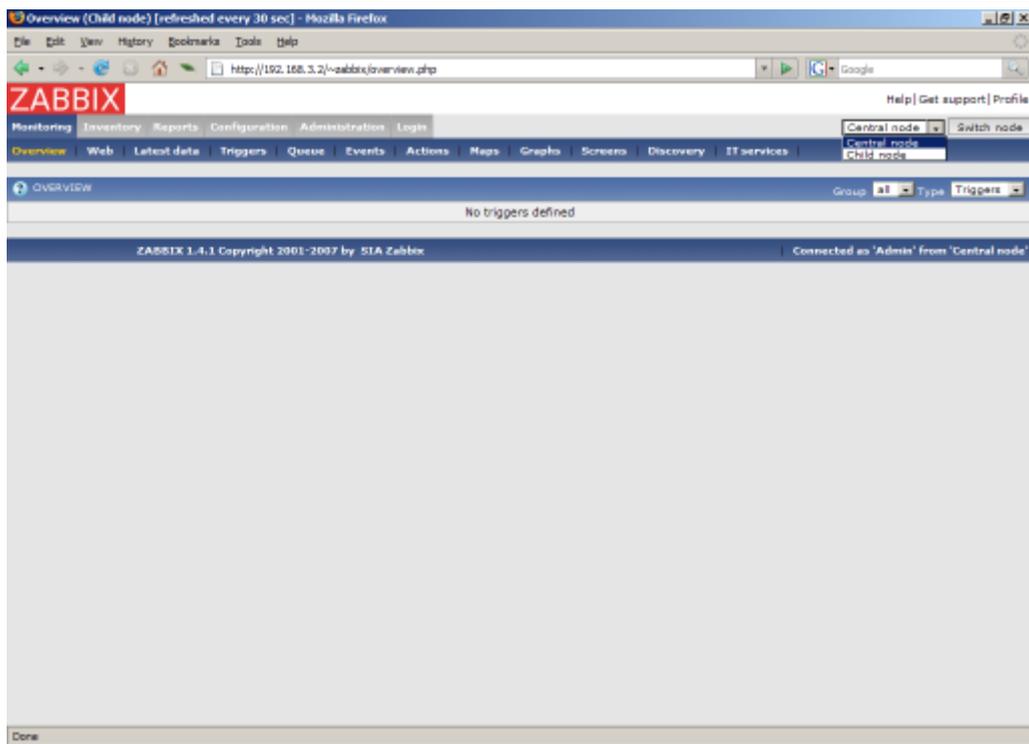
ドロップダウンから[ユーザグループ]を選択して、ユーザグループのメンバーを編集します。



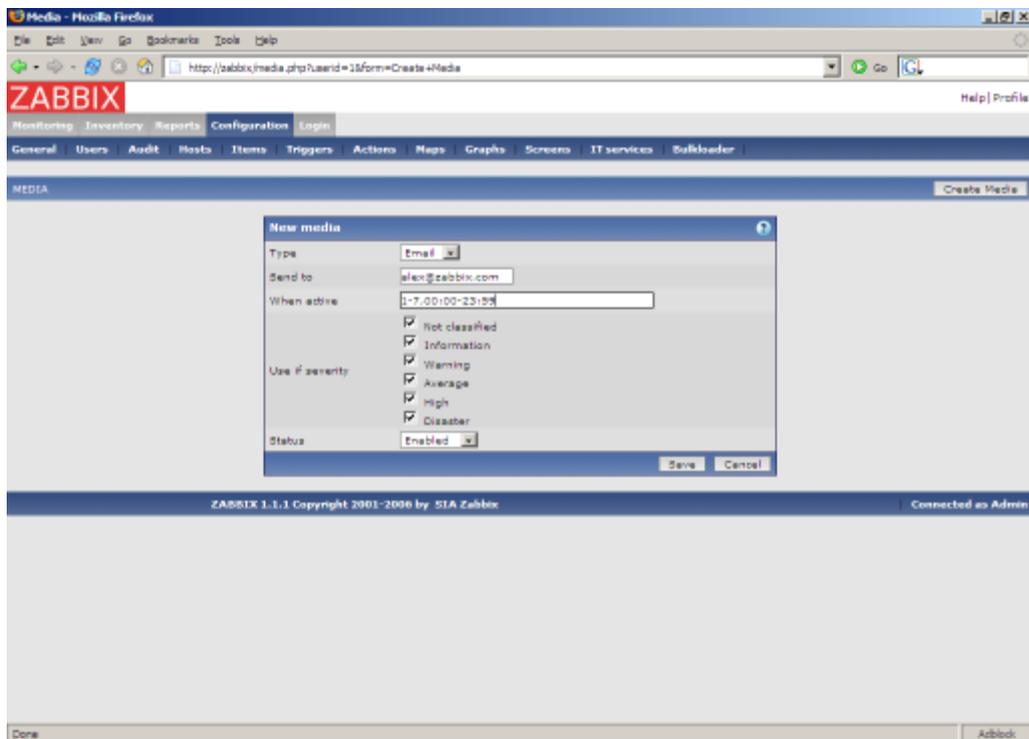
メンバーを変更するグループをクリックします。



ユーザに通知方法(メディア)を割り当てます。この時点ではまだメディアは割り当てられていません。



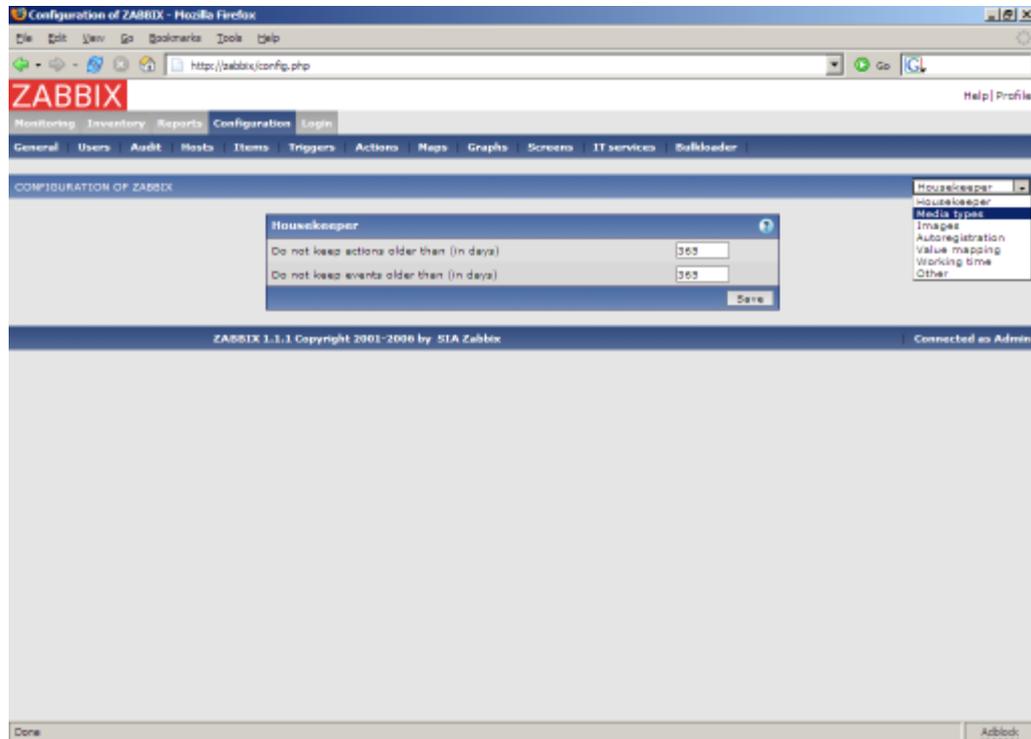
メールアドレスを入力し、メディアをアクティブにする深刻度を選択します。



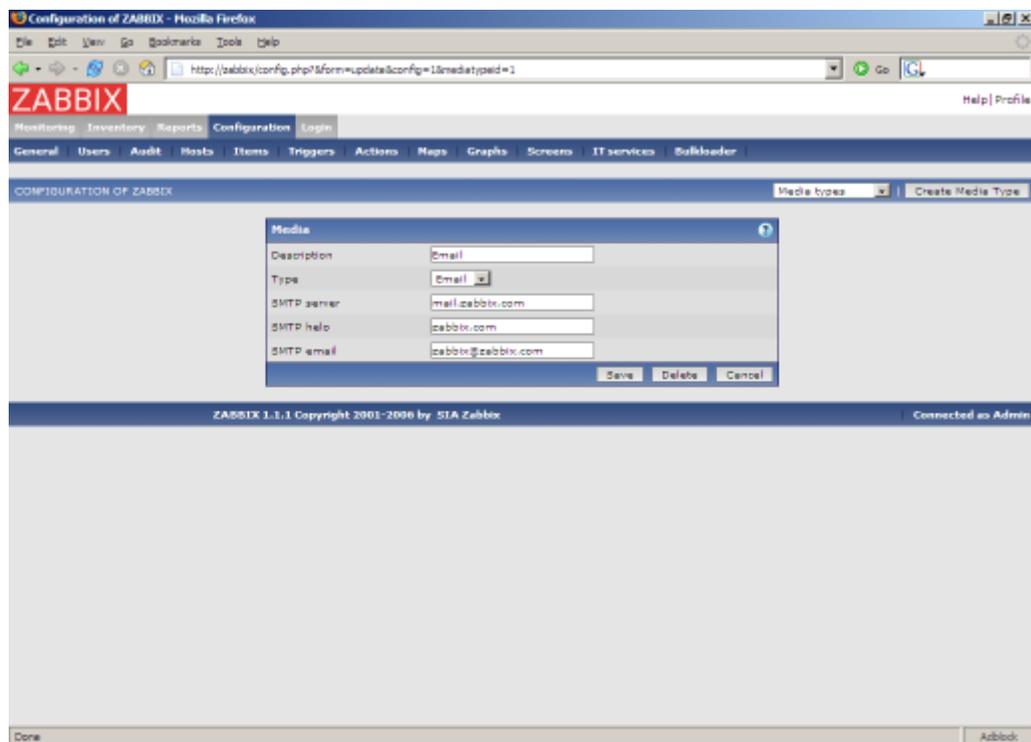
これでユーザの追加は完了です。追加したユーザでログインしてみてください。

5.3. メール設定

インストール直後は、ZABBIX に定義されている通知配信方法(メディアタイプ)は[メール]だけです。メールを設定するには、[メニュー]->[設定]->[メディアタイプ]を選択します。

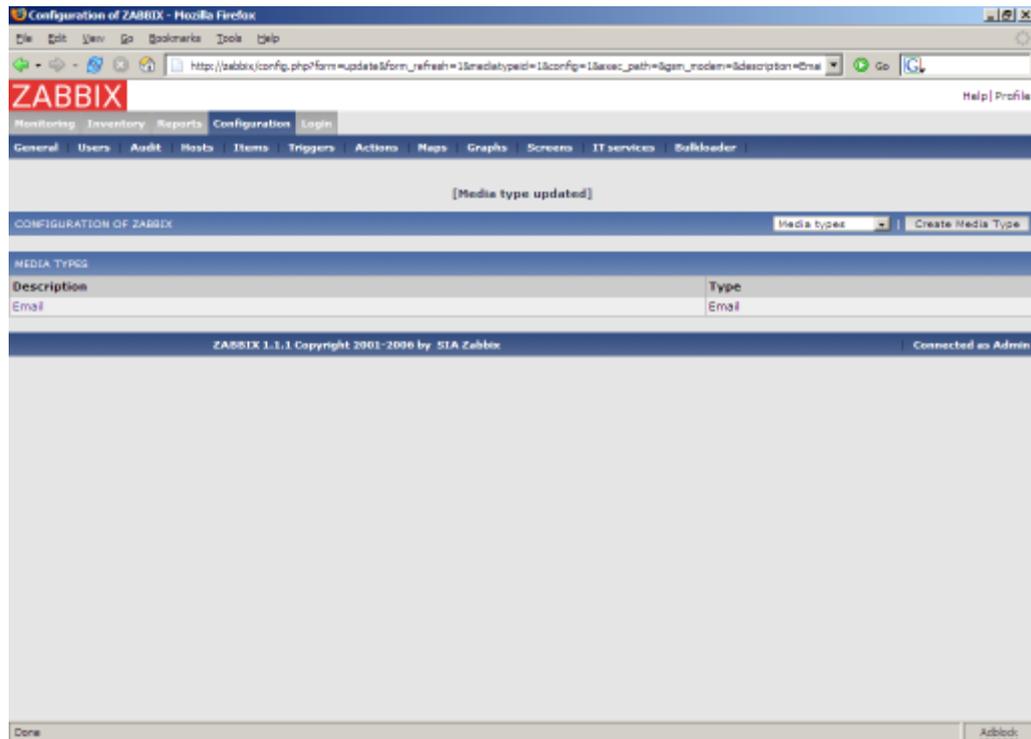


使用可能なメディアタイプのリストから[メール]を選択します。



[SMTP サーバ]、[SMTP helo]、[送信元メールアドレス]に適切な値を入力します。入力が終わった

ら[保存]をクリックします。

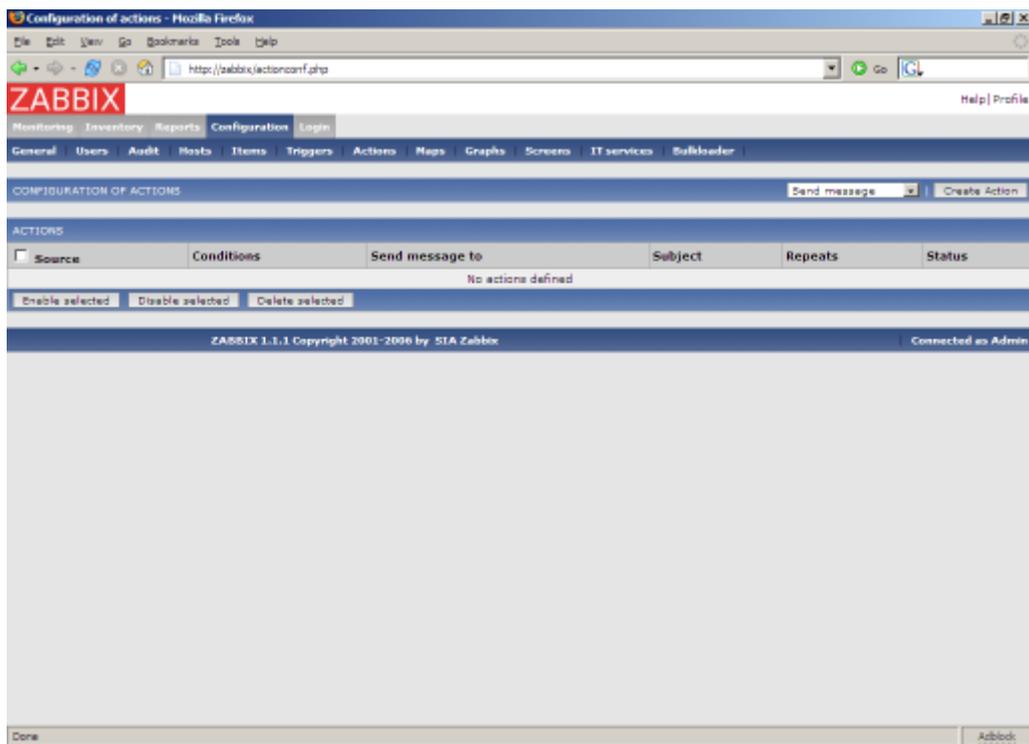


これでメディアタイプ[メール]が定義されました。メディアタイプを使用するには、ユーザとリンクする必要があります。

5.4. エージェントを使用するホストの追加

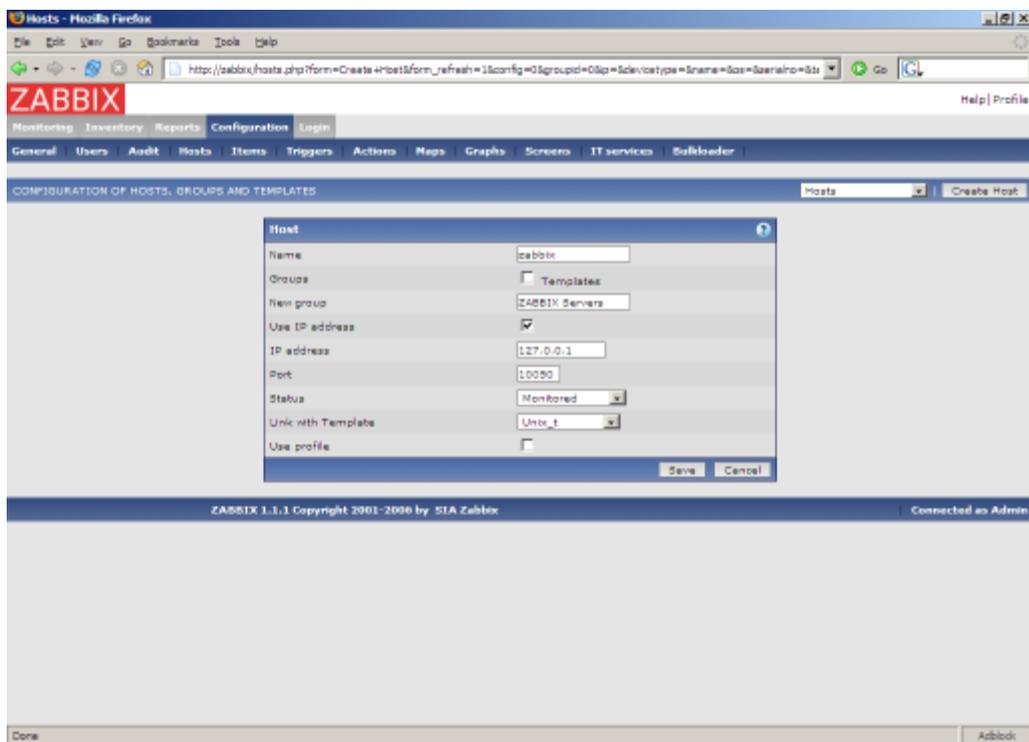
ここでは、ZABBIX エージェントを実行するホストを監視する場合の詳細について説明します。あらかじめエージェントをインストールして、正しく設定しておく必要があります。

ホストはまだ定義されていません。

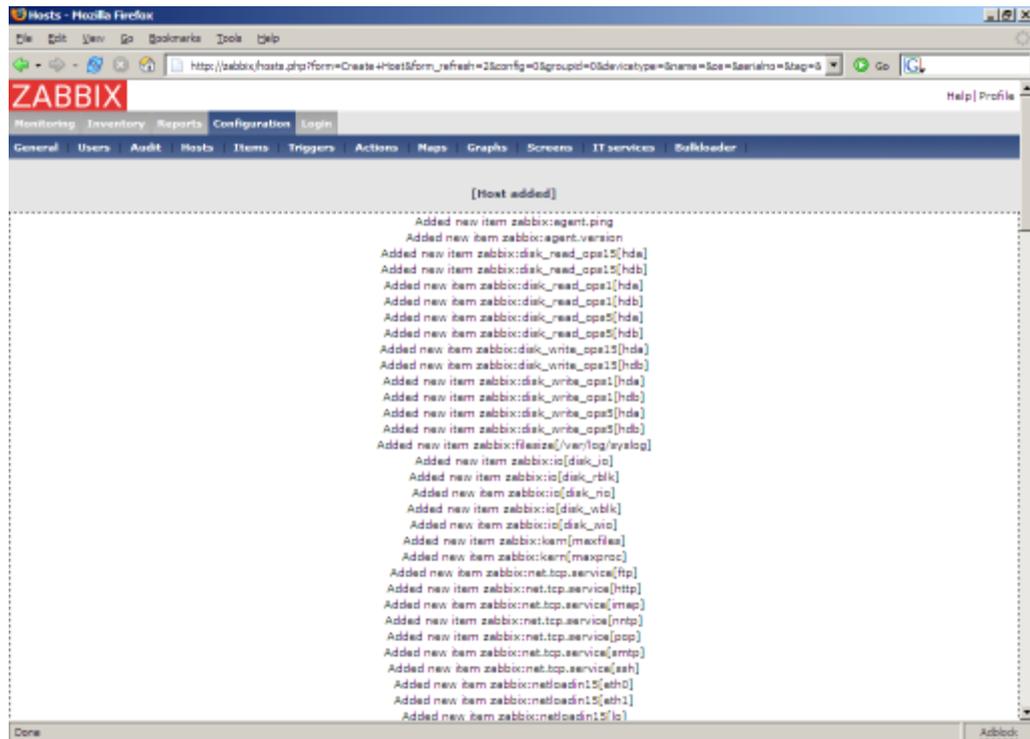


すでに ZABBIX サーバで ZABBIX エージェントが動作しており、このサーバを監視するものとします。
[ホストの作成]をクリックして、必要なデータを全て入力します。設定を単純化するために、標準テンプレート Unix_t を使用します。

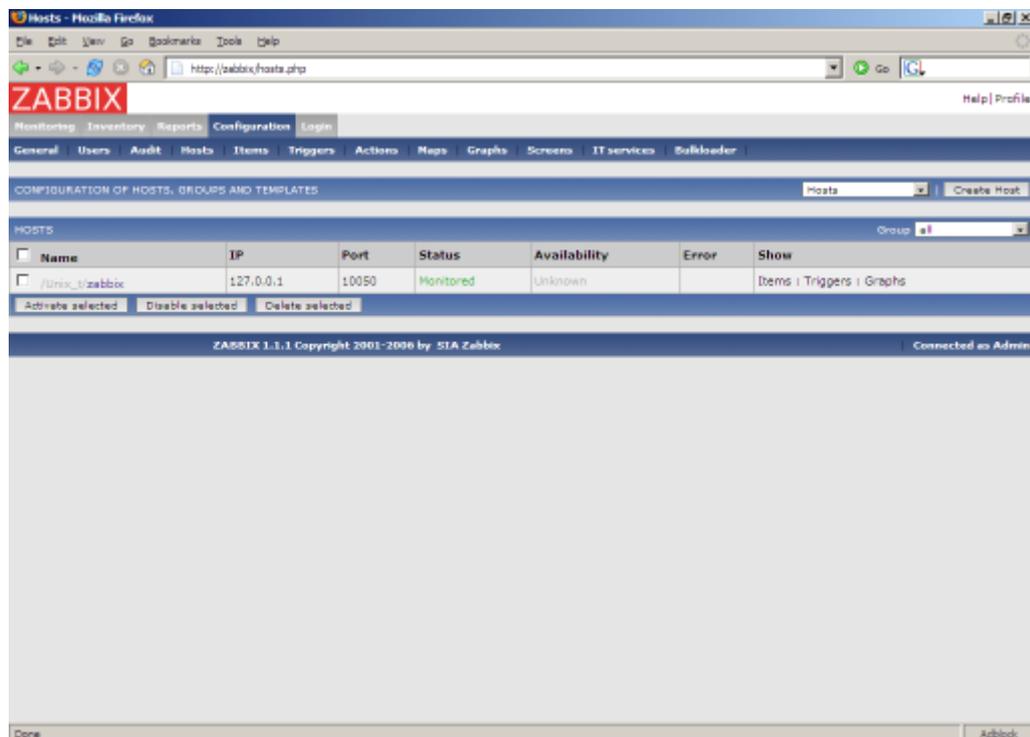
テンプレートを使用しない場合は、後で手作業でアイテムとトリガーをホストに追加する必要があります。



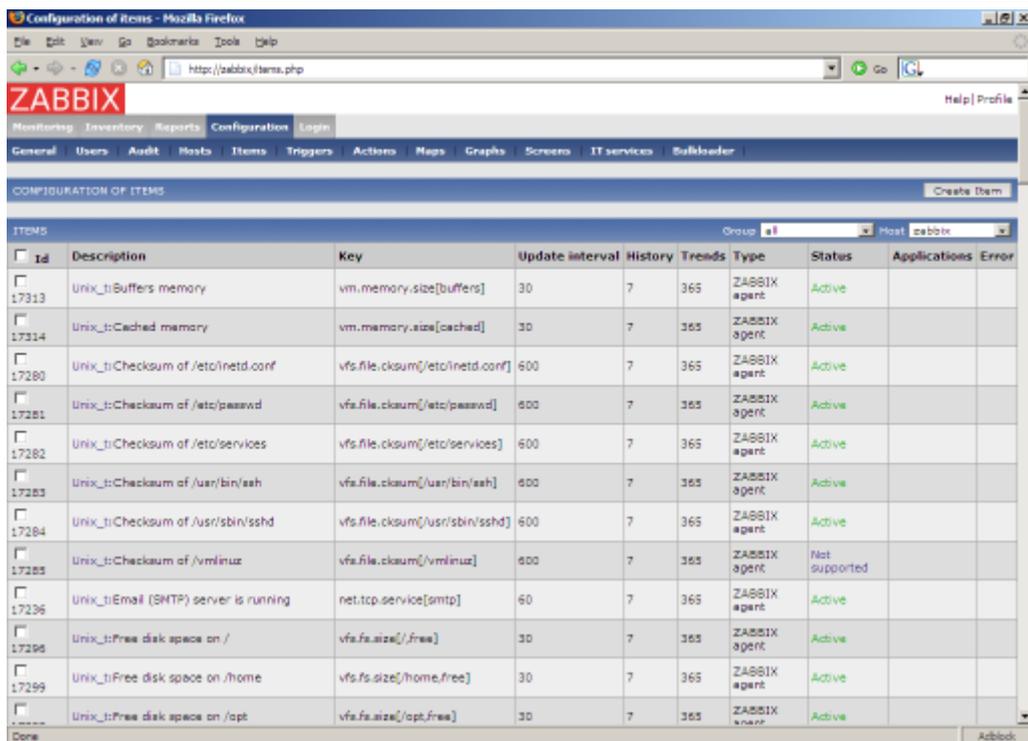
ホストが作成され、Unix_t に定義されているアイテムとトリガーがそのまま追加されます。



ホストのリストに戻り、追加したホストがリストに表示されていることを確認します。



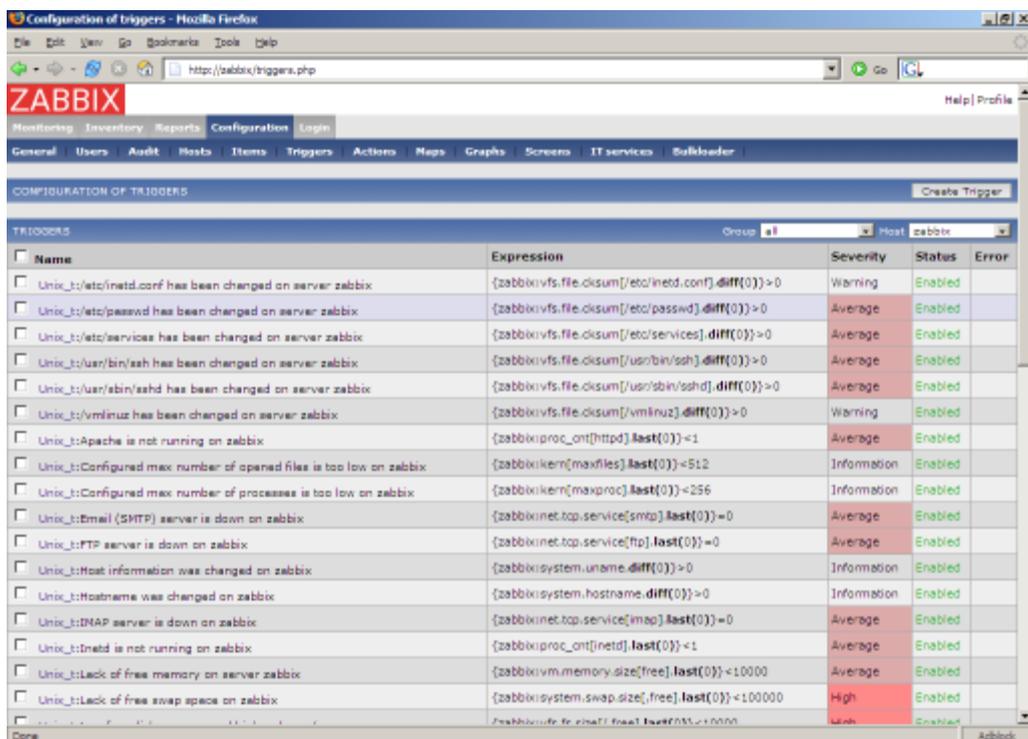
このホストに監視するアイテムが定義されているかどうかを確認します。[メニュー]->[設定]->[アイテム]を選択します。



The screenshot shows the 'Configuration of Items' page in Zabbix. The table below represents the data visible in the interface.

Id	Description	Key	Update interval	History	Trends	Type	Status	Applications	Error
17313	Unix::Buffers memory	vm.memory.size[buffers]	30	7	365	ZABBIX agent	Active		
17314	Unix::Cached memory	vm.memory.size[cached]	30	7	365	ZABBIX agent	Active		
17280	Unix::Checksum of /etc/inetd.conf	vfs.file.cksum[/etc/inetd.conf]	600	7	365	ZABBIX agent	Active		
17281	Unix::Checksum of /etc/passwd	vfs.file.cksum[/etc/passwd]	600	7	365	ZABBIX agent	Active		
17282	Unix::Checksum of /etc/services	vfs.file.cksum[/etc/services]	600	7	365	ZABBIX agent	Active		
17283	Unix::Checksum of /usr/bin/sash	vfs.file.cksum[/usr/bin/sash]	600	7	365	ZABBIX agent	Active		
17284	Unix::Checksum of /usr/sbin/sshd	vfs.file.cksum[/usr/sbin/sshd]	600	7	365	ZABBIX agent	Active		
17285	Unix::Checksum of /vmlinuz	vfs.file.cksum[/vmlinuz]	600	7	365	ZABBIX agent	Not supported		
17236	Unix::Email (SMTP) server is running	net.tcp.service[smtp]	60	7	365	ZABBIX agent	Active		
17296	Unix::Free disk space on /	vfs.fs.size[/,free]	30	7	365	ZABBIX agent	Active		
17299	Unix::Free disk space on /home	vfs.fs.size[/home,free]	30	7	365	ZABBIX agent	Active		
	Unix::Free disk space on /opt	vfs.fs.size[/opt,free]	30	7	365	ZABBIX agent	Active		

定義されているアイテムのリストが表示されます。次に、このホストにトリガーが定義されているかどうかを確認します。[メニュー]->[設定]->[トリガー]を選択します。



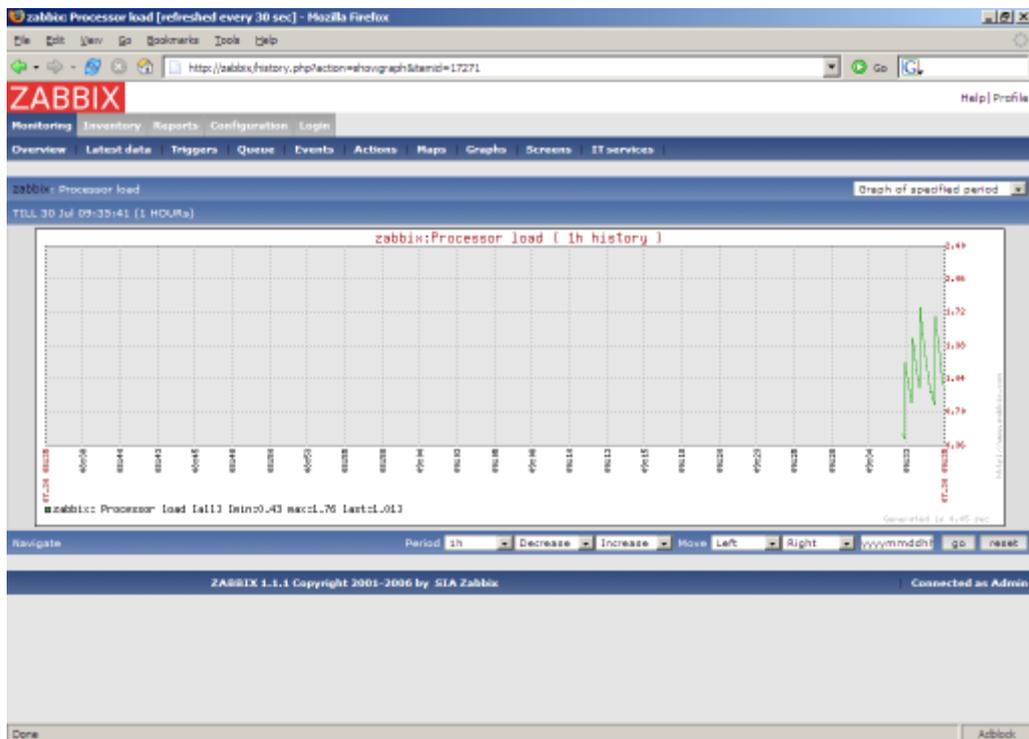
The screenshot shows the 'Configuration of Triggers' page in Zabbix. The table below represents the data visible in the interface.

Name	Expression	Severity	Status	Error
Unix::/etc/inetd.conf has been changed on server zabbix	{zabbixvfs.file.cksum[/etc/inetd.conf].diff(0)}>0	Warning	Enabled	
Unix::/etc/passwd has been changed on server zabbix	{zabbixvfs.file.cksum[/etc/passwd].diff(0)}>0	Average	Enabled	
Unix::/etc/services has been changed on server zabbix	{zabbixvfs.file.cksum[/etc/services].diff(0)}>0	Average	Enabled	
Unix::/usr/bin/sash has been changed on server zabbix	{zabbixvfs.file.cksum[/usr/bin/sash].diff(0)}>0	Average	Enabled	
Unix::/usr/sbin/sshd has been changed on server zabbix	{zabbixvfs.file.cksum[/usr/sbin/sshd].diff(0)}>0	Average	Enabled	
Unix::vmlinuz has been changed on server zabbix	{zabbixvfs.file.cksum[/vmlinuz].diff(0)}>0	Warning	Enabled	
Unix::Apache is not running on zabbix	{zabbixproc_ont[httpd].last(0)}<1	Average	Enabled	
Unix::Configured max number of opened files is too low on zabbix	{zabbixkern[maxfiles].last(0)}<512	Information	Enabled	
Unix::Configured max number of processes is too low on zabbix	{zabbixkern[maxproc].last(0)}<256	Information	Enabled	
Unix::Email (SMTP) server is down on zabbix	{zabbixnet.top.service[smtp].last(0)}=0	Average	Enabled	
Unix::FTP server is down on zabbix	{zabbixnet.top.service[ftp].last(0)}=0	Average	Enabled	
Unix::Host information was changed on zabbix	{zabbixsystem.uname.diff(0)}>0	Information	Enabled	
Unix::Hostname was changed on zabbix	{zabbixsystem.hostname.diff(0)}>0	Information	Enabled	
Unix::IMAP server is down on zabbix	{zabbixnet.top.service[imap].last(0)}=0	Average	Enabled	
Unix::Inetd is not running on zabbix	{zabbixproc_ont[inetd].last(0)}<1	Average	Enabled	
Unix::Lack of free memory on server zabbix	{zabbixvm.memory.size[free].last(0)}<10000	Average	Enabled	
Unix::Lack of free swap space on zabbix	{zabbixsystem.swap.size[free].last(0)}<100000	High	Enabled	

定義されているトリガーのリストが表示されます。次に、どのような情報を表示できるようになったかを調べます。[メニュー]->[最新データ]を選択します。

Description	Last check	Last value	Change	History
Buffers memory	30 Jul 09:35:00	222.25 MB	-	Graph
Cached memory	30 Jul 09:35:00	1.16 GB	+28 KB	Graph
Checksum of /etc/inetd.conf	30 Jul 09:32:55	1782859668	-	Graph
Checksum of /etc/passwd	30 Jul 09:32:54	1127549486	-	Graph
Checksum of /etc/services	30 Jul 09:32:54	2095574442	-	Graph
Checksum of /usr/bin/sah	30 Jul 09:32:55	890898524	-	Graph
Checksum of /usr/sbin/sahd	30 Jul 09:32:55	2842161037	-	Graph
Email (SMTP) server is running	30 Jul 09:35:00	1	-	Graph
Free disk space on /	30 Jul 09:35:00	5.36 MB	-	Graph
Free disk space on /home	30 Jul 09:35:00	5.36 MB	-	Graph
Free disk space on /opt	30 Jul 09:35:00	5.36 MB	-	Graph
Free disk space on /tmp	30 Jul 09:35:00	5.36 MB	-	Graph
Free disk space on /usr	30 Jul 09:35:00	5.36 MB	-	Graph
Free disk space on /var	30 Jul 09:35:00	5.36 MB	-	Graph
Free memory	30 Jul 09:35:00	165.25 MB	+732 KB	Graph
Free number of inodes on /	30 Jul 09:35:00	1741309	-	Graph
Free number of inodes on /home	30 Jul 09:35:00	1741309	-	Graph
Free number of inodes on /opt	30 Jul 09:35:00	1741309	-	Graph
Free number of inodes on /tmp	30 Jul 09:35:00	1741309	-	Graph
Free number of inodes on /usr	30 Jul 09:35:00	1741309	-	Graph
Free swap space	30 Jul 09:35:00	0 B	-	Graph
FTP server is running	30 Jul 09:35:00	0	-	Graph
Host information	30 Jul 09:32:54	Linux ubuntu 2.6.12- ...	-	History
Host name	30 Jul 09:32:55	ubuntu	-	History

次はグラフを表示します。[グラフ]をクリックします。



.. 最後にトリガーを表示します。[メニュー]->[トリガーのステータス]を選択します。

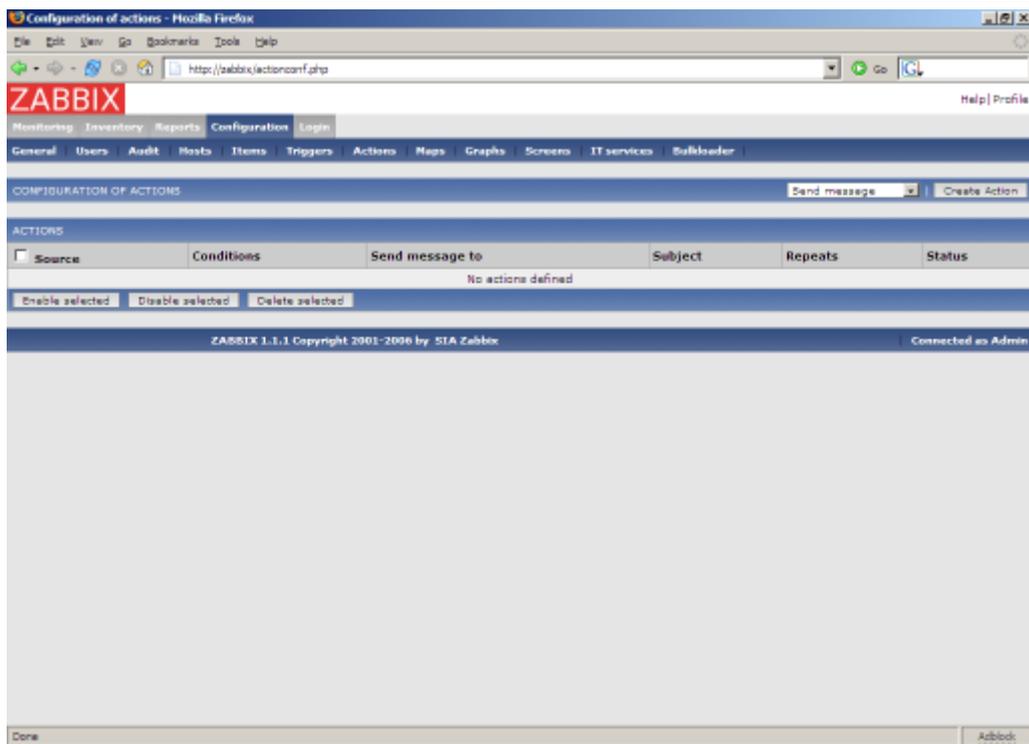
Name	Status	SEVERITY	Last change	Acknowledged	Comments
Lack of free swap space on zabbix	TRUE	High	30 Jul 09:32:55	No (Ack)	Add
FTP server is down on zabbix	TRUE	Average	30 Jul 09:32:55	No (Ack)	Add
IMAP server is down on zabbix	TRUE	Average	30 Jul 09:32:54	No (Ack)	Add
News (NNTP) server is down on zabbix	TRUE	Average	30 Jul 09:32:54	No (Ack)	Add
POP3 server is down on zabbix	TRUE	Average	30 Jul 09:32:55	No (Ack)	Add
Too many processes running on zabbix	TRUE	Average	30 Jul 09:37:00	No (Ack)	Add

追加したホストが ZABBIX によって監視されていることが確認できました。追加したホストに対して、以下の操作を実行できます。

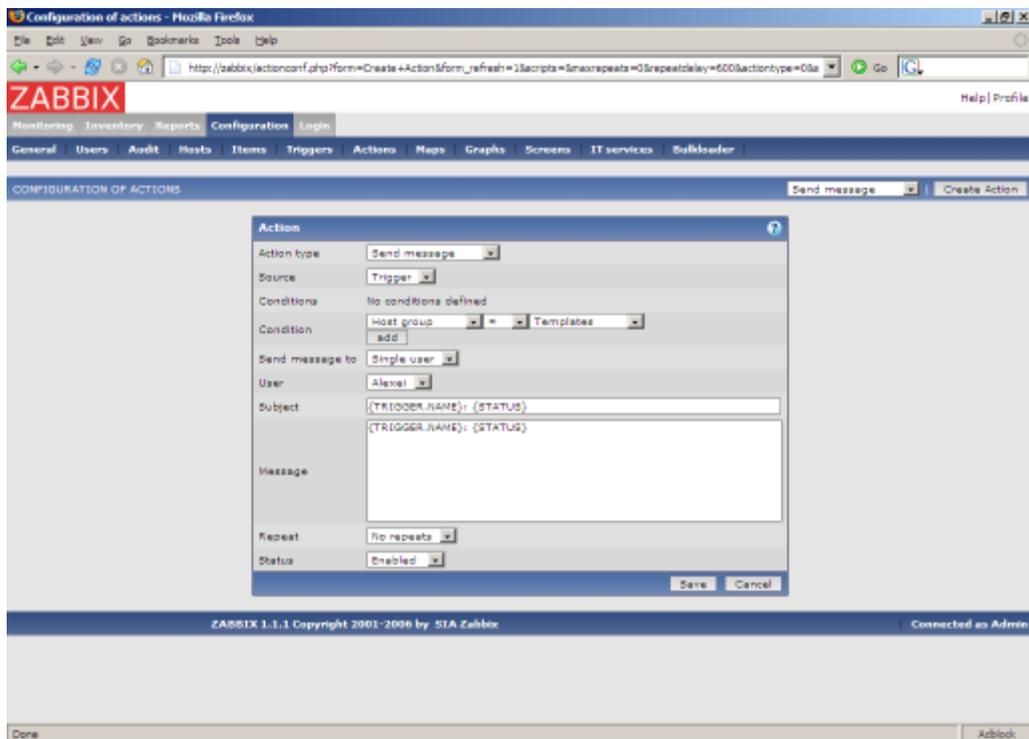
- 監視するアイテムのリストの変更
- トリガーアイテムのリストの変更
- アイテムの更新速度の調整
- ユーザ通知ルールの追加

5.5. 通知の設定

監視するホストを設定し、グラフやホストのステータスを表示できるようになりました。次に、基本的なメール通知を設定します。[メニュー]->[設定]->[アクション]を選択します。



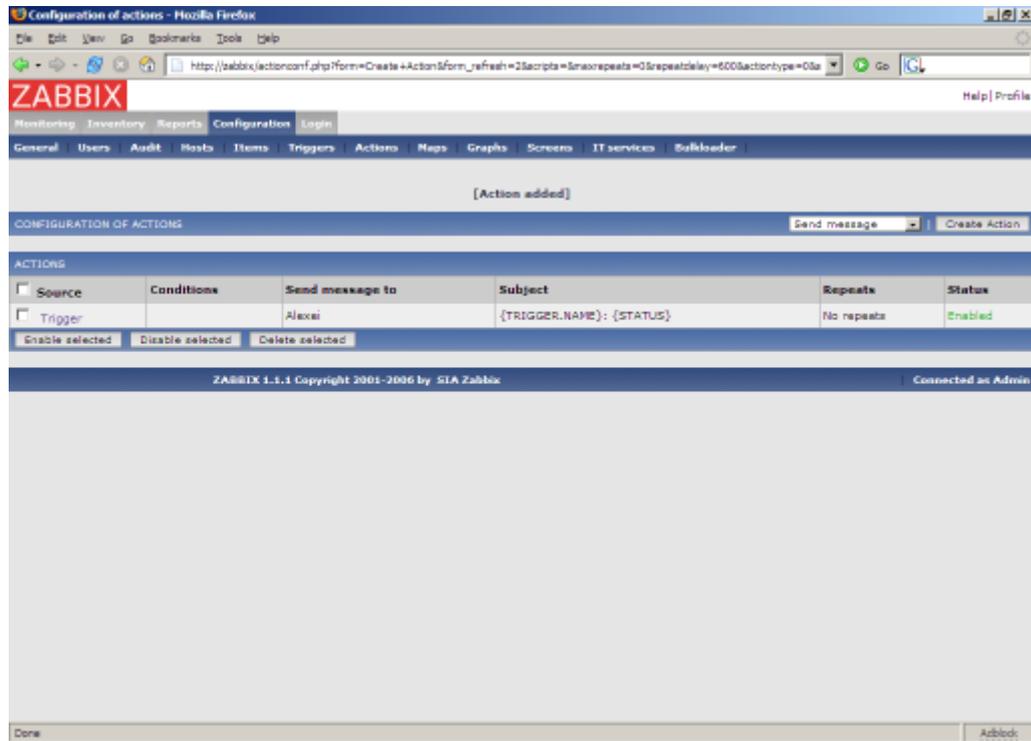
アクションはまだ定義されていません。[アクションの作成]をクリックします。



コンディションを指定しない場合、トリガーのステータスが変更されるたびにアクションがトリガーされます。

マクロ{TRIGGER.NAME}は、トリガー名で置き換えられます。マクロ{STATUS}は、トリガーの現在のステータスに応じて「ON」または「OFF」のどちらかに置き換えられます。

アクションは、選択したユーザまたはユーザグループにリンクされている全てのメディアに適用されます。



ここでは最も基本的な通知設定を示しましたが、以下の処理を設定することもできます。

- コンディションを使用して、通知を送信する際に適用する高度なフィルターを定義する
- 通知の繰り返し
- リモートコマンドの実行

6. XML のインポート/エクスポート

6.1. 目標

ZABBIX のインポート/エクスポート機能を使用すると、テンプレート、ホスト、アイテム、トリガー、およびグラフの設定パラメータを交換できます。

データは XML 形式でエクスポートされるので、読みやすく、変更も容易です。

- テンプレートの共有

ZABBIX ユーザ同士で設定パラメータを共有できます。

- サードパーティツールとの統合

広く普及している XML 形式を使用することで、サードパーティのツールやアプリケーションと統合したり、データをインポート/エクスポートすることができます。

6.2. 概要

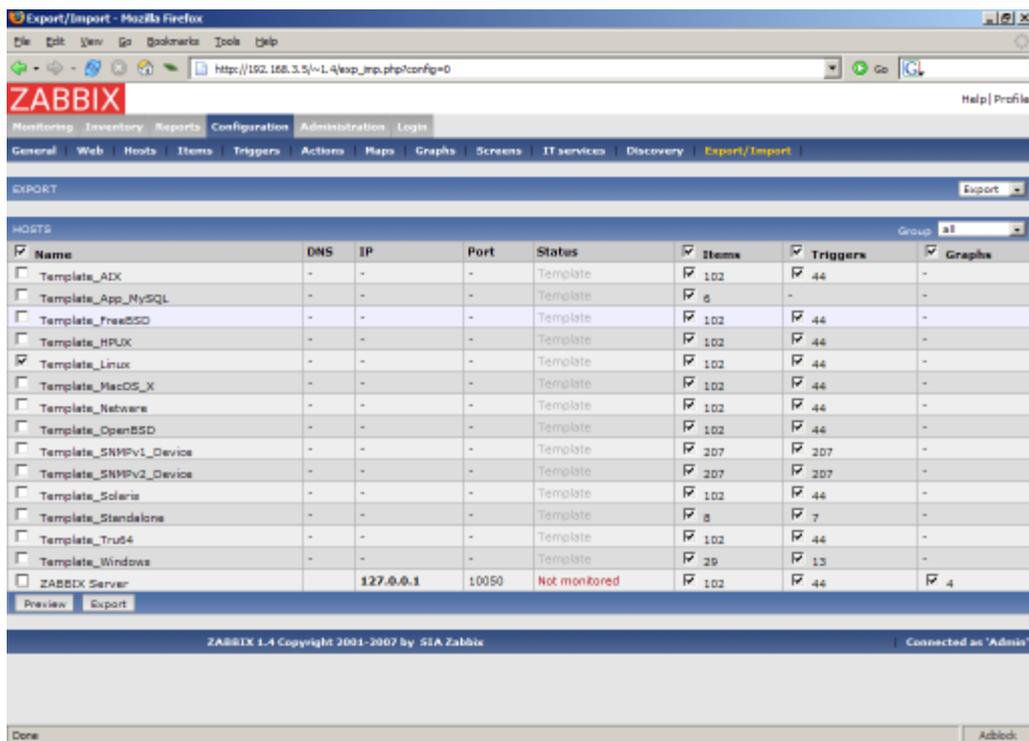
ZABBIX のインポート/エクスポートは、以下のデータを処理します。

- ホスト
- アプリケーション
- アイテム
- トリガー
- カスタムグラフ

6.3. データのエクスポート

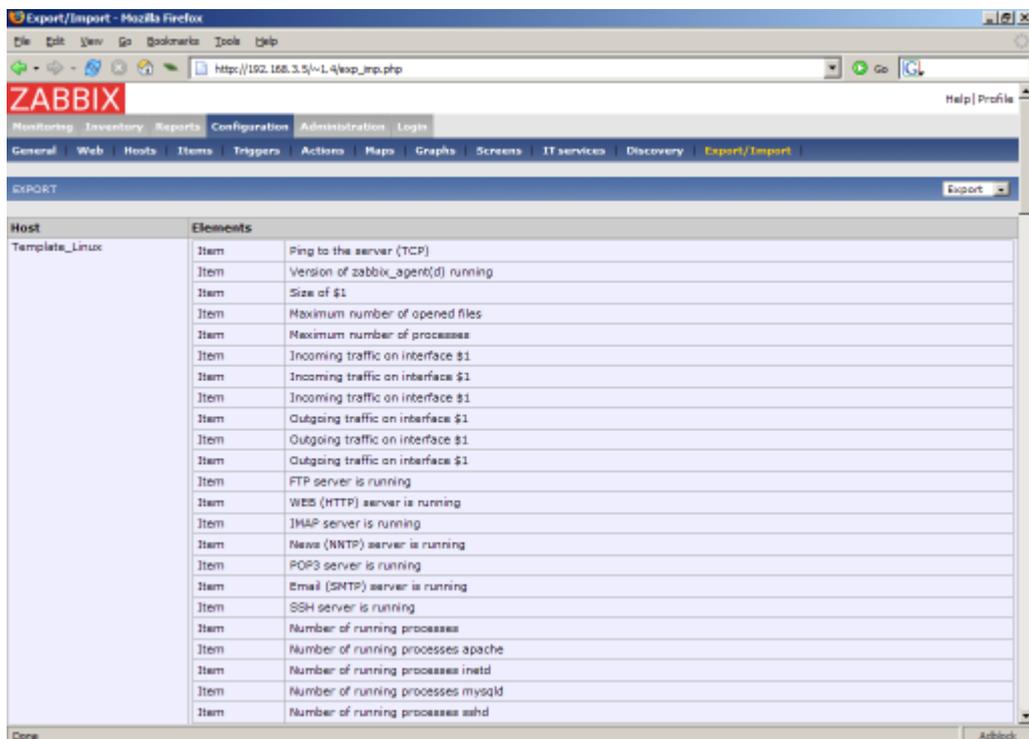
[メニュー]->[設定]->[エクスポート/インポート]を選択します。

ステップ 1	エクスポートする項目の選択
--------	---------------



ホスト「Template_Linux」のアイテムとトリガーを全て選択します。

[プレビュー]をクリックすると、エクスポートする項目のリストを表示できます。



ステップ 2 データのエクスポート

[エクスポート]をクリックすると、選択した項目が **zabbix_export.xml**(デフォルト名)という名前のローカル XML ファイルにエクスポートされます。

このファイルの内容(各タイプにつき項目を1つずつ)を以下に示します。

```
<?xml version="1.0"?>
<zabbix_export version="1.0" date="11.05.07" time="11.11">
  <hosts>
    <host name="ZABBIX Server">
      <useip>1</useip>
      <ip>127.0.0.1</ip>
      <port>10050</port>
      <status>1</status>
      <groups>
      </groups>
      <items>
        <item type="0" key="agent.ping" value_type="3">
          <description>Ping to the server (TCP)</description>
          <delay>30</delay>
          <history>7</history>
          <trends>365</trends>
          <snmp_port>161</snmp_port>
          <valuemap>Service state</valuemap>
          <applications>
            <application>General</application>
          </applications>
        </item>
        ....
      </items>
      <triggers>
        <trigger>
          <description>Version of zabbix_agent(d) was changed on
{HOSTNAME} </description>
          <expression>{{HOSTNAME}:agent.version.diff(0)}>0</expression>
          <priority>3</priority>
        </trigger>
        ....
      <graphs>
        <graph name="CPU Loads" width="900" height="200">
          <show_work_period>1</show_work_period>
          <show_triggers>1</show_triggers>
          <yaxismin>0.0000</yaxismin>
          <yaxismax>100.0000</yaxismax>
          <graph_elements>
            <graph_element
item="{HOSTNAME}:system.cpu.load[,avg15]">
              <color>990000</color>
              <yaxisside>1</yaxisside>
              <calc_fnc>2</calc_fnc>
              <periods_cnt>5</periods_cnt>
            </graph_element>
            <graph_element
item="{HOSTNAME}:system.cpu.load[,avg1]">
              <color>009900</color>
              <yaxisside>1</yaxisside>
              <calc_fnc>2</calc_fnc>
              <periods_cnt>5</periods_cnt>
            </graph_element>
            <graph_element
item="{HOSTNAME}:system.cpu.load[,avg5]">
```

```

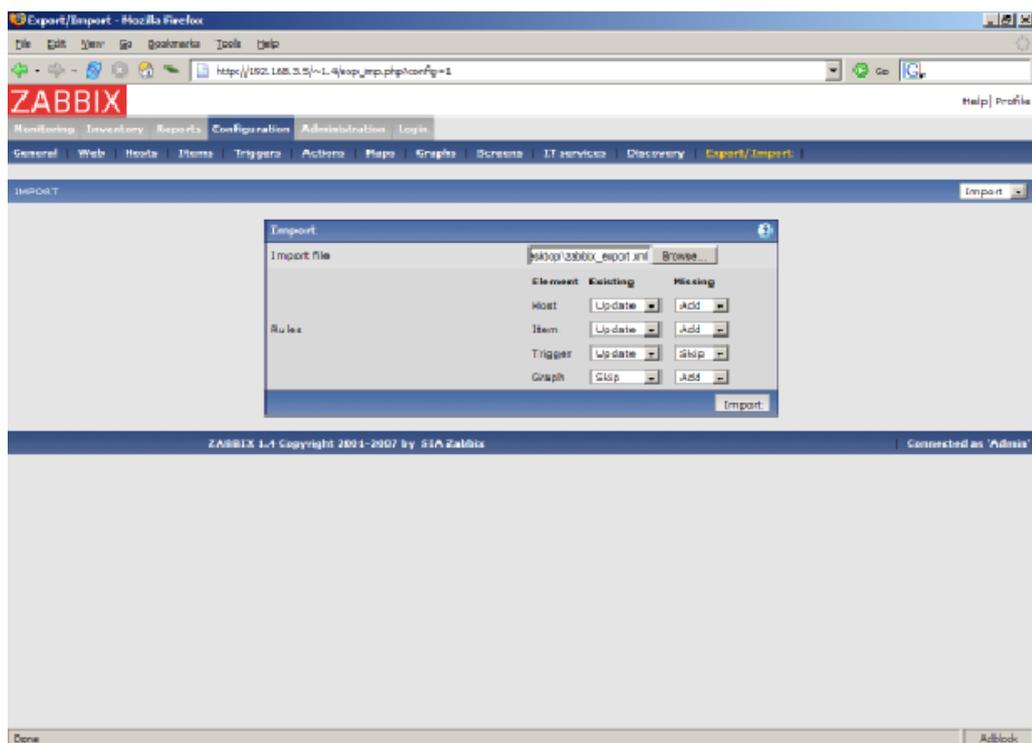
        <color>999900</color>
        <yaxisside>1</yaxisside>
        <calc_fnc>2</calc_fnc>
        <periods_cnt>5</periods_cnt>
    </graph_element>
</graph_elements>
</graph>
    ....
</graphs>
</host>
    ....
</hosts>
</zabbix_export>

```

6.4. データのインポート

[メニュー]->[設定]->[エクスポート/インポート]を選択します。

ステップ 1 データのインポートのパラメータを設定して[インポート]をクリック



以下のアイテムパラメータに注意してください。

パラメータ	説明
[ファイルをインポート]	XMLファイルのファイル名。

[ルール]

[項目]はXMLファイルの項目を定義します。

[更新]パラメータで**[存在する場合]**を選択すると、インポートによってファイルから読み込まれた値で既存の項目が更新されます。それ以外の値を選択した場合は、インポートしても既存の項目は更新されません。

[追加]パラメータで**[存在しない場合]**を選択すると、インポートによってファイルから読み込まれた値を使用して新しい項目が追加されます。それ以外の値を選択した場合は、インポートしても新しい項目は追加されません。

7. チュートリアル

ここでは、頻繁に実行する作業の詳細な手順について説明します。

7.1. ZABBIX エージェントの拡張

ここでは、ZABBIX エージェントを機能拡張する詳細な手順について説明します。

ステップ 1 必要なパラメータを取得するためのスクリプトまたはコマンドの作成

MySQL サーバが実行したクエリの合計数を取得するためのコマンドの例を以下に示します。

```
mysqladmin -uroot status|cut -f4 -d":"|cut -f1 -d"S"
```

このコマンドを実行すると、SQL クエリの合計数が出力されます。

ステップ 2 作成したコマンドをエージェントの設定ファイルに追加

ステップ 1 で作成したコマンドを `zabbix_agentd.conf` に追加します。

```
UserParameter=mysql.questions,mysqladmin -uroot status|cut -f4 -d":"|cut -f1 -d"S"
```

`mysql.questions` は一意な識別子です。任意の文字列(「queries」など)を指定できます。

以下のコマンドを実行して、このパラメータをテストします。

```
zabbix_agentd -t mysql.questions
```

ステップ 3 ZABBIX エージェントのリスタート

エージェントによって設定ファイルがリロードされます。

ステップ 4 監視するアイテムの追加

監視するホストに新しくキー「`mysql.questions`」のアイテムを追加します。アイテムのタイプには、`[ZABBIX エージェント]`または`[ZABBIX エージェント(アクティブ)]`のどちらかを指定する必要があります。

ZABBIX サーバで、戻り値のデータ型を正しく設定する必要があることに注意してください。

間違えて設定すると、ZABBIX がデータを受け取ることができません。

7.2. ログファイルの監視

ここでは、ログファイルの監視を設定する詳細な手順について説明します。ホストは ZABBIX フロントエンドで設定済みであるものとします。

ステップ 1 ZABBIX エージェントの設定

標準の手順に従って、監視するホストにエージェントをインストールおよび設定します。パラメータ Hostname は、ZABBIX フロントエンドで設定するホストの名前と一致させる必要があります。

また、zabbix_agentd.conf にパラメータ DisableActive が設定されていないことを確認します。

ステップ 2 ログファイル監視用の新しいアイテムを追加

以下のアイテムパラメータに注意してください。

パラメータ	説明
[タイプ]	[ZABBIXエージェント(アクティブ)]を設定します。
[キー]	「log[file<,regexp>]」を設定します。 例: log[/var/log/syslog]、log[/var/log/syslog,error] ユーザzabbixにはこのファイルに対する読み取りパーミッションが必要です。このパーミッションを付与しないと、アイテムのステータスが取得不可になります。 regexpが指定されている場合、ZABBIXエージェントはログファイルのエントリをregexpの正規表現でフィルタリングします。
[データ型]	[ログ]を設定します。
[更新間隔(秒)]	ログファイルが変更されたかどうかをZABBIXエージェントがチェックする間隔を設定します。通常は、できるかぎり早く変更を検出できるように1秒に設定します。

7.3. リモートアクション

ここでは、イベント発生時にあらかじめ定義したコマンドをリモート実行するように設定する詳細な手順について説明します。ZABBIX は設定済みで、動作しているものとします。

ステップ 1 新しいアクションの設定

標準の手順に従って、監視するホストにエージェントを設定します。

以下のアクションパラメータに注意してください。

パラメータ	説明
[オペレーションのタイプ]	[リモートコマンド]を設定します。
[リモートコマンド]	1行に1つずつリモート実行するコマンドを指定します。 例: host:/etc/init.d/apache restart zabbix_agentd.confでEnableRemoteCommandsに1を設定する必要があります。 リモートコマンドにはマクロも使用できます。

リモートコマンドの構文を以下に示します。

リモートコマンド	説明
<host>:<command>	コマンド「command」はホスト「host」上で実行されます。
<group>#<command>	コマンド「command」はホストグループ「group」に属する全てのホスト上で実行されます。

IPMI リモートコマンドの構文を以下に示します。

リモートコマンド	説明
<host>:IPMI <ipmicontrol> [value]	この構文は、単一のホストでIPMIコマンドを実行するためのものです。 サポートされているIPMIコントロール: 「reset」、「power」 サポートされている値: 「on」、「off」、または数値(デフォルトは1) 例: サーバのリスタート: host:IPMI reset on サーバのリブート: host:IPMI power off
<group>#IPMI <ipmicontrol> [value]	この構文は、ホストグループの全てのホストに対してIPMIコマンドを実行するためのものです。

重要:

設定したコマンドを実行するパーミッションがユーザ zabbix に付与されていることを確認する必要があります。

sudo を使用して特権コマンドを実行できるようにする方法もあります。

ZABBIX エージェントは、コマンドをバックグラウンドで実行します。

ZABBIX は、コマンドが正常に実行されたかどうかをチェックしません。

例 1 特定の条件が成立したときに Windows をリスタートするアクションの設定例

ZABBIX が問題を検出した場合に自動的に Windows をリスタートするには、以下のアクションを定義します。

パラメータ	説明
[オペレーションのタイプ]	[リモートコマンド]を設定します。
[リモートコマンド]	host:c:¥windows¥system32¥shutdown.exe -r -f 「host」はWindowsサーバのZABBIXホスト名に置き換えます。

7.4. Windows サービスの監視

ここでは、Windows サービスの監視を設定する詳細な手順について説明します。ZABBIX サーバと ZABBIX エージェントは設定済みで、動作しているものとします。

ステップ 1 サービス名の取得

サービス名を取得するには、サービスの MMC(Microsoft 管理コンソール)に移動して、ステータス(開始または停止)を監視するサービスのプロパティを表示します。[全般]タブに[サービス名]というフィールドがあります。その値をカッコ内に記述します。たとえば、「Workstation」サービスを監視する場合、サービス名は **lanmanworkstation** になります。

ステップ 2 サービス監視用のアイテムの追加

[キー]に「**service_state[lanmanworkstation]**」、[データ型]に「**数値(整数)**」、[値のマッピングの使用]に「**Windows service state**」を設定してアイテムを追加します。

8. エスカレーションと繰り返し通知

8.1. 目標

ZABBIX エスカレーションの目標を以下に示します。

- 問題(トリガーが障害状態)が存在する場合にユーザに定期的に通知
- 繰り返し通知
- 他のユーザグループへの問題のエスカレーション
- 問題の存続時間がN時間(秒、分など)を超える場合にアクション(通知とリモートコマンドの両方)を実行
- 全ての関係者へのリカバリメッセージ
- コンディションに基づくエスカレーション(認知されたイベントはエスカレートしない、など)

8.2. 概要

ZABBIX には、エスカレーションと繰り返し通知のための効果的で非常に柔軟な機能が備わっています。設定に応じて、未解決の問題と各エスカレーションステップに割り当てられている実行アクションが自動的にエスカレートされます(エスカレーションステップが増分されます)。

9. ウェブ監視

9.1. 目標

ZABBIX ウェブ監視の目標を以下に示します。

- ウェブアプリケーションのパフォーマンスの監視
- ウェブアプリケーションの可用性の監視
- HTTP と HTTPS のサポート
- 複数ステップで構成される複雑なシナリオ(HTTP 要求)のサポート

9.2. 概要

ZABBIX ウェブ監視機能は、効率的に動作し、非常に柔軟に設定できます。ウェブシナリオを定期的に行って、収集したデータをデータベースに保存します。保存されたデータはグラフ、トリガー、および通知に反映されます。

ウェブシナリオの各ステップで収集される情報を以下に示します。

- 応答時間
- ダウンロード速度(1 秒あたり)
- 応答コード

ZABBIX は、あらかじめ指定された文字列が、取得した HTML ページに含まれているかどうかチェックします。

ZABBIX ウェブ監視は、HTTP と HTTPS をサポートします。

9.3. ウェブシナリオ

シナリオは複数の HTTP 要求(ステップ)で構成され、ZABBIX サーバによって定期的に行われます。通常は、ウェブアプリケーションの機能の特定部分に対して 1 つのシナリオを定義します。シナリオを使用すると、ユーザエクスペリエンスを非常に簡単に監視できるようになります。

ウェブシナリオは、ホストアプリケーションにリンクしてグループ化します。

ウェブシナリオは定期的に行われ、1 つまたは複数のステップで構成されます。

1 つのシナリオの実行中は、Cookie は全て保存されます。

例 1 ZABBIX GUI の監視

ZABBIX GUI の可用性とパフォーマンスを監視する場合、まずログインしてから[概要]スクリーンと[トリガー]スクリーンの応答速度を調べて、最後にログアウトする必要があります。

このシナリオは以下のステップに分解できます。

1. ログイン

2. [概要]スクリーンに移動
3. [トリガー]スクリーンに移動
4. ログアウト

いずれかのステップの実行に失敗すると、シナリオの実行は異常終了します。

パラメータ	説明
[アプリケーション]	ウェブシナリオをリンクするアプリケーションを指定します。指定できるのは既存のアプリケーションだけです。 例: ZABBIXサーバ
[名前]	ウェブシナリオの名前。 [監視データ]->[ウェブ]を選択すると、この名前が表示されます。 例: ZABBIX GUI
[更新間隔(秒)]	シナリオの更新頻度(秒単位)を指定します。 例: 60
[エージェント]	ZABBIXがユーザエージェントとして使用するブラウザを指定します。ウェブブラウザごとに異なるコンテンツを生成するウェブサイトを監視する場合に使用します。 例: Opera 9.02 on Linux
[ステータス]	[有効]: シナリオは有効で、実行されます。 [無効]: シナリオは無効で、実行されません。
[変数]	ステップを設定する際に使用するマクロのリストです。 構文: {macro}=value マクロ{macro}は、ステップのURLやPostの変数で置き換えられます。 例: {user}=guest {password}=guest
[ステップ]	シナリオのステップ

シナリオを作成すると、以下の監視用アイテムが自動的に追加され、選択したアプリケーションにリンクされます。「Scenario」は、実際のシナリオ名で置き換えられます。

アイテム	説明
Download speed for scenario 'Scenario'	シナリオ全体、すなわち全ステップの平均の、ダウンロード速度(1秒あたりのバイト数)に関する情報を収集するアイテムです。 アイテムのキー: web.test.in[Scenario,,bps] データ型: 浮動小数点

Failed step of scenario 'Scenario'	シナリオのステップのうち失敗したステップの数をカウントするアイテムです。全てのステップが正常に実行された場合、0を返します。 アイテムのキー: web.test.fail[Scenario] データ型: 整数
---	--

上記のアイテムは、トリガーの作成および通知コンディションの定義に使用できます。

例 1 ウェブシナリオの異常終了を検出するトリガー

このトリガー条件式は、「{host: web.test.fail[Scenario]}.last(0)#0」と定義することができます。
「Scenario」は実際のシナリオ名で置き換えることを忘れないでください。

例 2 ウェブアプリケーションのパフォーマンス低下を検出するトリガー

このトリガー条件式は、「{host:web.test.in[Scenario,,bps]}.last(0)<10000」と定義することができます。

「Scenario」は実際のシナリオ名で置き換えることを忘れないでください。

9.4. ウェブステップ

ステップとは、簡単に言えば1つのHTTP要求のことです。ステップの実行順序はあらかじめ指定しておきます。

パラメータ	説明
[名前]	ステップの名前。 例: Login
[URL]	URL。 例: www.zabbix.com
[POST]	HTTP POST変数(存在する場合に指定)。 例: id=2345&userid={user} {user}がウェブシナリオのマクロとして定義されている場合、ステップ実行時にその値で置き換えられます。 置き換えられた情報がそのまま送信されます。
[タイムアウト]	ステップは、ここで指定した時間(秒単位)以内に完了する必要があります。実際には、このパラメータが定義するのは、URLへの接続にかかる最大時間およびHTTP要求の処理にかかる最大時間です。 したがって、ZABBIXは1つのステップでここで指定した時間の 2倍の秒数以内 に処理を完了する必要があります。 例: 15
[要求文字列]	文字列(Posix形式の正規表現で指定)は取得したコンテンツ内に含まれている必要があります。含まれない場合、このステップは失敗します。何も指定しない場合、このステップは常

	に成功します。 例: Homepage of ZABBIX
[ステータスコード]	成功とみなすHTTPステータスコードのリストを指定します。 リストに含まれていないステータスコードが返された場合、このステップは失敗します。 何も指定しない場合、このステップは常に成功します。 例: 200、210

ステップを作成すると、以下の監視用アイテムが自動的に追加され、選択したアプリケーションにリンクされます。「Scenario」と「Step」はそれぞれ、実際のシナリオ名とステップ名で置き換えられます。

アイテム	説明
Download speed for step 'Step' of scenario 'Scenario'	ステップのダウンロード速度(1秒あたりのバイト数)に関する情報を収集するアイテムです。 アイテムのキー: web.test.in[Scenario,Step,bps] データ型: 浮動小数点
Response time for step 'Step' of scenario 'Scenario'	ステップの応答時間(秒単位)に関する情報を収集するアイテムです。 アイテムのキー: web.test.time[Scenario,Step] データ型: 浮動小数点
Response code for step 'Step' of scenario 'Scenario'	ステップの応答コードを収集するアイテムです。 アイテムのキー: web.test.rspcode[Scenario,Step] データ型: 整数

上記のアイテムは、トリガーの作成および通知コンディションの定義に使用できます。

例 1 ZABBIX GUI のログインに時間がかかりすぎることを検出するトリガー

このトリガー条件式は、「{zabbix: web.test.time[ZABBIXGUI,Login]}.last(0)>3」と定義することができます。

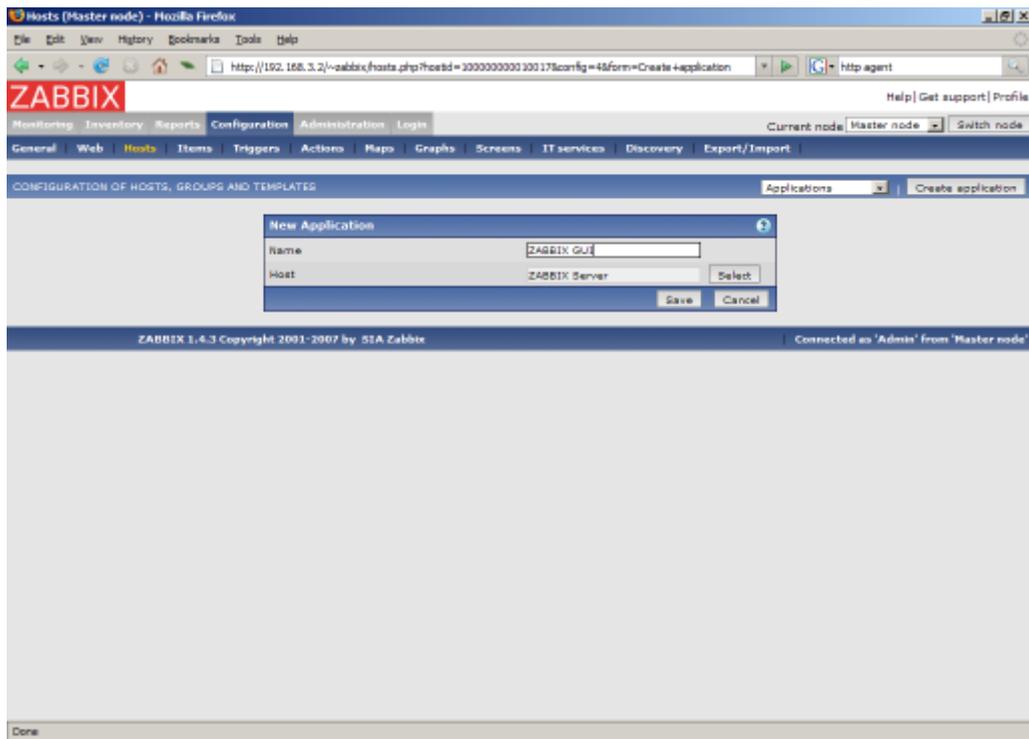
9.5. 実際に使用するシナリオ

ここでは、ZABBIX ウェブ監視機能を使用して、ZABBIX ウェブインタフェースを監視する手順について説明します。具体的には、その可用性、生成するコンテンツの正当性、および応答速度を監視します。

まず、ユーザ名とパスワードを使用してログインし、[設定]->[一般設定]ページに移動します。

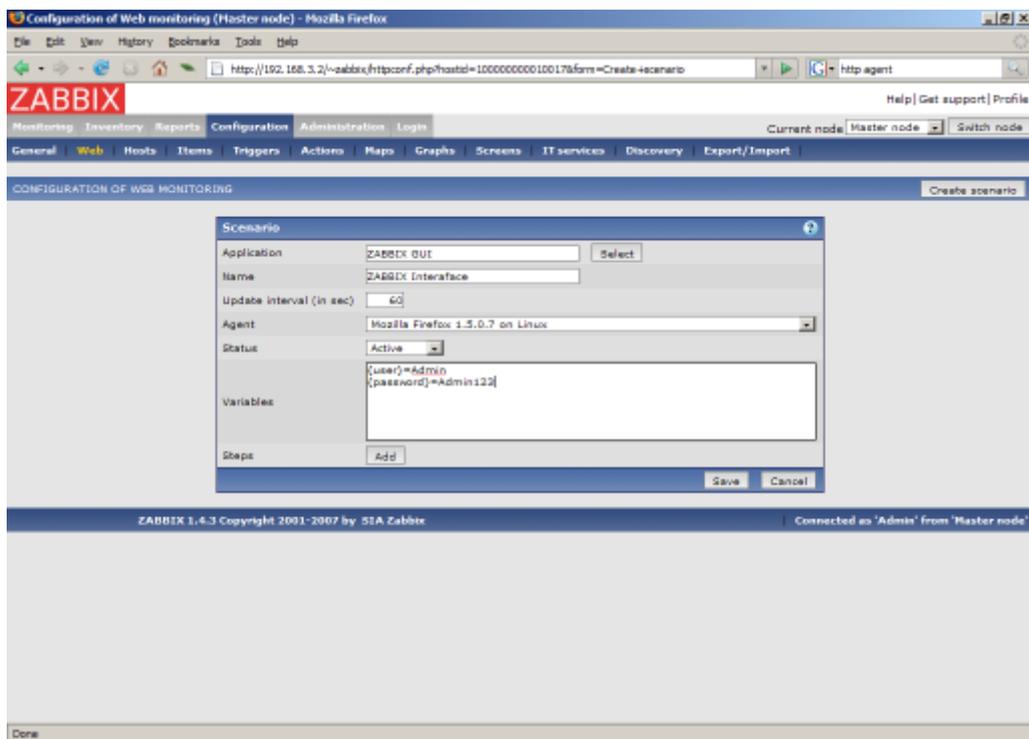
ステップ 1 新しいホストアプリケーションの追加

すでに適切なアプリケーションが動作している場合は、このステップを実行する必要はありません。ホストが存在しない場合は、ホストの作成も行う必要があります。



ステップ 2 新しいウェブシナリオの追加

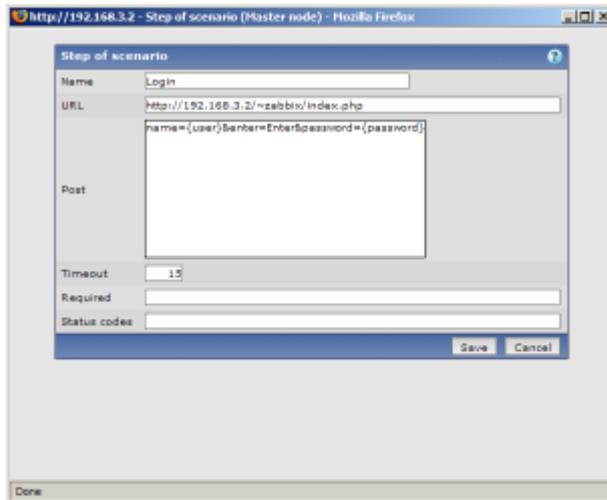
ZABBIX ウェブインタフェースを監視するための新しいシナリオを追加します。このシナリオは複数のステップから構成されます。



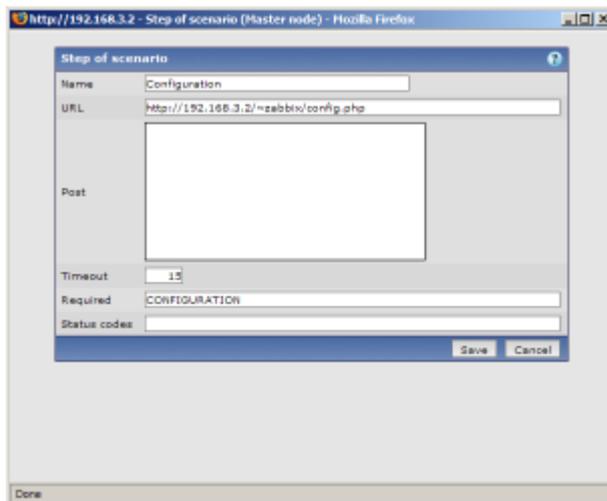
また、{user}と{password}の2つのマクロも作成します。

ステップ 3 シナリオのステップの定義

監視するためのステップを追加します。

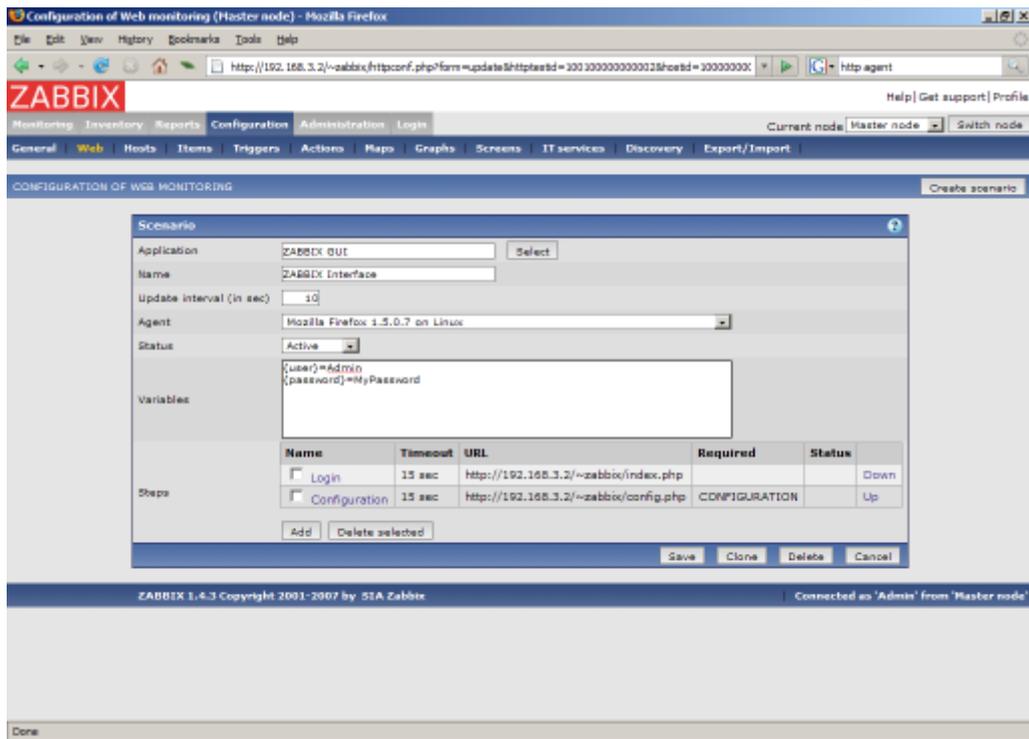


シナリオのステップ 1 です。{user}と{password}の2つのマクロを使用します。

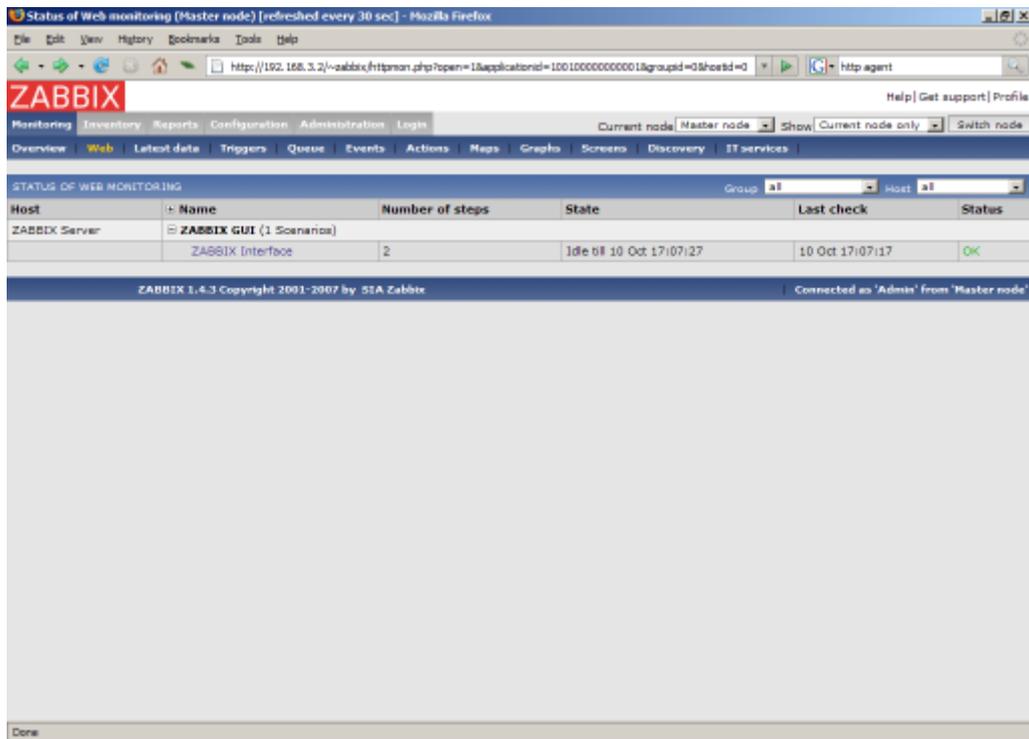


シナリオのステップ 2 です。

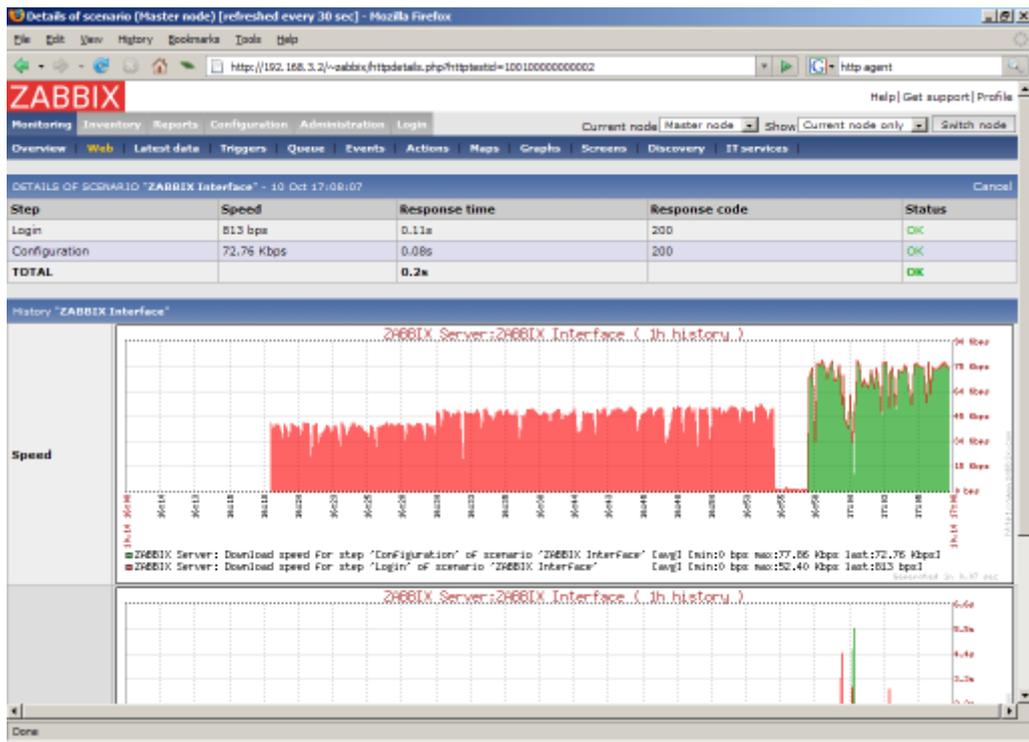
ステップ 4 シナリオの保存



[監視データ]->[ウェブ]を選択すると、アプリケーションとリンクされているシナリオのリストが表示されます。



シナリオをクリックすると、統計情報が表示されます。



10. ログファイルの監視

10.1. 概要

ZABBIX でログファイルを集中的に監視し、分析することができます。

ログファイルに特定の文字列または文字列パターンが追加されたときに、通知を使用してユーザに警告することができます。

10.2. ログファイルを監視するメカニズム

ログファイルを監視するには、ホスト上でZABBIXエージェントが動作している必要があります。また、ログファイルの監視に使用するアイテムには、[タイプ]に[ZABBIX エージェント(アクティブ)]、[データ型]に[ログ]、[キー]に「log[path to log file<,pattern>]」を設定する必要があります。

重要:

- サーバとエージェントは、監視するログのサイズをカウンタで追跡します。
- エージェントがログファイルを読み込む場合、前回の続きから読み始めます。
- エージェントがログファイルを前回の続きから読み始めることができるように、分析済みのバイト数(カウンタ)が ZABBIX データベースに保存され、エージェントに送信されます。
- エージェントが受信した分析済みバイト数よりもログファイルのサイズが小さくなった場合は、カウンタは 0 にリセットされ、エージェントはログファイルの先頭から読み始めます。
- ZABBIX エージェントは、[リフレッシュ期間(Refresh period)]に設定された時間間隔(秒単位)でログファイルに新しく追加されたレコードを処理します。
- ZABBIX エージェントが 1 秒あたりに送信するログファイルの最大行数は 10 行です。これはネットワークや CPU の各リソースが過負荷になることを防ぐための制限です。

11. オートディスカバリ

11.1. 目標

ZABBIX オートディスカバリモジュールには、様々な目標があります。

- 展開の単純化
オートディスカバリを使用すると、ZABBIX の展開が大幅に単純化され、作業時間が短縮されます。また、ユーザフレンドリなアプライアンスを作成できます。
- 管理の単純化
オートディスカバリを適切に設定することで、ZABBIX システムの管理を大幅に単純化できます。
- 常に変化する環境に対応
オートディスカバリを使用すると、常に変化する環境でも少ない管理コストで ZABBIX を使用できます。

11.2. 概要

ZABBIX オートディスカバリ機能は、効率的に動作し、非常に柔軟に設定できます。オートディスカバリは、以下の情報を使用します。

- IP アドレスの範囲
- 外部サービス(FTP、SSH、ウェブ、POP3、IMAP、TCP など)の可用性
- ZABBIX エージェントから受信する情報
- SNMP エージェントから受信する情報

オートディスカバリでは、以下の情報は検出しません。

- ネットワークポートの検出

ZABBIX オートディスカバリモジュールがサービスやホスト(IP アドレス)を検出するたびにイベントが生成されます。このイベントを使用して、以下のアクションを実行するためのルールを作成できます。

- ユーザ通知の生成
- ホストの追加と削除
- テンプレートへのホストの追加
- テンプレートからのホストの削除
- テンプレートへのホストのリンク
- テンプレートからのホストのリンクの削除
- リモートスクリプトの実行

アクションを設定する際、ホストまたはサービスのアップタイム/ダウンタイムを考慮することができます。

11.3. オートディスカバリのメカニズム

オートディスカバリは、基本的にディスカバリとアクションの2段階で構成されます。

まずホストまたはサービスを検出し、ディスカバリイベントまたはいくつかのイベントを生成します。

次にイベントを処理して、検出されたデバイス、IP、ステータス、アップタイム/ダウンタイムなどのタイプに応じて特定のアクションを実行します。

11.3.1. ディスカバリ

ZABBIXは定期的にオートディスカバリルールに定義されているIPアドレスの範囲をスキャンします。その頻度は、ルールごとに個別に設定します。

各ルールには、IPアドレスの範囲に対して実行するサービスチェックを定義します。

オートディスカバリモジュールによって生成されるイベントのソースは[ディスカバリ]になります。

ZABBIXでは、以下のイベントが生成されます。

イベント	生成される状況
サービスアップ	ZABBIXでアクティブなサービスが検出されたとき(毎回)
サービスダウン	ZABBIXでサービスを検出できないとき(毎回)
ホストアップ	IPに対して少なくとも1つのサービスがアップ状態の場合
ホストダウン	全てのサービスから応答がない場合
サービス検出	サービスがダウンタイム後に復帰したか、初めて検出された場合
サービス喪失	アップ状態のサービスが喪失した場合
ホスト検出	ホストがダウンタイム後に復帰したか、初めて検出された場合
ホスト喪失	アップ状態のホストが喪失した場合

11.3.2. アクション

オートディスカバリによって生成されるイベントに対して使用可能な全てのコンディションの説明については、「アクションコンディション」を参照してください。

オートディスカバリによって生成されるイベントに対して実行可能な全てのオペレーションの説明については、「オペレーション」を参照してください。

11.4. オートディスカバリルール

ZABBIXは、オートディスカバリルールを使用してホストとサービスを検出します。

オートディスカバリルールのパラメータを以下に示します。

パラメータ	説明
[名前]	ルールの名前。例: Local network
[IPアドレスの範囲]	チェックするIPアドレスの範囲。以下に示すいずれかの形式で指定できます。 単一IPアドレス: 192.168.1.33

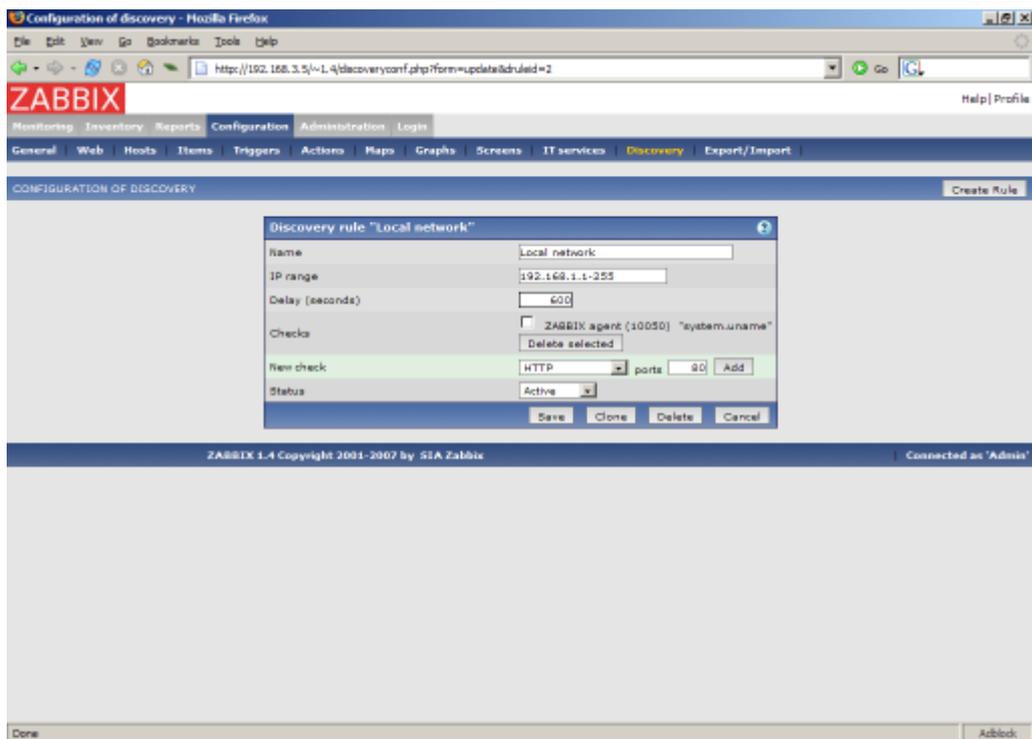
	IPアドレスの範囲: 192.168.1.1-255 リスト: 192.168.1.1-255,192.168.2.1-100,192.168.2.200
[間隔 (seconds)]	このルールの実行頻度を定義します。
[チェック]	ホストとサービスを検出するためのチェック方法のリスト。 サポートされているチェックのリスト: SSH、LDAP、SMTP、FTP、HTTP、POP、NNTP、IMAP、TCP、ZABBIXエージェント、SNMPv1エージェント、SNMPv2エージェント パラメータポートは以下のいずれかの形式で指定できます。 単一ポート: 22 ポートの範囲: 22-45 リスト: 22-45,55,60-70
[ステータス]	[有効] - ルールは有効で、ZABBIXサーバによって実行されます。 [無効] - ルールは無効で、実行されません。

11.5. 実際に使用するシナリオ

ここでは、ローカルネットワークの 192.168.1.1~192.168.1.255 の範囲の IP アドレスをチェックするオートディスカバリを設定する手順について説明します。このシナリオでは、以下の設定を行います。

- ZABBIX エージェントが動作しているホストのみ検出する
- チェックは 10 分間隔で実行する
- アップタイムが 1 時間以上のホストは監視対象に追加する
- ダウンタイムが 24 時間以上のホストは監視対象から削除する
- Windows ホストでは Template_Windows を使用する
- Linux ホストでは Template_Linux を使用する
- Linux ホストはグループ「Linux servers」に追加する
- Windows ホストはグループ「Windows servers」に追加する

ステップ 1 IP アドレスの範囲に対するオートディスカバリルールの定義



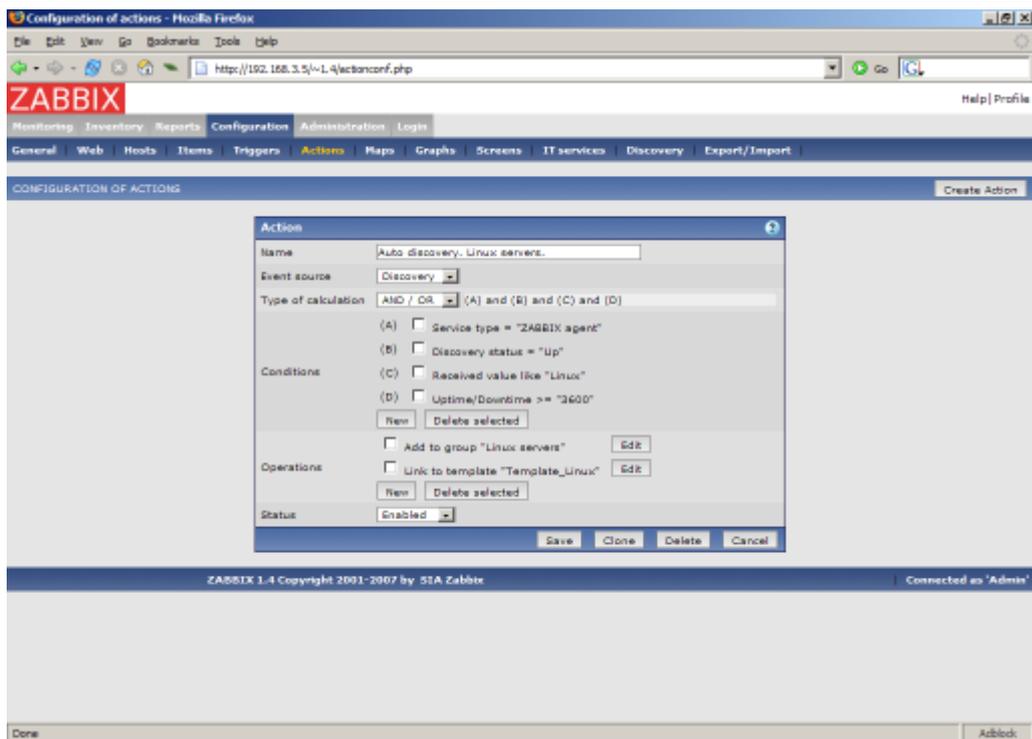
ZABBIXは、ZABBIXエージェントに接続してsystem.unameを取得することによって、192.168.1.1～192.168.1.255の範囲のIPアドレスのホストを検出しようと試みます。エージェントから受信した値を使用して、オペレーティングシステムごとに異なるアクションを適用できます。

たとえば、WindowsマシンはWindows_Templateに、LinuxマシンはLinux_Templateに、それぞれリンクします。

ルールは10分(600秒)間隔で実行されます。

ルールを追加すると、自動的にディスカバリが開始され、チェック結果に応じて指定されたアクションを実行するためのディスカバリイベントが生成されます。

ステップ 2 新たに検出された Linux サーバを追加するアクションの定義



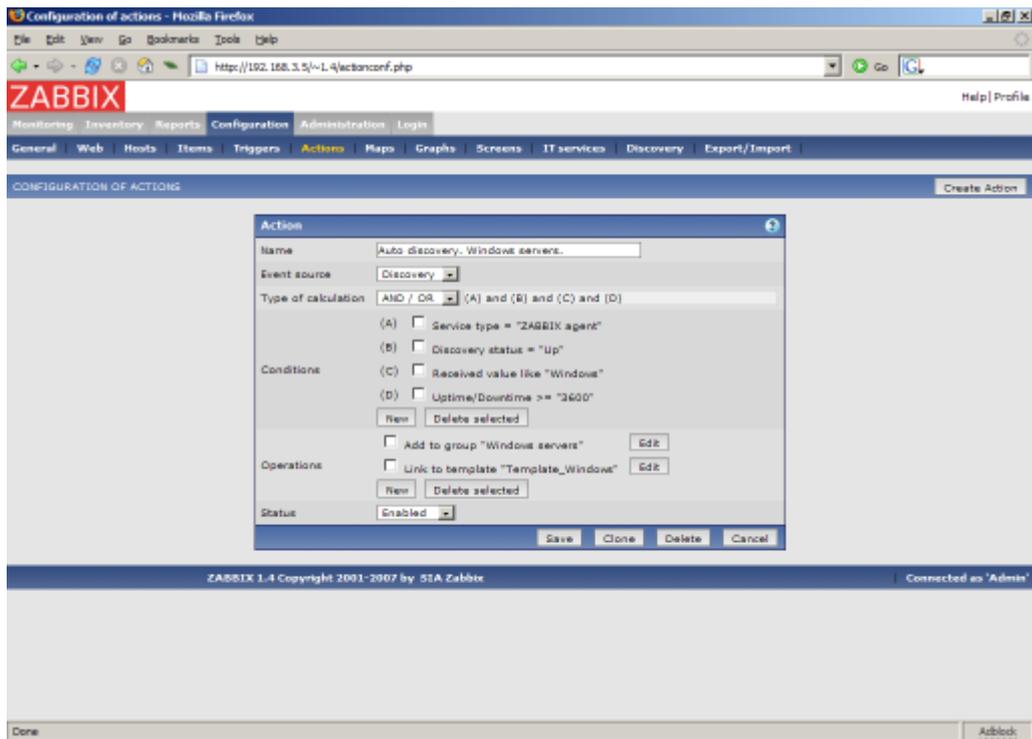
このアクションは以下のコンディションが成立する場合に有効になります。

- サービス「ZABBIX エージェント」が動作している
- system.uname(ルールを定義する際に使用した ZABBIX エージェントのキー)の値に「Linux」という文字列が含まれる
- アップタイムが 1 時間(3600 秒)を超えている

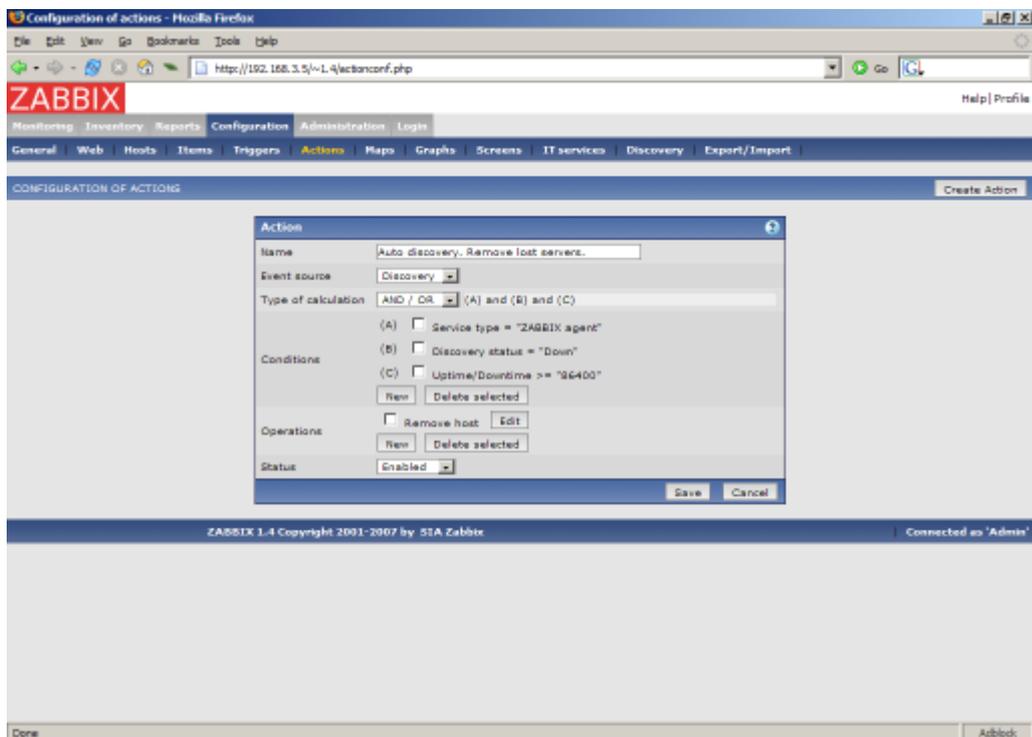
アクションによって以下のオペレーションが実行されます。

- 新たに検出したホストをグループ「Linux servers」に追加する(まだ追加されていないホストの場合はホストの追加も実行する)
- ホストをテンプレート「Template_Linux」にリンクする(「Template_Linux」に含まれるアイテムとトリガーを使用してホストの監視が自動的に開始される)

ステップ 3 新たに検出された Windows サーバを追加するアクションの定義



ステップ 4 動作していないサーバを削除するアクションの定義



サービス「ZABBIX エージェント」が 24 時間(86400 秒)以上停止している場合、該当するサーバは削除されます。

12.高度な SNMP 監視

12.1. 特別な MIB

最もよく使用される一部の SNMP MIB は ZABBIX によって自動的に数値表現に変換されます。たとえば、**ifIndex** は **1.3.6.1.2.1.2.2.1.1** に変換され、**ifIndex.0** は **1.3.6.1.2.1.2.2.1.1.0** に変換されます。

以下の表は特別な MIB のリストです。

特別なMIB	識別子	説明
ifIndex	1.3.6.1.2.1.2.2.1.1	各インタフェースの一意的な値。
ifDescr	1.3.6.1.2.1.2.2.1.2	インタフェースに関する情報を含むテキスト文字列。この文字列には、製造元の名前、製品名、およびハードウェアインタフェースのバージョンが含まれている必要があります。
ifType	1.3.6.1.2.1.2.2.1.3	インタフェースのタイプ。プロトコルスタックのネットワーク層の直下にある物理/リンクプロトコルによって区別されます。
ifMtu	1.3.6.1.2.1.2.2.1.4	インタフェースで送受信できる最大データグラムサイズ(オクテットで指定)。
ifSpeed	1.3.6.1.2.1.2.2.1.5	インタフェースの現在の推定帯域幅(ビット/秒)。
ifPhysAddress	1.3.6.1.2.1.2.2.1.6	プロトコルスタックのネットワーク層の直下にあるプロトコル層におけるインタフェースのアドレス。
ifAdminStatus	1.3.6.1.2.1.2.2.1.7	インタフェースの現在の管理状態。
ifOperStatus	1.3.6.1.2.1.2.2.1.8	インタフェースの現在の動作状態。
ifInOctets	1.3.6.1.2.1.2.2.1.10	インタフェースで受信された合計オクテット数(フレーミング文字を含む)。
ifInUcastPkts	1.3.6.1.2.1.2.2.1.11	上位層プロトコルに配信されたサブネットワークユニキャストパケット数。
ifInNUcastPkts	1.3.6.1.2.1.2.2.1.12	上位層プロトコルに配信された非ユニキャスト(つまり、サブネットワークブロードキャストまたはサブネットワークマルチキャスト)パケット数。
ifInDiscards	1.3.6.1.2.1.2.2.1.13	上位層プロトコルへの配信を防止する際にエラーが検出されていないにもかかわらず、破棄するように指定された着信パケット数。このようなパケットを破棄する1つの理由として、バッファ領域を解放することが考えられます。
ifInErrors	1.3.6.1.2.1.2.2.1.14	上位層プロトコルへの配信を防止する際にエラーが検出された着信パケット数。
ifInUnknownProtos	1.3.6.1.2.1.2.2.1.15	インタフェースを介して受信されたパケットのうち、不明なプロトコルまたはサポートされていないプロトコルが原因で破棄された

		パケット数。
ifOutOctets	1.3.6.1.2.1.2.2.1.17	インタフェースから送信された合計オクテット数(フレーミング文字を含む)。
ifOutNUcastPkts	1.3.6.1.2.1.2.2.1.18	上位層プロトコルによってサブネットワークユニキャストアドレスに配信するように要求された合計パケット数(破棄されたパケットまたは送信されなかったパケットを含む)。
ifOutDiscards	1.3.6.1.2.1.2.2.1.19	配信を防止する際にエラーが検出されていないにもかかわらず、破棄するように指定された発信パケット数。このようなパケットを破棄する1つの理由として、バッファ領域を解放することが考えられます。
ifOutErrors	1.3.6.1.2.1.2.2.1.20	エラーが原因で配信されなかった発信パケット数。
ifOutQLen	1.3.6.1.2.1.2.2.1.21	出力パケットキューの長さ(パケット単位)。

12.2. 動的インデックスの使用

アイテム OID 用の特別な構文を使用して動的データ(ネットワークインタフェースのランダム ID など)を処理できます。

構文は以下のとおりです。

```
<base OID of data>["index", "<base OID of index>", "<string to search for>"]
```

たとえば、Cisco デバイスの GigabitEthernet0/1 インタフェースの ifInOctets 値を取得するには、以下の OID を使用できます。

```
ifInOctets["index", "ifDescr", "GigabitEthernet0/1"]
```

パラメータ	説明
[データのベースOID(base OID of data)]	データの取得に使用するベースOID。
[index(インデックス)]	処理方法。現在サポートされている方法は以下の1つです。 [index(インデックス)] - インデックスを検索し、それをベースOIDに追加します。
[データのベースOID(base OID of index)]	文字列の検索に使用するOID。
[検索する文字列(string to search for)]	値との完全一致検索で使用する文字列。大文字と小文字が区別されます。

別の例として、Apache プロセスのメモリ使用量を取得する方法を以下に示します。

```
HOST-RESOURCES-MIB::hrSWRunPerfMem["index", "HOST-RESOURCESMIB::hrSWRunPath", "/usr/sbin/apache2"]
```

...

```
HOST-RESOURCES-MIB::hrSWRunPath.5376 = STRING: "/sbin/getty"  
HOST-RESOURCES-MIB::hrSWRunPath.5377 = STRING: "/sbin/getty"
```

```
HOST-RESOURCES-MIB::hrSWRunPath.5388 = STRING: "/usr/sbin/apache2"  
HOST-RESOURCES-MIB::hrSWRunPath.5389 = STRING: "/sbin/sshd"
```

...

該当するインデックスは 5388 です。このインデックスは目的の値を取得するためにデータ OID に追加されます。

```
HOST-RESOURCES-MIB::hrSWRunPerfMem.5376 = INTEGER: 528 KBytes  
HOST-RESOURCES-MIB::hrSWRunPerfMem.5377 = INTEGER: 528 KBytes  
HOST-RESOURCES-MIB::hrSWRunPerfMem.5388 = INTEGER: 31468 KBytes  
HOST-RESOURCES-MIB::hrSWRunPerfMem.5389 = INTEGER: 31740 KBytes  
HOST-RESOURCES-MIB::hrSWRunPerfMem.5390 = INTEGER: 32116 KBytes  
HOST-RESOURCES-MIB::hrSWRunPerfMem.5391 = INTEGER: 30420 KBytes  
HOST-RESOURCES-MIB::hrSWRunPerfMem.5392 = INTEGER: 32560 Kbytes
```

注:

動的インデックスはSNMPクエリの数を増加させるため慎重に使用してください。ZABBIXではキャッシュは実行されないため、アイテム値を取得するときは常に検索が実行されます。

13. IPMI デバイスの監視

13.1. 目標

ZABBIX IPMI 監視には、様々な目標があります。

- IPMI デバイスの状態と可用性の監視
- リモート IPMI ベースの管理機能

リモートでのリスタート、シャットダウン、停止などのコマンドを ZABBIX フロントエンドから自動または手動で実行できます。

13.2. IPMI パラメータ

ZABBIX IPMI 監視は、IPMI をサポートしているデバイス(HP iLO、Sun ハードウェアなど)に対してのみ機能します。

IPMI 監視を使用するためには、IPMI コマンドを処理するようにホストを設定する必要があります。また、IPMI エージェントの IP アドレス、ポート番号、ユーザ名、およびパスワードを適切に設定する必要があります。

詳細については、ホストの設定を参照してください。

13.3. IPMI アクション

以下の 2 種類のアクションを定義できます。

- 自動アクション(自動的に実行)
- IPMI スクリプト(ZABBIX GUI から手動で実行可能)

詳細については、本書の対応するセクションを参照してください。

14. プロキシの使用

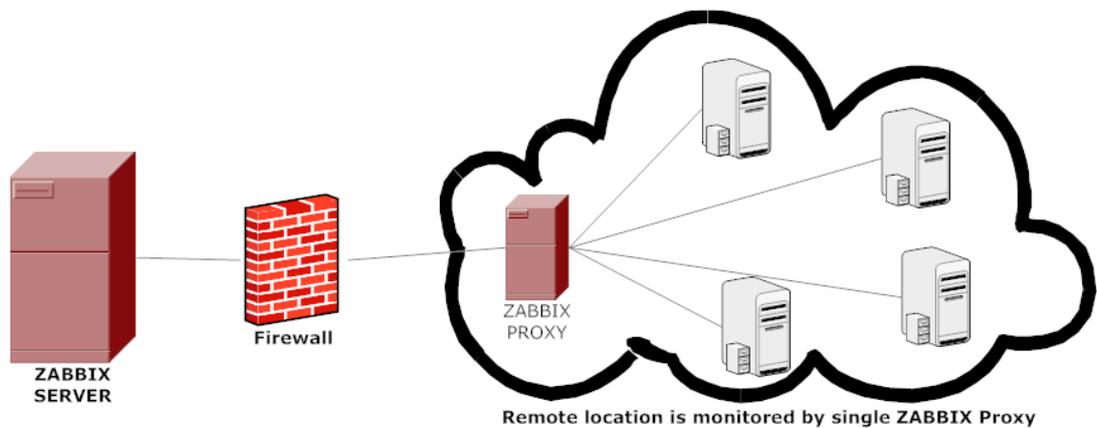
ZABBIX プロキシを使用すると、ZABBIX 環境のメンテナンスが大幅に単純化され、中央の ZABBIX サーバのパフォーマンスが向上します。

また、全てのエージェントとプロキシから 1 台の ZABBIX サーバに情報が送信され、全てのデータが中央で収集される場合、ZABBIX プロキシを使用することにより集中監視および分散監視を最も簡単に実装できます。

14.1. プロキシの特長

ZABBIX プロキシは、以下のような多くの目的で使用できます。

- 数千ものデバイスを監視するときに ZABBIX サーバの負荷を分散
- リモートの場所の監視



- 信頼性の低い通信が行われている場所の監視
- 分散監視のメンテナンスの単純化

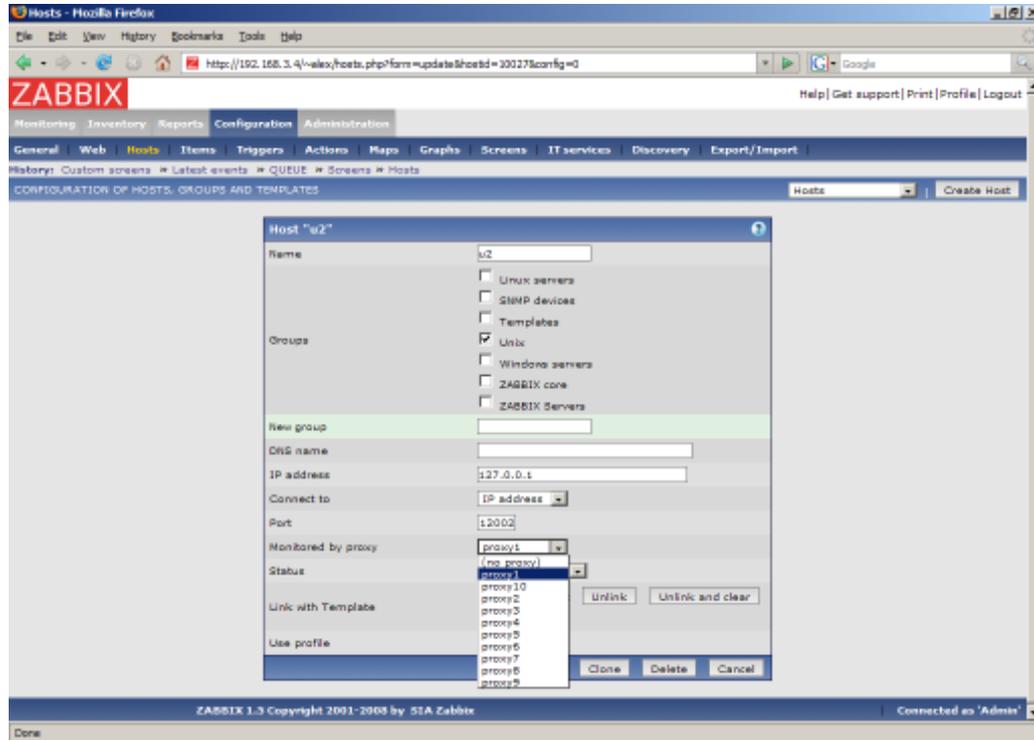
14.2. プロキシとノードの違い

プロキシとノードのどちらを使用するかを決定する際には、様々な点を考慮する必要があります。

	軽量	GUI	独立して動作	メンテナンスが簡単	DBの自動作成	ローカル管理	組み込みハードウェアに対応	一方向TCP接続	設定の集中化	通知の生成
ノード	×	○	○	×	×	○	×	○	×	○
プロキシ	○	×	○	○	○	×	○	○	○	×

14.3. 設定

ZABBIX サーバまたは ZABBIX プロキシで全てのホストを監視できます。この設定はホスト定義スクリーンで行います。



ホストをプロキシで監視するように設定すると、プロキシによりホストのパフォーマンスおよび可用性データが収集されます。データはプロキシで収集され、その後の処理のために ZABBIX サーバに送信されます。

15.分散監視

ZABBIX は、階層型分散監視をサポートするように設定できます。

15.1. 目標

分散監視には、様々な目標があります。

- 単一または複数の拠点からの監視処理全体のコントロール:

ZABBIX 管理者は、単一拠点の ZABBIX ウェブフロントエンドから全ノードの設定をコントロールできます。

- 階層型監視:

複数階層から構成される複雑な環境の監視に使用します。

- 複雑で大規模な環境の監視:

特に地理的に分散する複数の拠点を監視する場合に役に立ちます。

- 負荷の高い ZABBIX サーバのオーバーヘッドの分散:

1 台の ZABBIX サーバで数千台のホストを監視する必要がある場合でも対応できます。

15.2. 概要

ZABBIX を使用すると、効果的かつ信頼できる方法で分散 IT インフラストラクチャを監視できます。分散設定全体の設定は、単一拠点で共通のウェブインターフェースから実行できます。

ZABBIX は、分散設定で最大 **1000** 個のノードをサポートします。各ノードは、その場所の監視を行います。ノードはローカルに設定したり、マスターノードから設定することができます。マスターノードには、全ての子ノードの設定データのコピーが置かれています。子ノードの設定は、オフラインモード、すなわちマスターノードと子ノードが接続されていない状態でも実行できます。

階層型分散監視を使用すると、ノードをツリー構造で構成できます。各ノードは、そのマスターノードにのみレポートします。

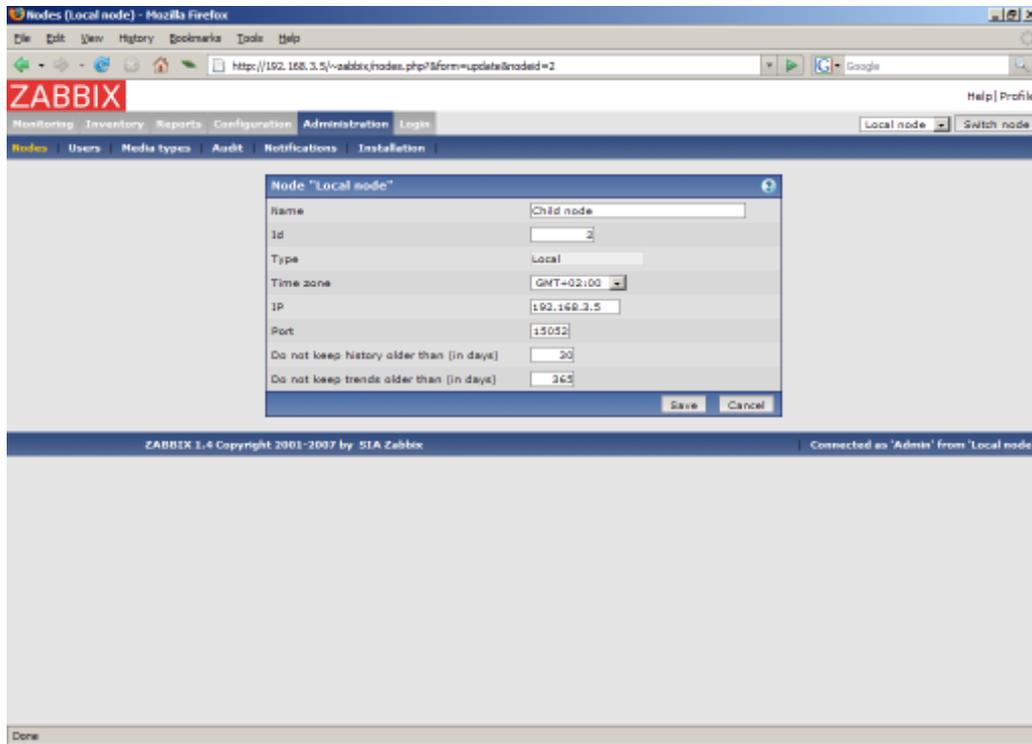
全てのノードは、通信障害が発生した後も動作可能であり、履歴情報とイベントはローカルに保存されます。通信が復旧したときに子ノードがマスターノードにデータを送信するように設定できます。

ZABBIX 分散設定に新しいノードを接続したり、分散設定からノードを切断しても、分散設定の機能は一切損なわれることはなく、要求されるノードをリスタートする必要もありません。

各ノードは独自に設定することができ、通常の ZABBIX サーバとして動作します。

15.3. 設定

15.3.1. ノードの設定



ノードのパラメータを以下に示します。

パラメータ	説明
[名前]	一意なノード名。
[ID]	一意なノードID。
[タイプ]	[ローカル] - ローカルノード [リモート] - リモートノード
[タイムゾーン]	ノードのタイムゾーン。ノード間で時間関連のデータを転送する際に、タイムスタンプは自動的にローカルのタイムゾーンに変換されます。
[IPアドレス]	ノードのIPアドレス。ZABBIXトラッパーはこのIPアドレスで待ち受ける必要があります。
[ポート]	ノードのポート番号。ZABBIXトラッパーはこのポート番号で待ち受ける必要があります。デフォルトは10051です。
[履歴の保存期間(日)]	ローカル以外の履歴データにのみ適用されます。ノードの履歴を保存する期間(日単位)を指定します。
[トレンドの保存期間(日)]	ローカル以外のトレンドデータにのみ適用されます。ノードのトレンドを保存する期間(日単位)を指定します。

15.3.2. 単純な設定

ここでは、中央ノード(ノード名: Central Node)と子ノード(ノード名: Child Node)から構成される単純な設定を例として説明します。

中央ノードは子ノードの設定を完全にコントロールします。子ノードは中央ノードにイベント、履歴、およびトレンドをレポートします。

中央ノードの NodeID は 1、子ノードの NodeID は 2 です。

中央ノードの IP アドレス: 192.168.3.2、ポート番号: 10051

子ノードの IP アドレス: 192.168.3.5、ポート番号: 15052

中央ノード:

ステップ 1 ZABBIX のインストール

標準のインストール手順に従ってデータベースを作成し、ZABBIX のフロントエンドとバイナリをインストールします。

ステップ 2 サーバ設定ファイルに NodeID を設定

ファイル `zabbix_server.conf` に以下の行を設定します。

```
NodeID=1
```

ステップ 3 データベースデータの変換

ZABBIX サーバを使用して、最初のノードが使用できるように一意な ID を変換します。

```
cd bin
```

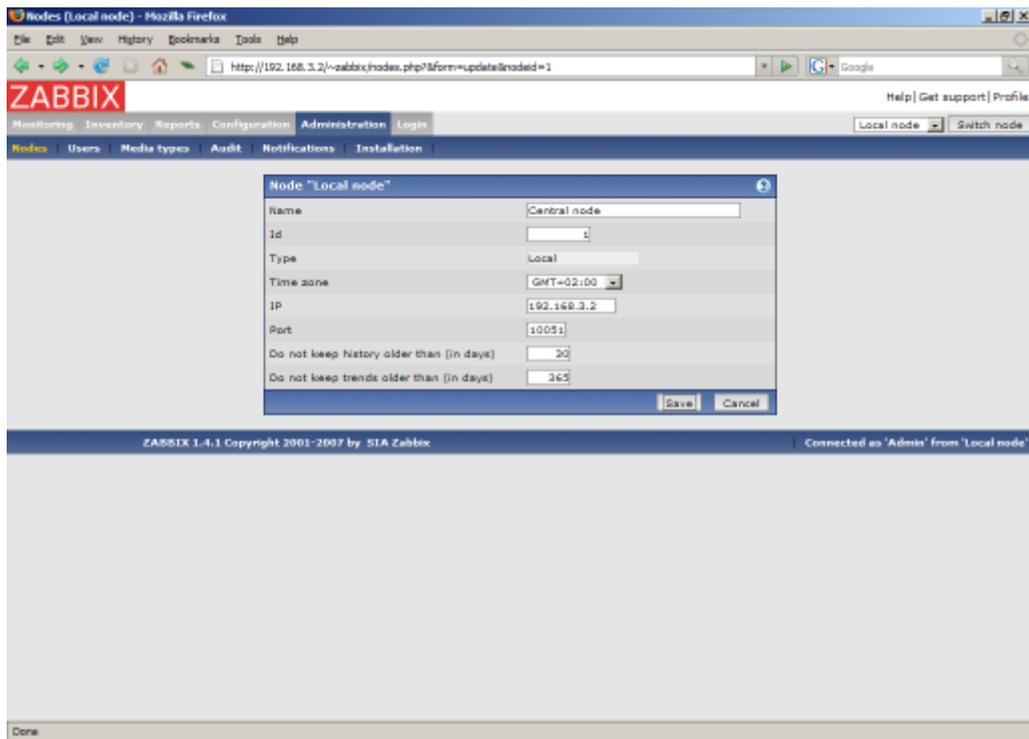
```
./zabbix_server -n 1 -c /etc/zabbix/zabbix_server.conf
```

```
Converting tables ..... done.
```

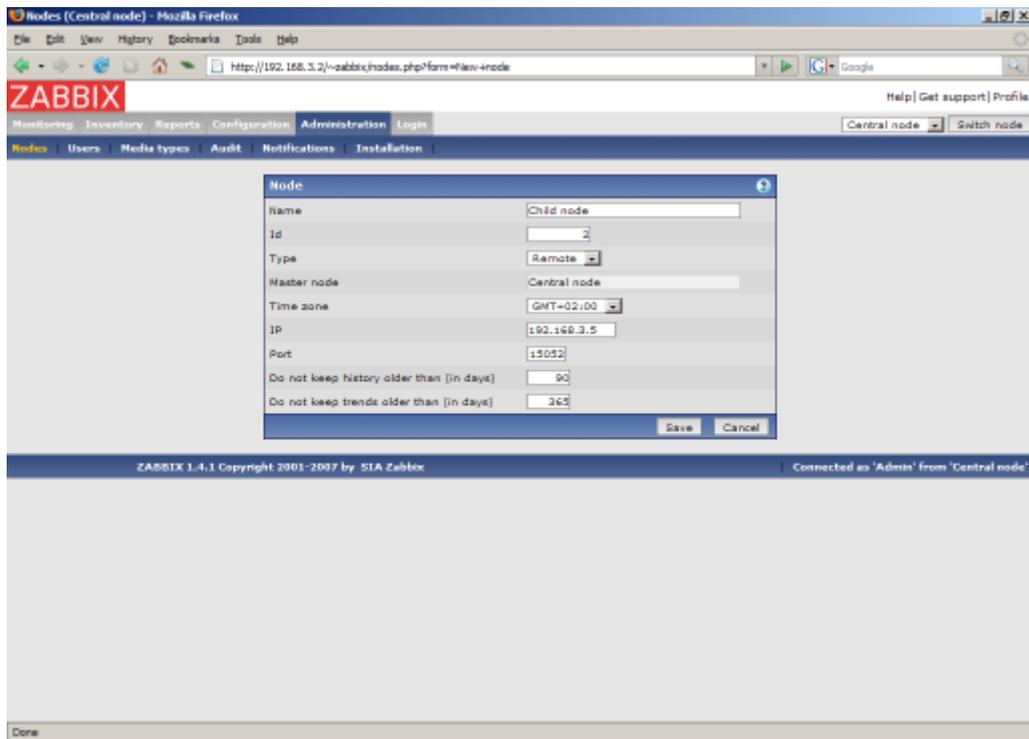
```
Conversion completed.
```

注: このコマンドは1回だけ実行する必要があります。ここで指定したオプションは、ZABBIXサーバを起動する際に指定する必要はありません。

ステップ 4 ノードパラメータの設定



ステップ 5 子ノードの追加



ステップ 6 マスターノードの起動

サーバログファイルのスタートアップメッセージに、設定した NodeID が表示されることを確認します。
 31754:20070629:150342 server #16 started [Node watcher. Node ID:1]

子ノード:

ステップ 1 ZABBIX のインストール

標準のインストール手順に従ってデータベースを作成し、ZABBIX のフロントエンドとバイナリをインストールします。

ステップ 2 サーバ設定ファイルに NodeID を設定

ファイル `zabbix_server.conf` に以下の行を設定します。

```
NodeID=2
```

ステップ 3 データベースデータの変換

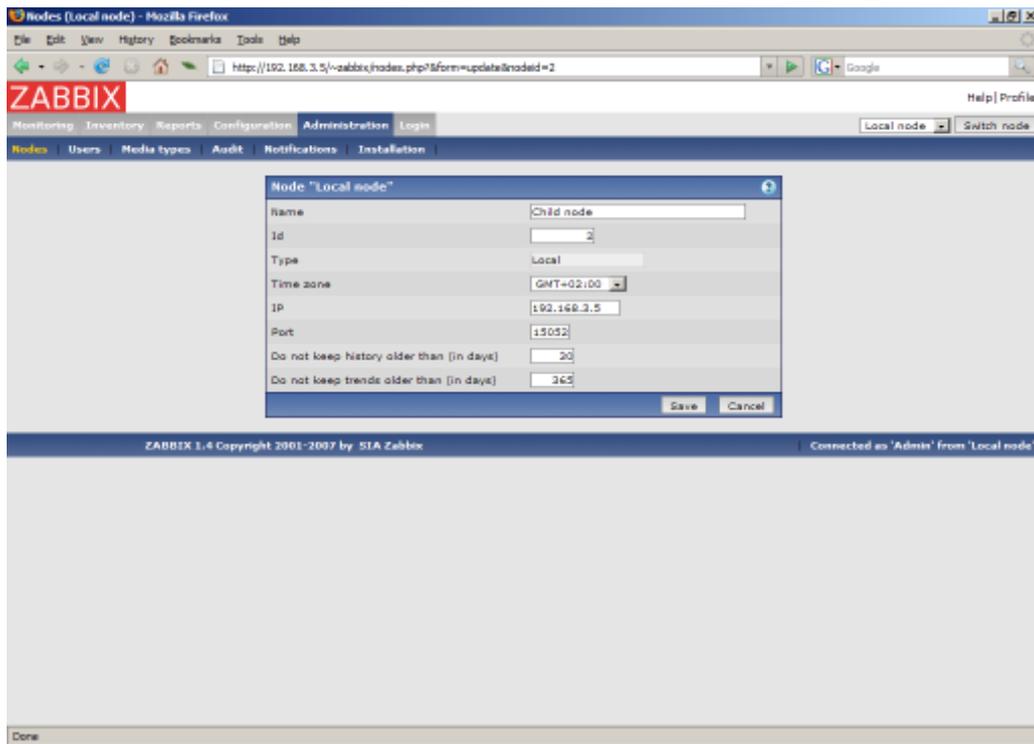
ZABBIX サーバを使用して、最初のノードが使用できるように一意な ID を変換します。

```
cd bin
./zabbix_server -n 2 -c /etc/zabbix/zabbix_server.conf
Converting tables ..... done.
```

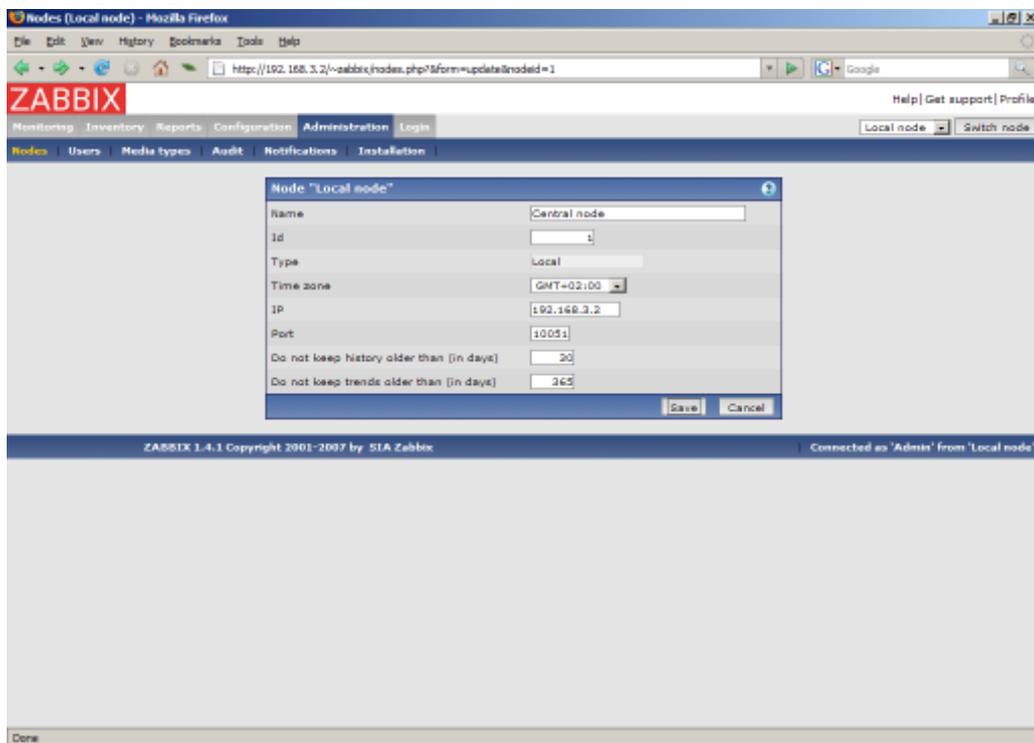
```
Conversion completed.
```

注: このコマンドは1回だけ実行する必要があります。ここで指定したオプションは、ZABBIXサーバを起動する際に指定する必要はありません。

ステップ 4 ノードパラメータの設定



ステップ 5 マスターノードの追加



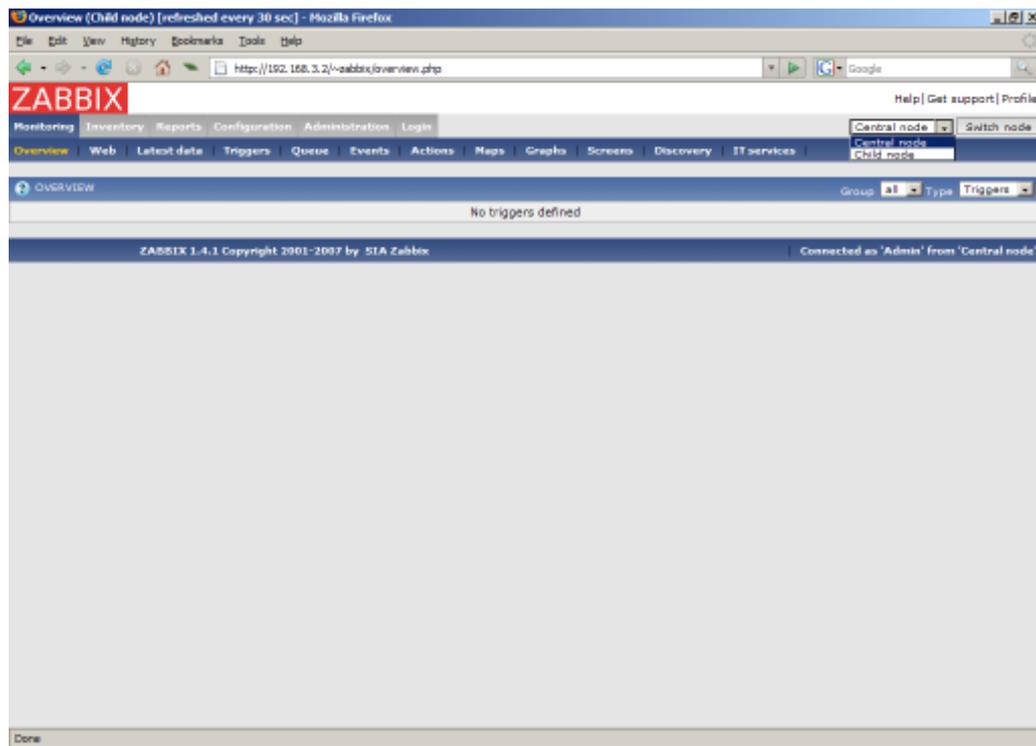
ステップ 6 子ノードの起動

サーバログファイルのスタートアップメッセージに、設定した NodeID が表示されることを確認します。

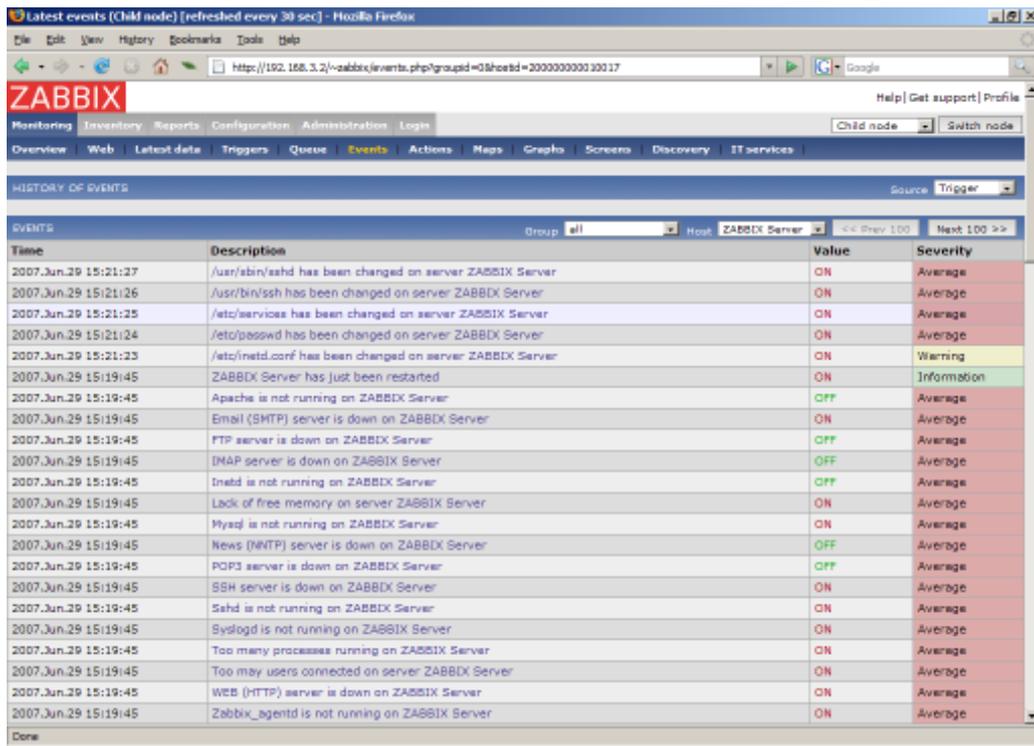
```
27524:20070629:150622 server #9 started [Node watcher. Node ID:2]
```

動作確認

ノードを定義すると、有効なノードの選択肢として自動的に表示されます。

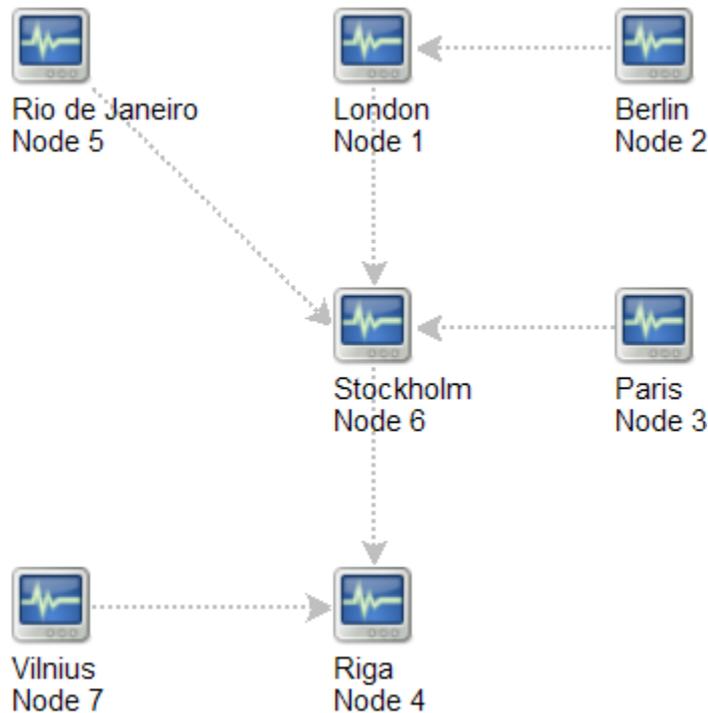


子ノードを監視するためのホストを追加して、マスターノードにイベントが送信されることを確認します。



15.3.3. 複雑な設定

ここでは、7 個のノードで構成される設定例について説明します。各ノードはローカル(ローカルのウェブインターフェースを使用)で、またはそのマスターノードから設定できます。



この例では、Riga という名前のノード(ノード 4)が全ての子ノードからイベントを収集します。さらに、各ノードのヒストリ情報も収集するように設定することもできます。

15.4. プラットフォームの独立性

ノードは、他のノードと無関係に、独自のプラットフォーム(OS、ハードウェア)およびデータベースエンジンを使用できます。また、子ノードは ZABBIX フロントエンドを使用しなくてもインストールできます。

実用的には、SQLite または MySQL MyISAM が動作する ZABBIX サーバには多少能力が劣るハードウェアを使用して、上位レベルのノードには高性能なハードウェアと MySQL InnoDB、Oracle、または PostgreSQL のいずれかのバックエンドを組み合わせて使用する方法が考えられます。

15.5. 単一ノードの設定

分散環境では各ノードに一意的なノード ID を割り当てる必要があります。

そのための詳細なステップを以下に示します。

ステップ 1 標準のインストール手順の実行

標準のインストール手順を実行します。ただし、ZABBIX サーバは起動しないでください。続いて ZABBIX フロントエンドのインストールと設定を行い、ZABBIX データベースを作成して data.sql からデータを追加します。

ステップ 2 zabbix_server.conf の設定

NodeID を ZABBIX サーバ設定ファイルに追加します。NodeID は一意的なノード ID である必要があります。

ステップ 3 マスターノードと子ノードの設定

ZABBIX フロントエンドを使用して、**ステップ 2** で設定したノードと直接通信するノードの詳細を設定します。IP アドレスとポート番号が全て正確に設定されていることを確認します。

ステップ 4 ZABBIX ノードの起動

ZABBIX サーバを起動します。

```
shell> ./zabbix_server
```

全て正しく設定されている場合、ZABBIX ノードは自動的に設定および分散設定内の全ノードとのデータ交換を開始します。サーバログファイルには、以下のようなメッセージが表示されます。

```
...
11656:20061129:171614 NODE 2: Sending data of node 2 to node 1 datalen
3522738
11656:20061129:171614 NODE 2: Sending data of node 2 to node 1 datalen
20624
...
```

15.6. ノード間のスイッチ

分散設定内のノードに接続する場合、GUI の右上隅にある接続可能な子ノードのリストを使用します。表示されているのは、現在のノードです。

GUI に表示される情報は全て、このリストで選択したノードの情報です。

15.7. データフロー

15.7.1. 子からマスターに送信されるデータ

子ノードはそれぞれ、設定変更、履歴データ、およびイベントをそのマスターノードに定期的送信します。

データ	頻度
設定変更	120秒ごと
イベント	10秒ごと
履歴	10秒ごと

通信障害が発生した場合、子ノードはデータを再送信します。

トレンドは受信した履歴データに基づいてローカルに計算されます。

ノード間で運用データが送信されることはありません。たとえば、アイテム関連の情報(最終チェック、最新値など)はローカルにのみ存在します。

注: イベントと履歴の送信をコントロールするには、設定パラメータNodeNoEventsとNodeNoHistoryを使用します。

15.7.2. マスターから子に送信されるデータ

マスターノード(少なくとも 1 つの子ノードを持つノード)はそれぞれ、設定変更を子ノードに(直接またはマスターノードに直接接続されている他の子ノードを経由して)定期的送信します。

データ	頻度
設定変更	120秒ごと

マスターノードの設定が子ノードに送信されることはありません。

15.7.3. ファイアウォール設定

ノード間通信で使用するのは TCP プロトコルだけです。

データフロー	送信元ポート	送信先ポート
子からマスター	任意	10051
マスターから子	任意	10051

これは ZABBIXトラッパが使用するデフォルトポートです。

15.8. パフォーマンスに関する検討事項

どのノードも分散設定では多めの処理リソースを必要とします。マスターノードには、十分な処理能力およびローカルデータと子ノードから受信した全データを保存できるだけの容量が必要です。また、ネットワーク通信には、新しいデータを遅延なく転送できるだけの速度が要求されます。

16.ZABBIX GUI のメンテナンスモード

ZABBIX GUI を一時的に無効化して、フロントエンドへのアクセスを禁止することができます。

このモードは、ZABBIX データベースをユーザによる変更から保護し、データベースの整合性を保つために役立ちます。

ZABBIX GUI をメンテナンスモードにしている間は、ZABBIX データベースを停止できます。

16.1. 目標

メンテナンスモードには、様々な目標があります。

- ZABBIX データベースをユーザによる変更から保護
- データベースメンテナンスの実行
- メンテナンス作業の理由をユーザに通知
- メンテナンスモードの正常実行中、IP アドレスの範囲内のユーザに対して GUI の操作を許可
- メンテナンスの終了時に通常モードに自動復帰

16.2. 設定

メンテナンスモードを有効にするには、`conf/maintenance.conf.php` ファイルで以下の行のコメント化を解除する必要があります。

```
// Maintenance mode
define('ZBX_DENY_GUI_ACCESS',1);

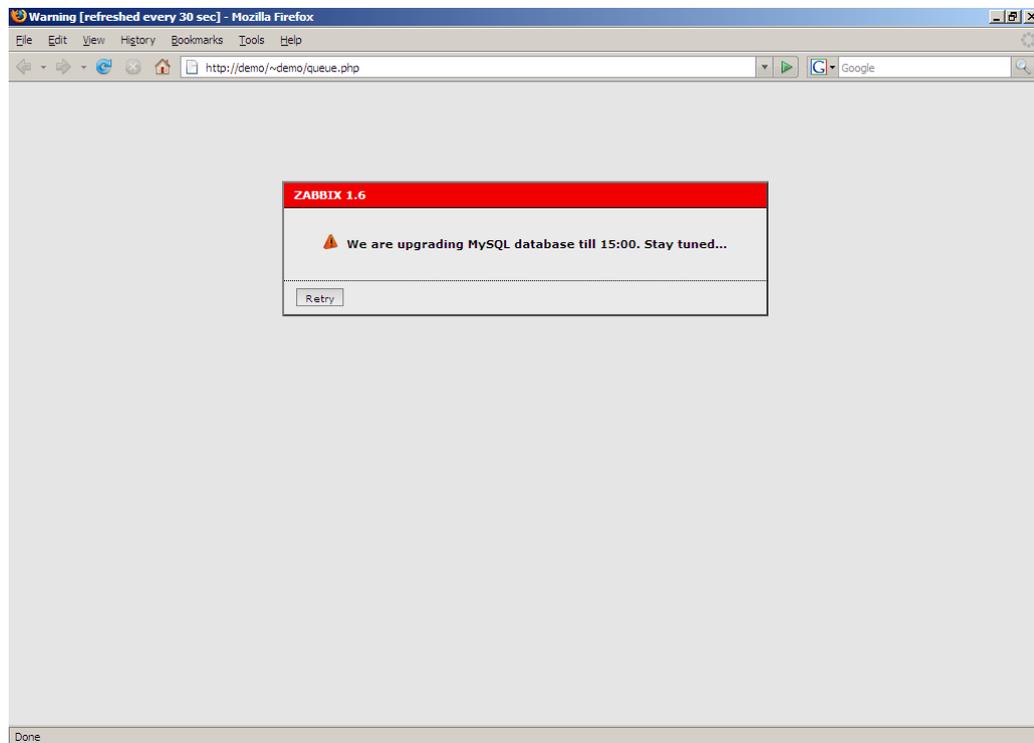
// IP range, who allowed to connect to FrontEnd
$ZBX_GUI_ACCESS_IP_RANGE = array('127.0.0.1');

// MSG showed on Warning screen!
$_REQUEST['warning_msg'] = 'ZABBIX is under maintenance.';
```

パラメータ	説明
[ZBX_DENY_GUI_ACCESS]	メンテナンスモードを有効にします。 1 - メンテナンスモードが有効になります。それ以外は無効です。
[ZBX_GUI_ACCESS_IP_RANGE]	メンテナンスモードでないときに、指定したIPアドレスからの接続を許可します。 以下に例を示します。 192.168.1.1-255
[warning_msg]	インフォメーションメッセージです。

16.3. メンテナンスモードのスクリーン

メンテナンスモードの間は、以下のスクリーンが表示されます。このスクリーンは 30 秒ごとに更新され、メンテナンスが終了したら自動的に通常の状態に戻ります。



17. ウェブインタフェース

ZABBIX ウェブインタフェースには、様々な便利な機能が備わっています。

- ほとんど全てのスクリーンでフルスクリーンモードをサポート
- Ctrl を押しながらマウスをクリックすることで複数のリスト項目(ホスト、アイテム、トリガーなど)を選択可能
- [トリガーのステータス]表示で音声アラームのオン/オフを切り替え可能
- 独自の好みや会社の色スキーマに合わせて新しいテーマを作成可能

17.1. 独自のテーマの作成

ZABBIX には、デフォルトで多数の事前定義済みテーマが用意されています。必要に応じて、以下のステップに従って独自のテーマを作成することもできます。作成したテーマを ZABBIX コミュニティと共有することも自由です。

ステップ 1 独自の CSS ファイルの作成

ZABBIX に付属の既存の CSS ファイルを基準にして作成できます。たとえば、styles/css_bb.css にある Black&Blue CSS ファイルを使用して新しい css_new.css を作成できます。

ステップ 2 適切な場所への新しい CSS ファイルの配置

作成した css_new.css ファイルを styles/ディレクトリに配置します。

ステップ 3 include/forms.inc.php の編集

このファイルを編集用に関き、css_bb.css を検索します。ここで、2 つのコードを追加する必要があります。

元のコード:

```
$cmbTheme = new CComboBox('theme',$theme);  
$cmbTheme->AddItem(ZBX_DEFAULT_CSS,S_SYSTEM_DEFAULT);  
$cmbTheme->AddItem('css_ob.css',S_ORIGINAL_BLUE);  
$cmbTheme->AddItem('css_bb.css',S_BLACK_AND_BLUE);
```

修正後のコード:

```
$cmbTheme = new CComboBox('theme',$theme);  
$cmbTheme->AddItem(ZBX_DEFAULT_CSS,S_SYSTEM_DEFAULT);  
$cmbTheme->AddItem('css_ob.css',S_ORIGINAL_BLUE);
```

```
$cmbTheme->AddItem('css_bb.css',S_BLACK_AND_BLUE);  
$cmbTheme->AddItem('css_new.css','MY_COOL_THEME');
```

ステップ 4 新しいテーマの有効化

ZABBIX GUI では、このテーマをデフォルトのテーマに設定するか、ユーザプロファイルでテーマを変更できます。

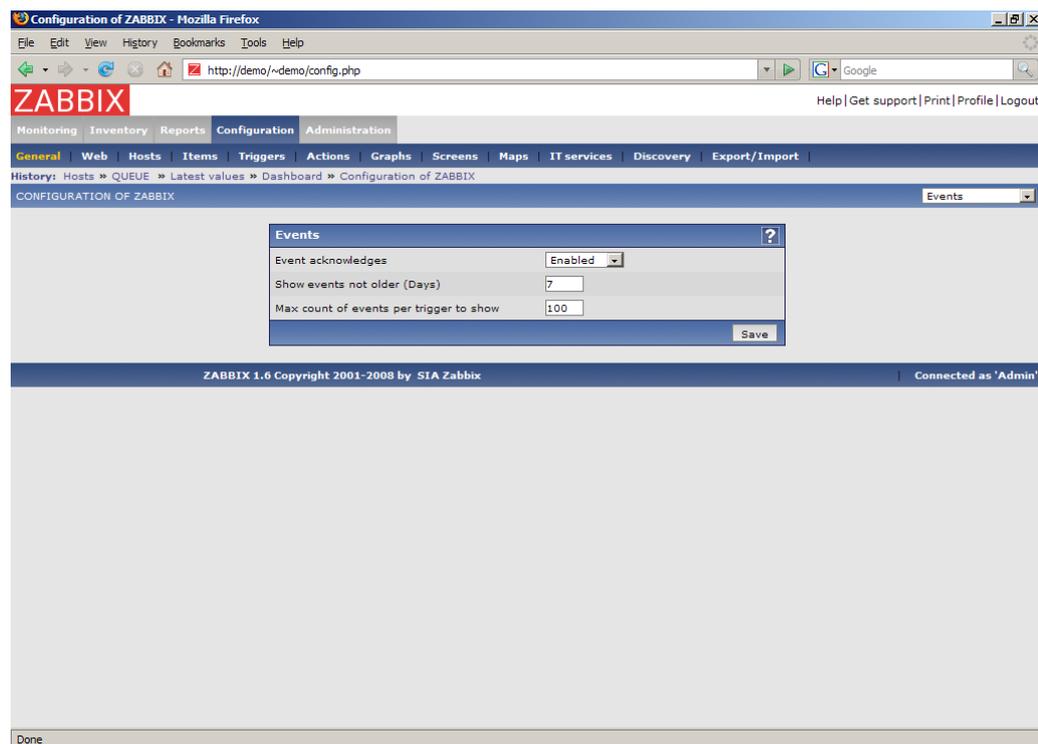
新しいテーマの外観と操作感をお楽しみください。

17.2. 設定

17.2.1. 一般設定

17.2.1.1. イベント

このスクリーンでは、イベント関連の設定を定義します。

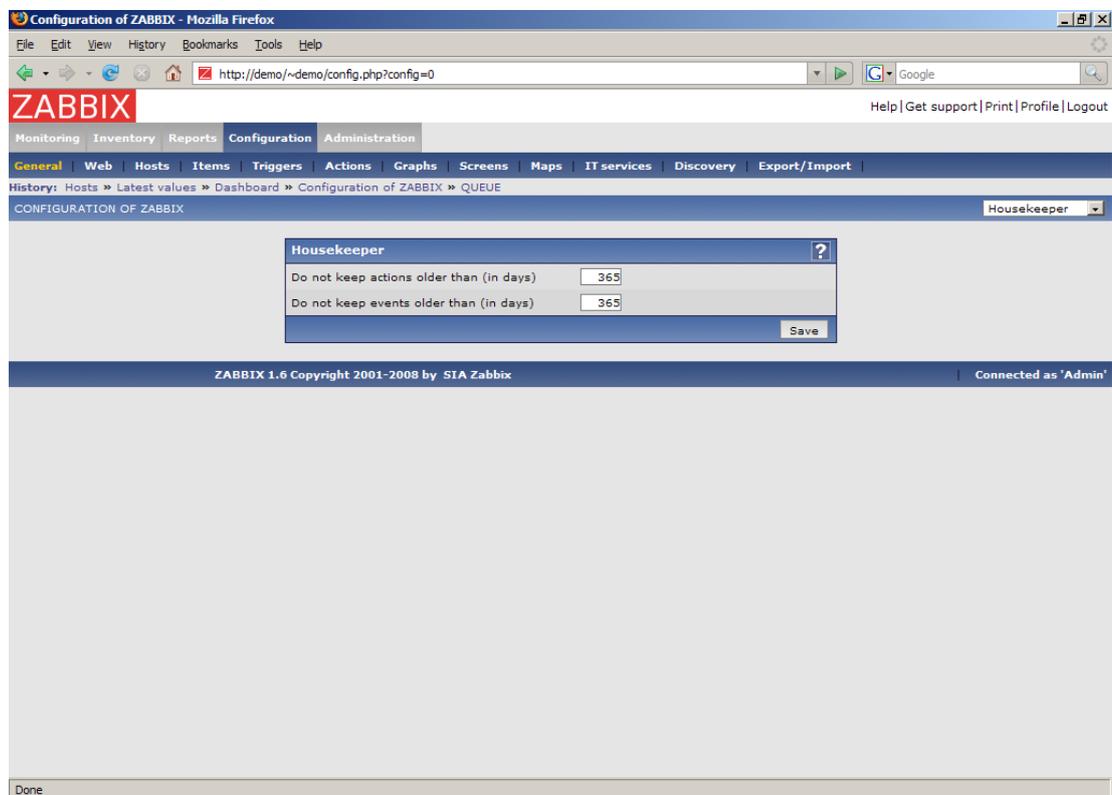


設定パラメータを以下に示します。

パラメータ	説明
[イベントの対応コメント]	ZABBIXインターフェースでイベントの認知を有効にするかどうかを定義します。
[イベントの表示期間](日数)	[トリガーのステータス]スクリーンにイベントを表示する日数を定義します。デフォルトは7日です。
[トリガーあたりのイベント表示の最大数]	[トリガーのステータス]スクリーンで各トリガーに対して表示するイベントの最大数を定義します。デフォルトは100です。

17.2.1.2. housekeeper

housekeeper は、ZABBIX サーバによって定期的に行われるプロセスです。古くなった情報やユーザが削除した情報をシステムから削除します。

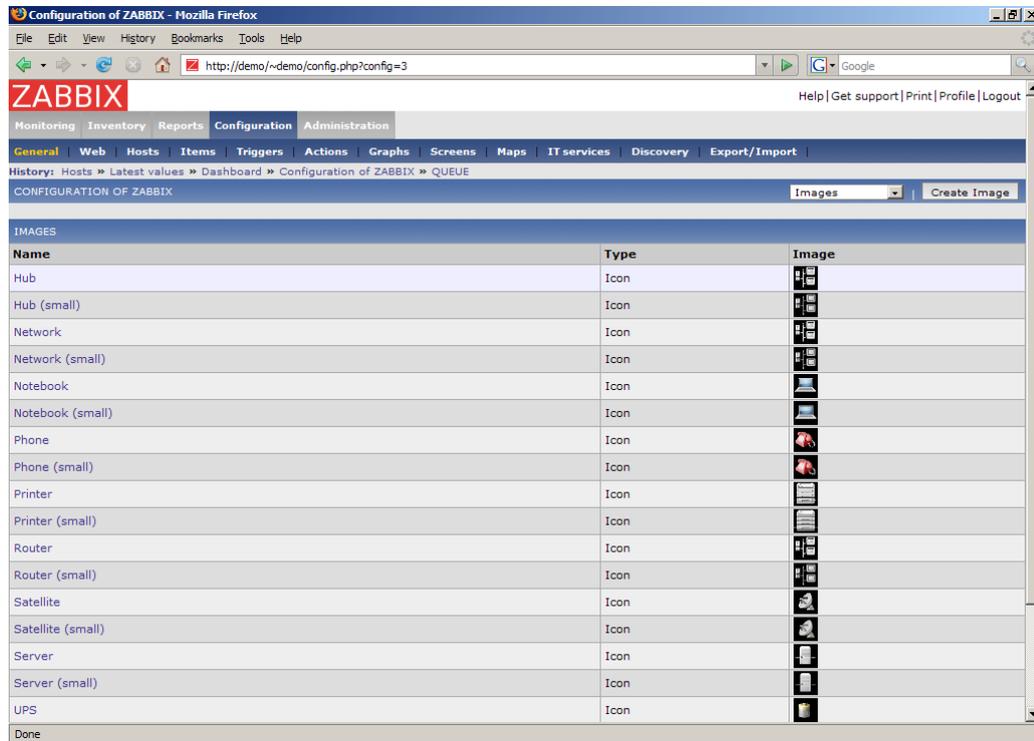


設定パラメータを以下に示します。

パラメータ	説明
[アクションの保存期間(日)]	実行されたアクション(メール、jabber、SMSなど)の履歴をデータベースに保存する日数を定義します。保存日数を超えたアクションは削除されます。
[イベントの保存期間(日)]	イベントの履歴をデータベースに保存する日数を定義します。保存日数を超えたイベントは削除されます。

17.2.1.3. イメージ

イメージのリスト



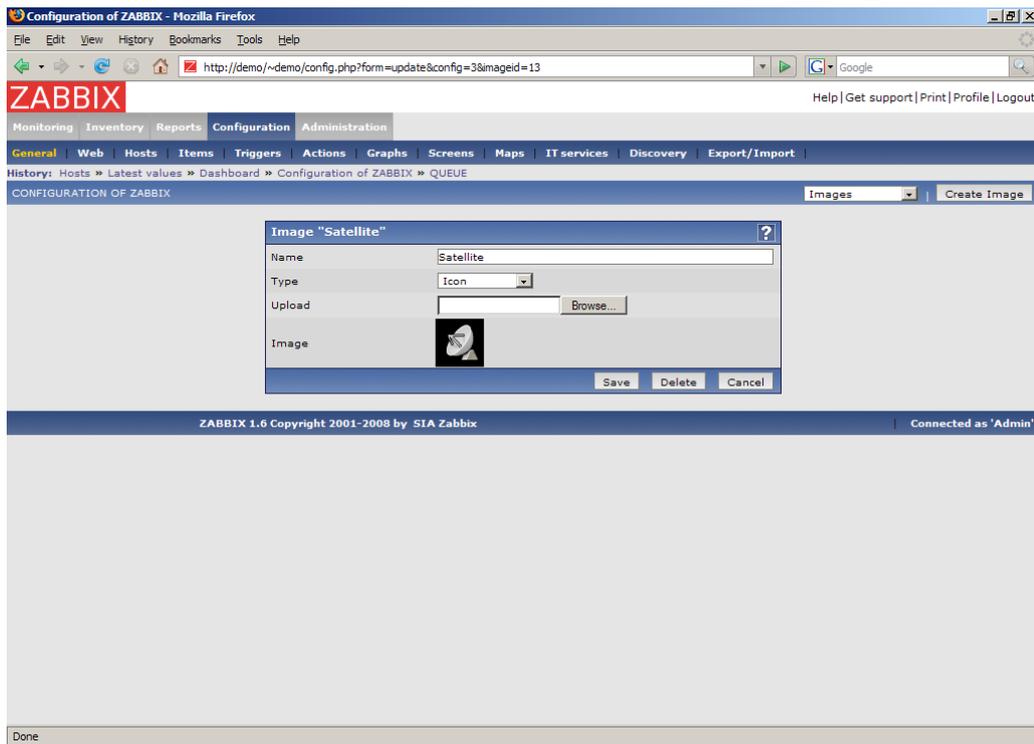
The screenshot shows the ZABBIX Configuration interface in a Mozilla Firefox browser. The page title is 'Configuration of ZABBIX - Mozilla Firefox'. The address bar shows 'http://demo/~demo/config.php?config=3'. The ZABBIX logo is visible in the top left. The navigation menu includes 'Monitoring', 'Inventory', 'Reports', 'Configuration', and 'Administration'. The 'Configuration' menu is expanded, showing sub-menus like 'General', 'Web', 'Hosts', 'Items', 'Triggers', 'Actions', 'Graphs', 'Screens', 'Maps', 'IT services', 'Discovery', and 'Export/Import'. The 'Hosts' menu is selected, and the 'Configuration of ZABBIX' page is displayed. The 'IMAGES' section is active, showing a table with columns 'Name', 'Type', and 'Image'. The table lists various device types and their corresponding icons.

Name	Type	Image
Hub	Icon	
Hub (small)	Icon	
Network	Icon	
Network (small)	Icon	
Notebook	Icon	
Notebook (small)	Icon	
Phone	Icon	
Phone (small)	Icon	
Printer	Icon	
Printer (small)	Icon	
Router	Icon	
Router (small)	Icon	
Satellite	Icon	
Satellite (small)	Icon	
Server	Icon	
Server (small)	Icon	
UPS	Icon	

イメージの定義

ZABBIX のイメージはデータベースに保存されます。イメージには以下の 2 種類があります。

- アイコン
- 背景



アイコンは、システムマップの項目を表示する際に使用します。

背景はシステムマップの背景イメージとして表示します。

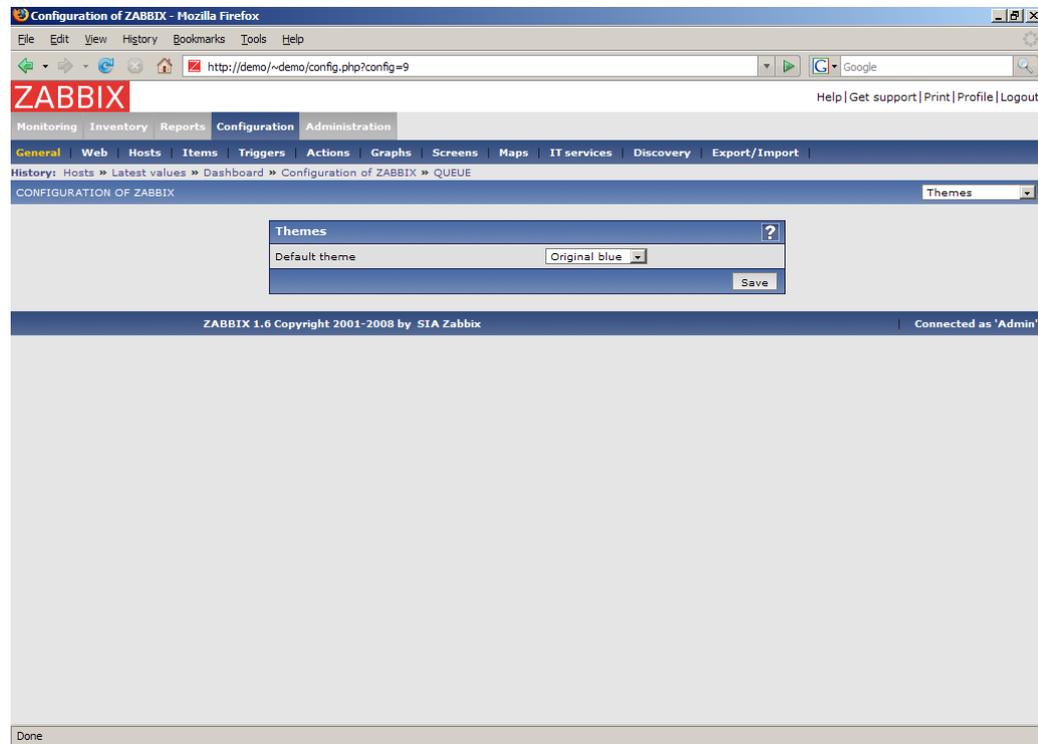
イメージの属性を以下に示します。

パラメータ	説明
[名前]	イメージの一意的な名前。
[タイプ]	[アイコン]または[背景]のどちらかを設定します。
[アップロード]	ZABBIXにアップロードするローカルファイル(PNG、JPEG)の名前。

注: アップロードできるイメージのサイズに制限はありませんが、サイズが1.5MBを超えるイメージはマップに表示できない可能性があります。その場合は、**php.ini**の**max_memory_size**の値を増やします。

17.2.1.4. テーマ

ZABBIXでは、ZABBIXフロントエンドの外観と操作感をカスタマイズするためのテーマがサポートされています。



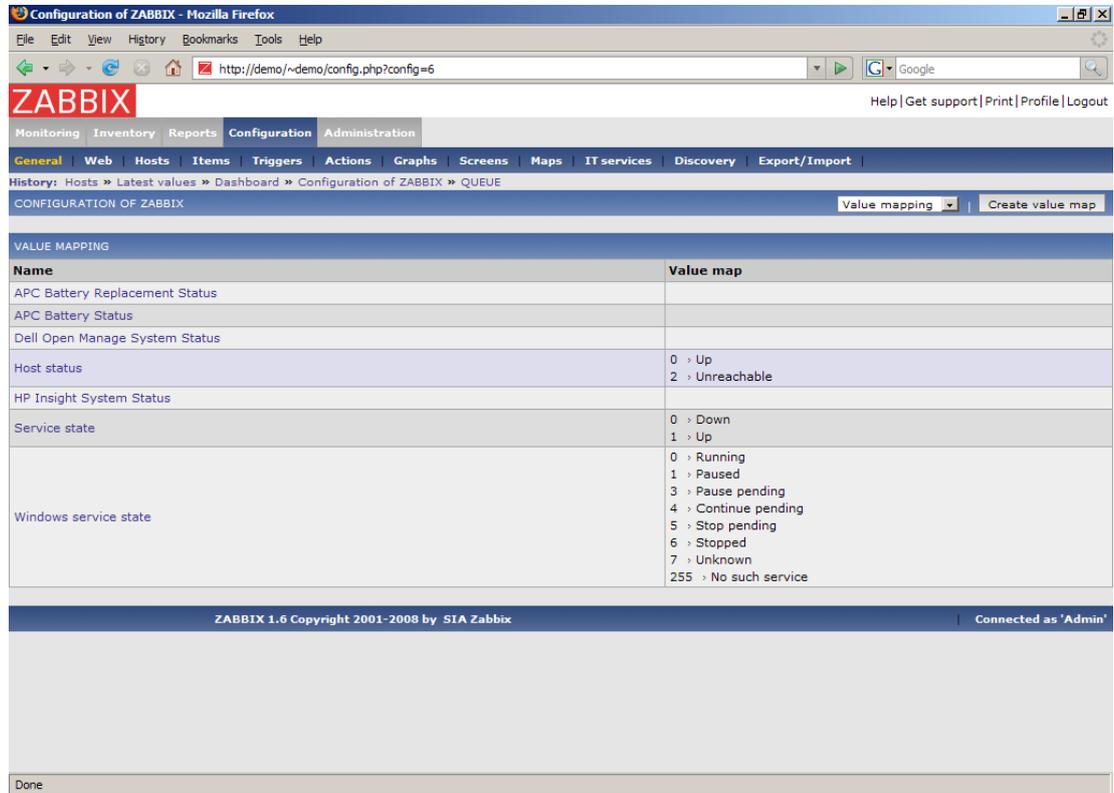
使用可能なパラメータを以下に示します。

パラメータ	説明
[デフォルトのテーマ]	全てのユーザに対して使用されるテーマ。デフォルトのテーマは[Original blue]です。ユーザは、ユーザプロフィールでデフォルトのテーマを上書きできます。

17.2.1.5. 値のマッピング

値のマッピングを使用すると、数値に文字列を割り当てることができます。

値のマッピングは、ZABBIX フロントエンドに表示されるデータおよびメール/jabber/SMS/その他の手段によって送信する情報に使用できます。



The screenshot shows the ZABBIX Configuration page in a Mozilla Firefox browser. The page title is "Configuration of ZABBIX - Mozilla Firefox". The browser address bar shows "http://demo/~demo/config.php?config=6". The ZABBIX logo is visible in the top left. The navigation menu includes Monitoring, Inventory, Reports, Configuration, and Administration. The Configuration menu is expanded, showing sub-menus: General, Web, Hosts, Items, Triggers, Actions, Graphs, Screens, Maps, IT services, Discovery, and Export/Import. The main content area is titled "CONFIGURATION OF ZABBIX" and has a "Value mapping" dropdown menu and a "Create value map" button. Below this is a table with the following data:

Name	Value map
APC Battery Replacement Status	
APC Battery Status	
Dell Open Manage System Status	
Host status	0 > Up 2 > Unreachable
HP Insight System Status	
Service state	0 > Down 1 > Up
Windows service state	0 > Running 1 > Paused 3 > Pause pending 4 > Continue pending 5 > Stop pending 6 > Stopped 7 > Unknown 255 > No such service

At the bottom of the page, it says "ZABBIX 1.6 Copyright 2001-2008 by SIA Zabbix" and "Connected as 'Admin'".

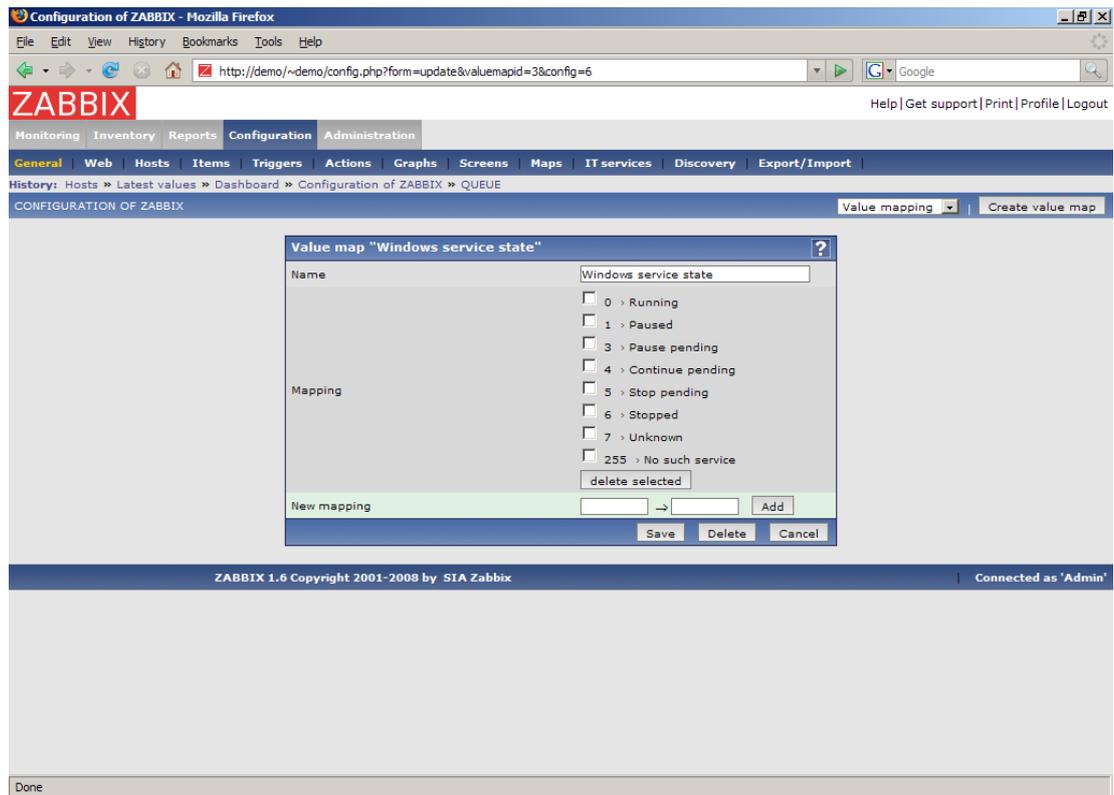
たとえば、アイテムが値として「0」または「1」を取る場合、以下のように値のマッピングを定義すると、ユーザが読みやすい文字列で表現することができます。

'0' => 'Not Available'

'1' => 'Available'

注: 値のマッピングを使用できるのは、タイプが「数値(整数)」のアイテムだけです。

値のマッピングの定義

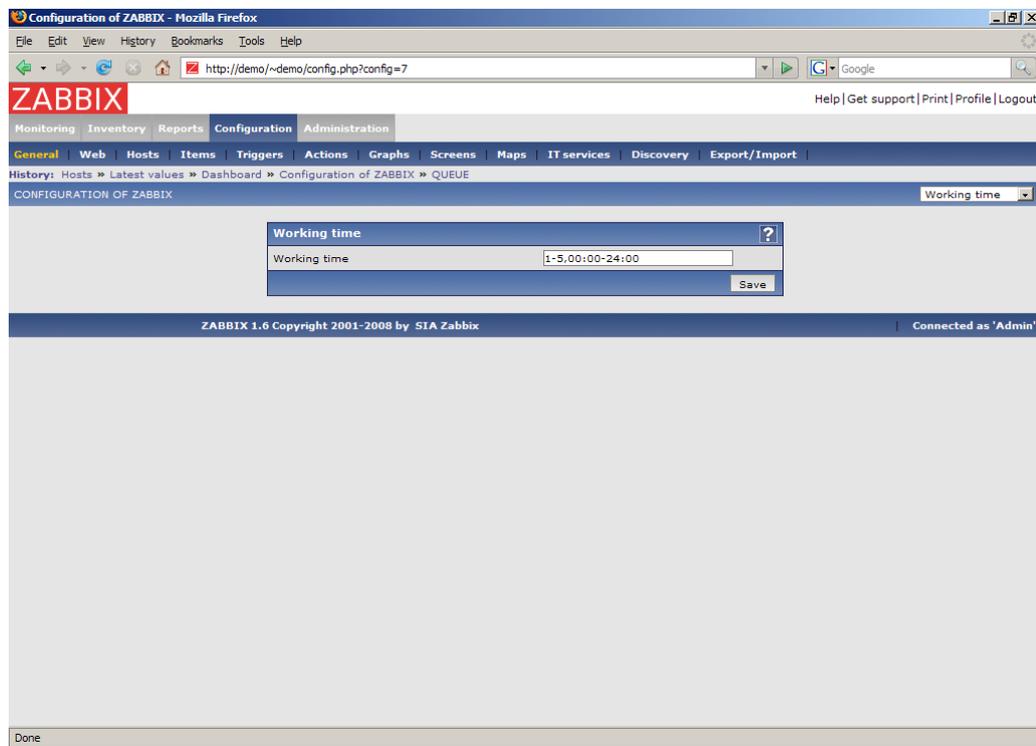


値のマッピングのパラメータを以下に示します。

パラメータ	説明
[名前]	値のマッピングの一意な名前。
[マッピング]	マッピングのリスト。
[新規マッピング]	追加するマッピング。

17.2.1.6. ワーキングタイム

[ワーキングタイム]は、ワーキングタイムを定義するパラメータで、システム全体から参照されます。
 現在、ワーキングタイムはグラフのみに白背景で表示され、非ワーキングタイムは灰色背景で表示されます。



ワーキングタイムは以下の形式で指定します。

dd-dd, hh:mm-hh:mm;dd-dd, hh:mm-hh:mm, ...

形式	説明
dd	曜日: 1 - 月曜日、2 - 火曜日、...、7 - 日曜日
hh	時間: 00-24
mm	分: 00-59

何も指定しない場合、「01-07,00:00-23:59」が設定されたものとみなされます。

以下に例を示します。

1-5,09:00-18:00

1-5,09:00-18:00;6-7,10:00-16:00

17.2.1.7. その他

取得不可アイテムの更新

ユーザパラメータのエラーやエージェントによってサポートされていないなどの理由で取得不可になるアイテムがあります。

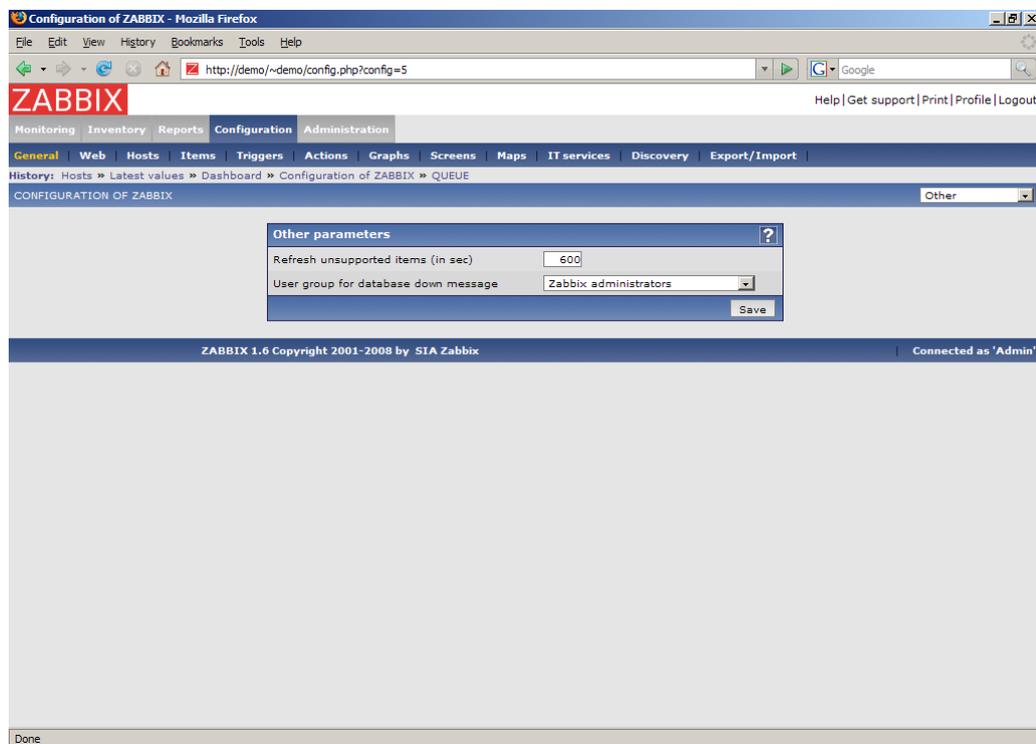
ZABBIX では、設定によって、取得不可アイテムを定期的にアクティブにすることができます。

データベースウォッチドッグ

ZABBIX サーバの可用性は、バックエンドデータベースの可用性にほぼ依存しています。データベースが動作しなければ、ZABBIX サーバは機能しません。

データベースウォッチドッグは ZABBIX サーバの特別なプロセスで、致命的な障害が発生したことを ZABBIX 管理者に通知します。

データベースが停止すると、ウォッチドッグからユーザグループに通知が送信されます。ZABBIX サーバは停止しませんが、データベースが復旧して処理を続行できるようになるまで待機します。



パラメータ	説明
[取得不可アイテムの更新間隔(秒)]	取得不可アイテムをアクティブにする間隔(秒単位)を指定します。0を指定すると、自動でアクティブにする処理は行われません。 プロキシでは、取得不可アイテムは10分間隔でチェックされます。 このパラメータをプロキシに対して設定することはできません。
[データベース停止メッセージの送信先グループ]	アラームメッセージの送信先ユーザグループまたは[なし]を設定します。

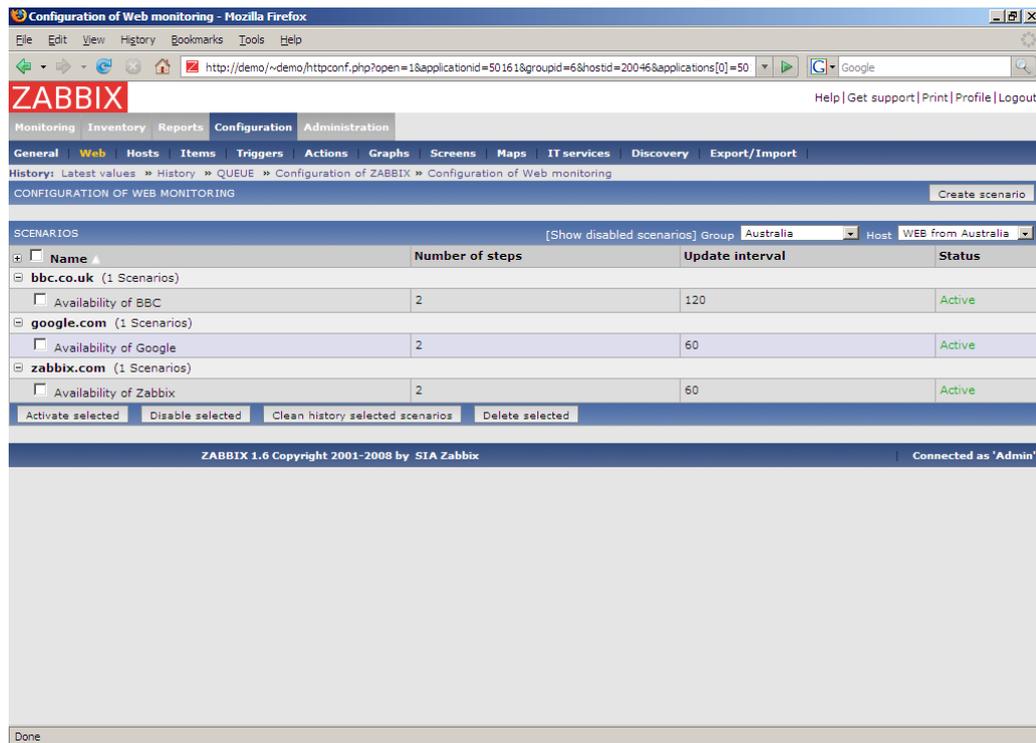
注: データベースウォッチドッグは、MySQLでのみサポートされています。

17.2.2. ウェブ

このスクリーンは、ウェブシナリオの監視を管理するために使用します。

ウェブシナリオのリスト

アクティブなウェブシナリオのリストが表示されます。

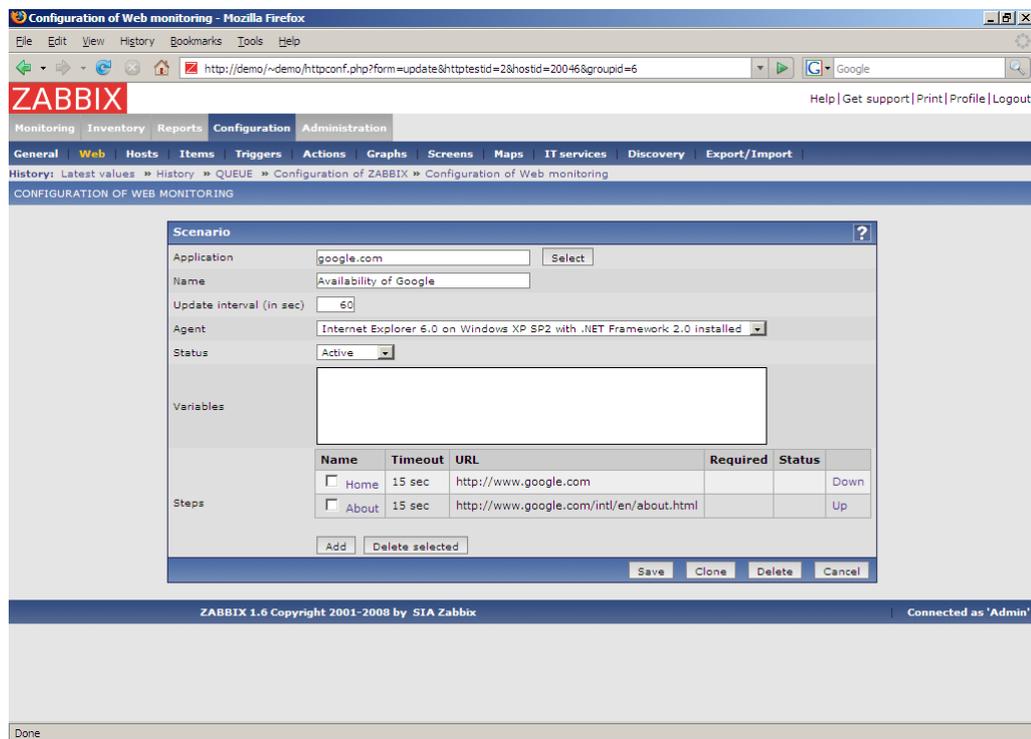


表示されるデータを以下に示します。

パラメータ	説明
[名前]	ウェブシナリオの一意的な名前。
[ステップ数]	シナリオを構成する個々のステップ(HTTP要求)の数。
[更新間隔(秒)]	ウェブシナリオの実行頻度。
[ステータス]	シナリオのステータス。 [有効] - シナリオは有効です。 [無効] - シナリオは無効です。デフォルトでは、無効のシナリオは表示されません。

ウェブシナリオの設定

このスクリーンは、個々のウェブシナリオのパラメータを定義するために使用します。



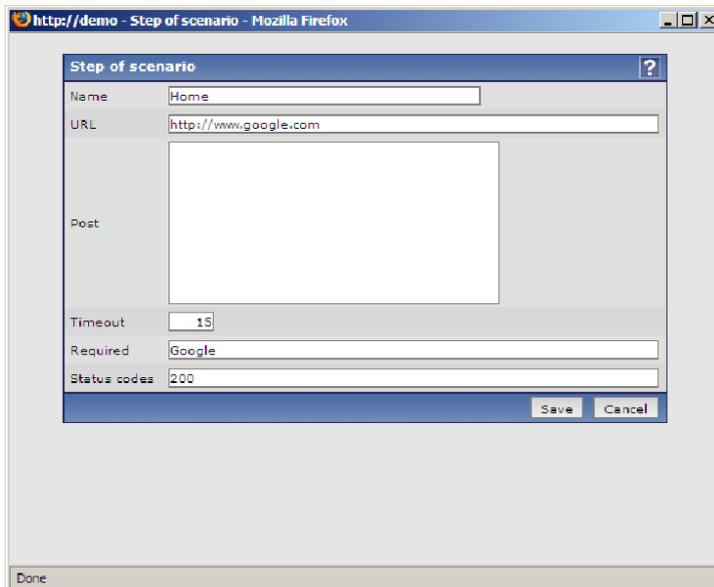
設定パラメータを以下に示します。

パラメータ	説明
[アプリケーション]	シナリオのリンク先のホストアプリケーション。
[名前]	ウェブシナリオの一意な名前。
[更新間隔(秒)]	ウェブシナリオの実行頻度。
[エージェント]	クライアントエージェントの文字列。ZABBIXにより、Firefox、MS Explorerなどのアプリケーションが疑似的に指定されます。 各種ブラウザに対して異なるコンテンツを返すウェブサイトを扱う場合に役立ちます。
[ステータス]	シナリオのステータス。 [有効] - シナリオは有効です。 [無効] - シナリオは無効です。デフォルトでは、無効のシナリオは表示されません。
[変数]	シナリオのステップ(URLおよびPost変数)で使用可能な変数(マクロ)のリスト。 以下の形式で指定します。 {macro1}=value1 {macro2}=value2 以下に例を示します。 username=Alexei password=kj3h5kJ34bd マクロは{username}および{password}として参照できます。こ

	これらの変数は自動的に実際の値に置き換えられます。
[ステップ]	シナリオで実行されるステップのリスト。 [名前] - ステップ名 [タイムアウト] - タイムアウト [URL] - 接続先 [要求文字列] - 要求文字列 [ステータス] - ステップのステータス

ウェブステップの設定

このスクリーンは、ウェブシナリオの個々のステップのパラメータを定義するために使用します。



設定パラメータを以下に示します。

パラメータ	説明
[名前]	一意なステップ名。
[URL]	接続してデータを取得するURL。以下に例を示します。 http://www.zabbix.com https://www.google.com
[POST]	POST変数のリスト。[URL]パラメータでGET変数を渡すことができます。
[タイムアウト]	タイムアウト時間(秒単位)。この時間以内にURLの処理を完了する必要があります。
[要求文字列]	要求文字列。取得されるコンテンツ(HTML)にはこの文字列が含まれている必要があります。含まれていない場合は、ステップが失敗します。 何も指定しなければ、チェックは実行されません。
[ステータスコード]	予期されるHTTPコードのリスト。ZABBIXでリストに存在しないコードが取得されると、ステップが失敗します。 何も指定しなければ、チェックは実行されません。 例: 200,201,210-299

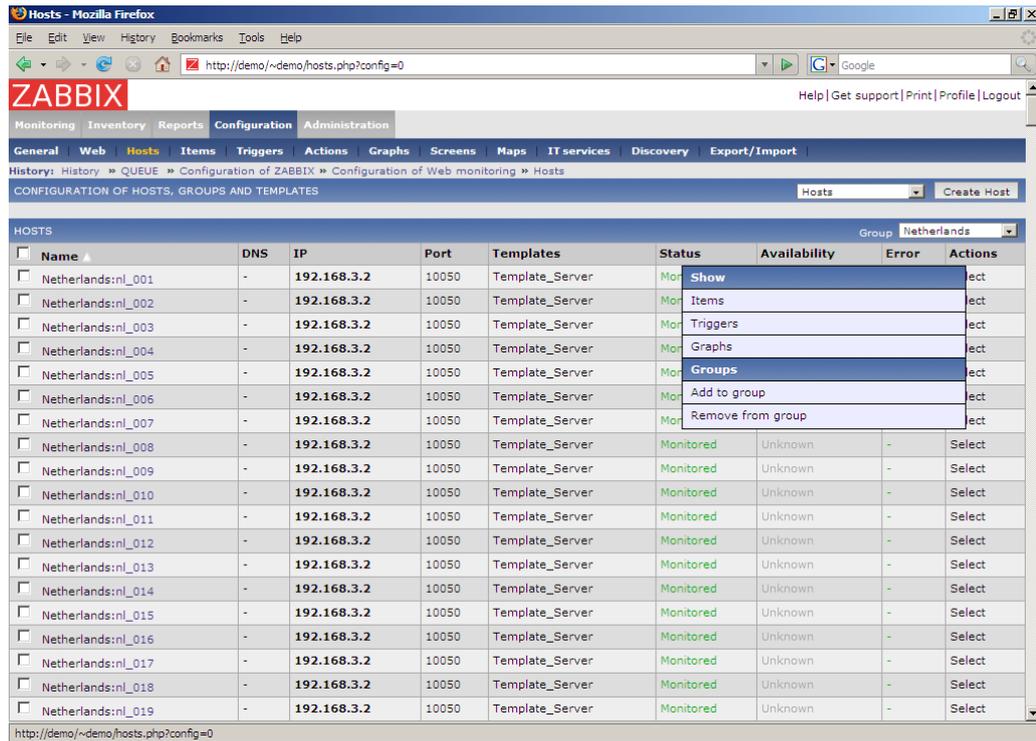
17.2.3. ホスト

17.2.3.1. ホスト

このスクリーンは、ホスト関連の情報を管理するために使用します。

ホストのリスト

このスクリーンには、監視対象のホストのリストが表示されます。



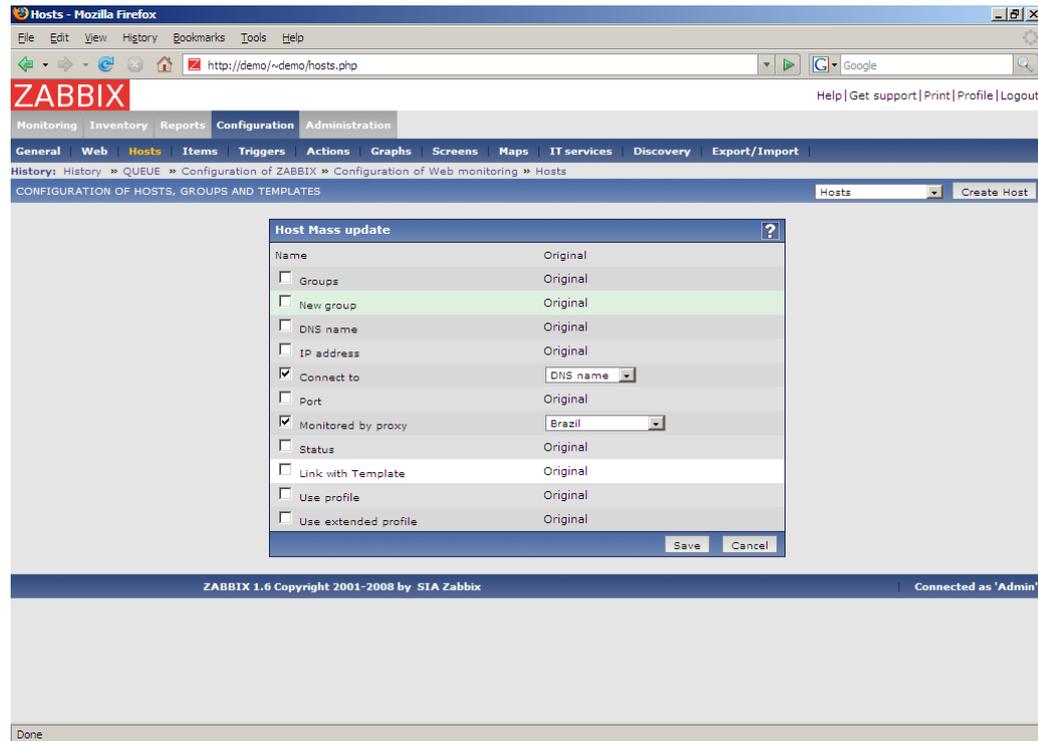
表示されるデータを以下に示します。

パラメータ	説明
[名前]	一意なホスト名。
[DNS]	ホストDNS名(使用されている場合)。
[IPアドレス]	ホストIPアドレス(使用されている場合)。
[ポート]	ZABBIXエージェントのポート番号。エージェントが使用されていない場合は無視されます。
[テンプレート]	ホストにリンクされたテンプレートのリスト。
[ステータス]	ホストのステータス。 [有効] - ホストは有効で、監視されています。 [無効] - ホストは無効です。
[状態]	エージェント(Zabbix、SNMP)の可用性。 [監視中] - エージェントはアップ状態で、動作しています。

	[不明] - エージェントは利用できません。
[エラー]	エージェントベースのチェックの使用に関するエラー。

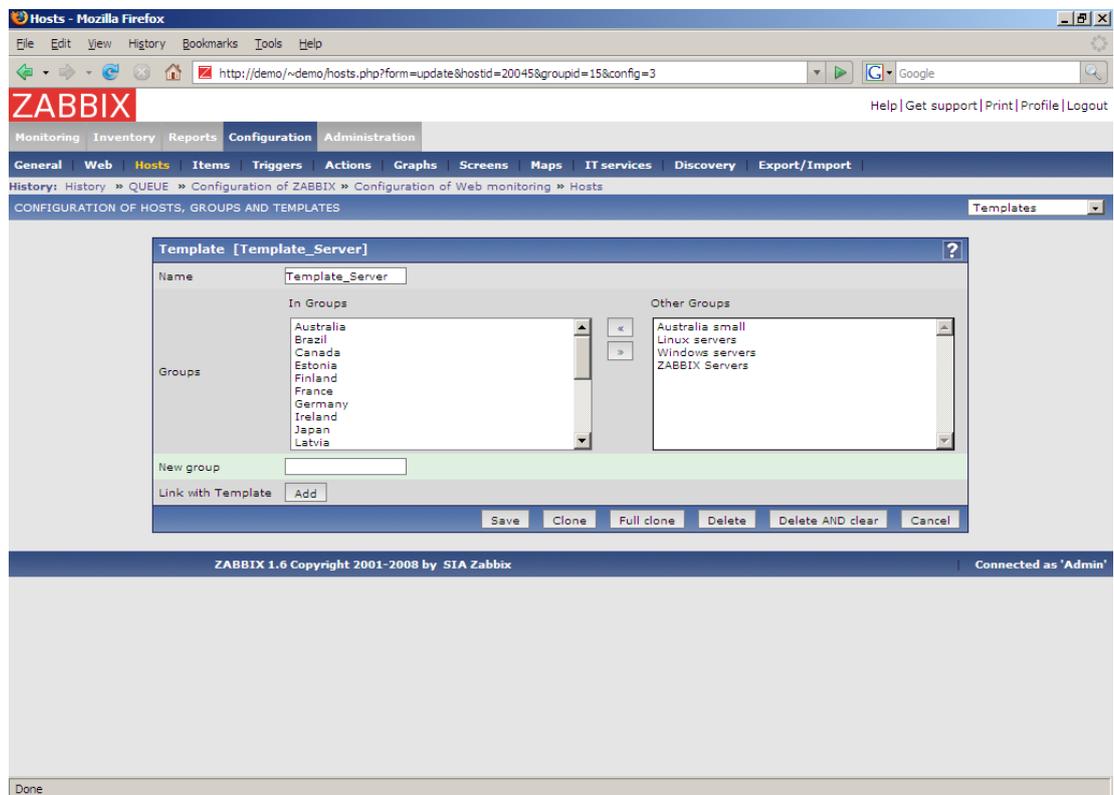
ホストの一括更新スクリーン

このスクリーンにアクセスするには、ホストを選択し、[一括更新]ボタンをクリックします。このスクリーンでは、複数のホストの属性を非常に効率的に変更できます。



ホストの設定

このスクリーンには、ホストの詳細が表示されます。



設定パラメータを以下に示します。

パラメータ	説明
[名前]	一意なホスト名。
[グループ]	ホストが属するホストグループのリスト。
[新規グループ作成]	新しいグループを作成してホストにリンクできます。何も指定しなければ、無視されます。
[DNS名]	ホストDNS名(省略可能)。
[IPアドレス]	ホストIPアドレス(省略可能)。
[接続方法]	ZABBIXサーバでは、以下の設定を使用してエージェントからデータが取得されます。 [DNS名] - ホストDNS名に接続します。 [IPアドレス] - ホストIPに接続します(推奨)。
[ポート]	ZABBIXエージェントのTCPポート番号。デフォルト値は10050です。
[プロキシによる監視]	ホストは、ZABBIXサーバまたはプロキシの1つで監視できます。 [(プロキシなし)] - ホストはZABBIXサーバによって監視されます。

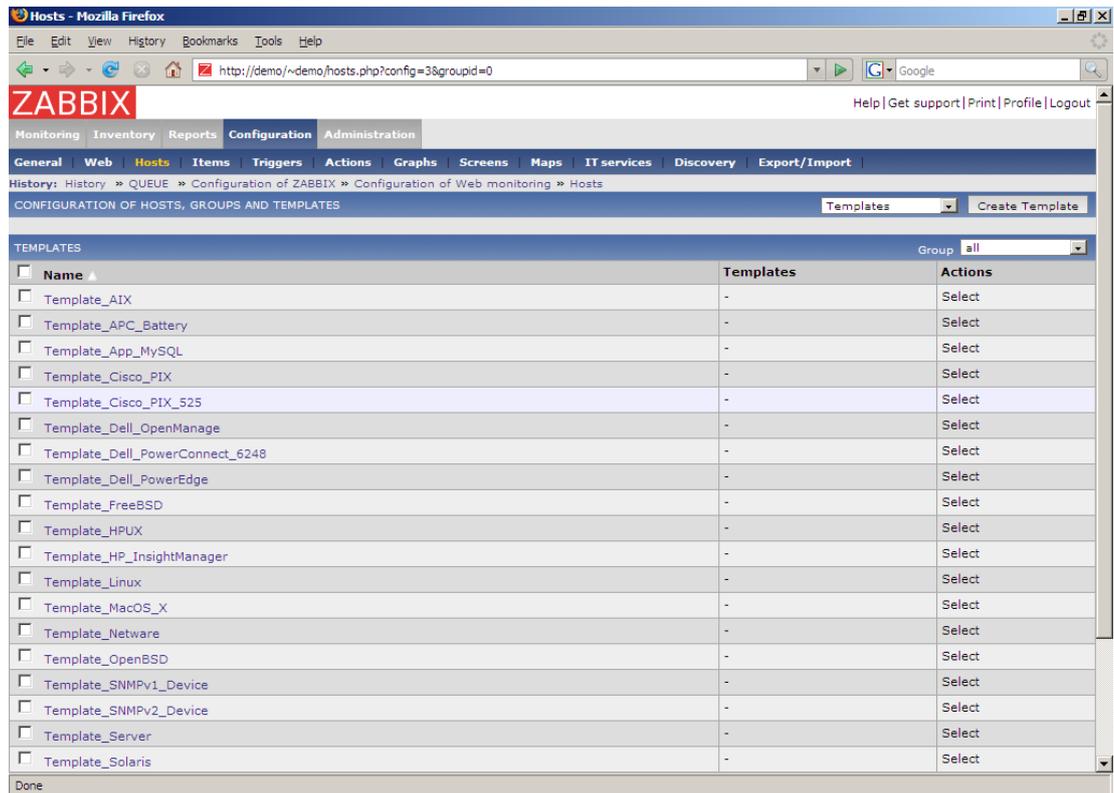
	[プロキシ名] - ホストはプロキシ([プロキシ名] で指定)によって監視されます。
[ステータス]	ホストのステータス。 [有効] - ホストが有効で、監視できる状態にあります。 [無効] - ホストが無効で、監視できません。
[テンプレートとのリンク]	ホストを1つまたは複数のテンプレートとリンクします。アイテム、トリガー、およびグラフに関する情報はテンプレートから継承されます。 [リンクを削除] - テンプレートからリンクを削除しますが、アイテム、トリガー、およびグラフに関する情報は保持します。 [リンクと保存データを削除] - テンプレートからリンクを削除するとともに、テンプレートから継承された情報も全て削除します。
[IPMIを使用]	対象のホストに対してIPMI管理機能を有効にします。
[IPMI IPアドレス]	IPMI管理デバイスのIPアドレス。
[IPMIポート]	IPMIデバイスのポート番号。
[IPMI特権レベル]	ここでデフォルトの設定([ユーザ])を保持します。
[IPMIユーザ名]	認証用のユーザ名。
[IPMIパスワード]	認証用のパスワード。
[プロファイルを使用]	ホストプロファイルの使用を有効または無効にします。
[拡張プロファイルを使用]	拡張ホストプロファイルの使用を有効または無効にします。

17.2.3.2. テンプレート

このスクリーンは、ホストテンプレートを管理するために使用します。

テンプレートのリスト

このスクリーンには、テンプレートのリストが表示されます。

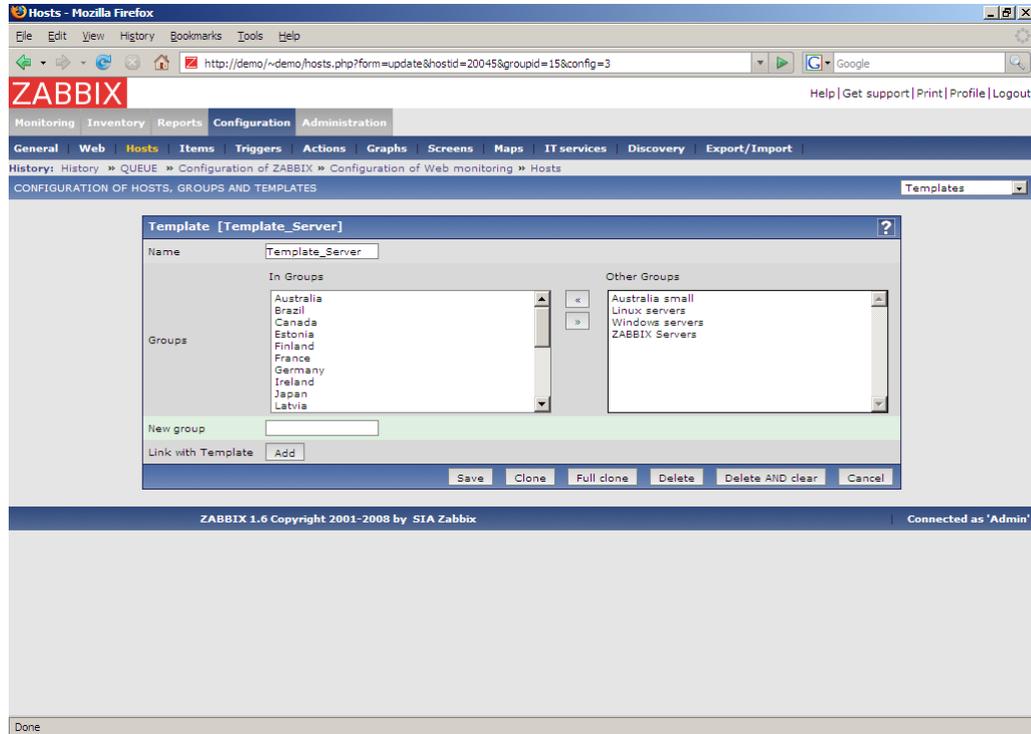


表示されるデータを以下に示します。

パラメータ	説明
[名前]	テンプレート名。
[テンプレート]	対象のテンプレートにリンクされているホストのリスト。

テンプレートの設定

このスクリーンには、テンプレートの詳細が表示されます。



設定パラメータを以下に示します。

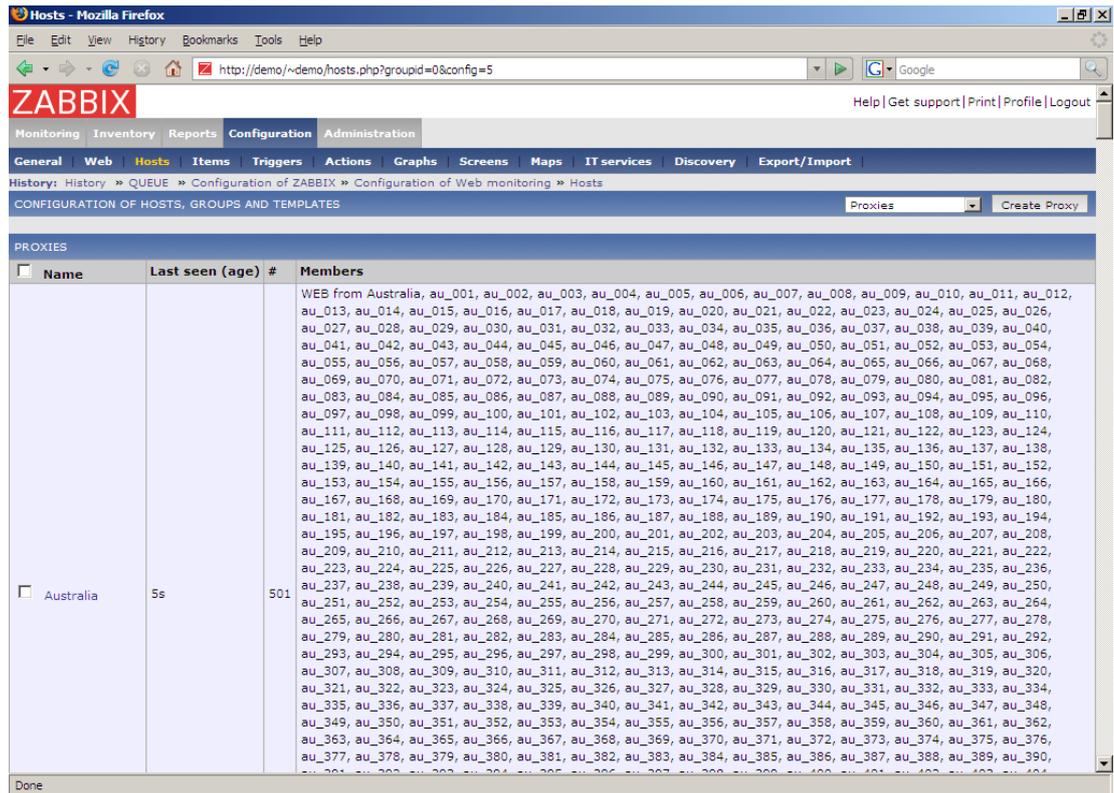
パラメータ	説明
[名前]	一意なテンプレート名。
[グループ]	テンプレートが属するホストグループのリスト。
[新規グループ作成]	新しいグループを作成してテンプレートにリンクできます。何も指定しなければ、無視されます。
[テンプレートとのリンク]	テンプレートを一つまたは複数のテンプレートとリンクします。アイテム、トリガー、およびグラフに関する情報はテンプレートから継承されます。

17.2.3.3. プロキシ

このスクリーンは、プロキシを管理するために使用します。

プロキシのリスト

このスクリーンには、プロキシのリストが表示されます。

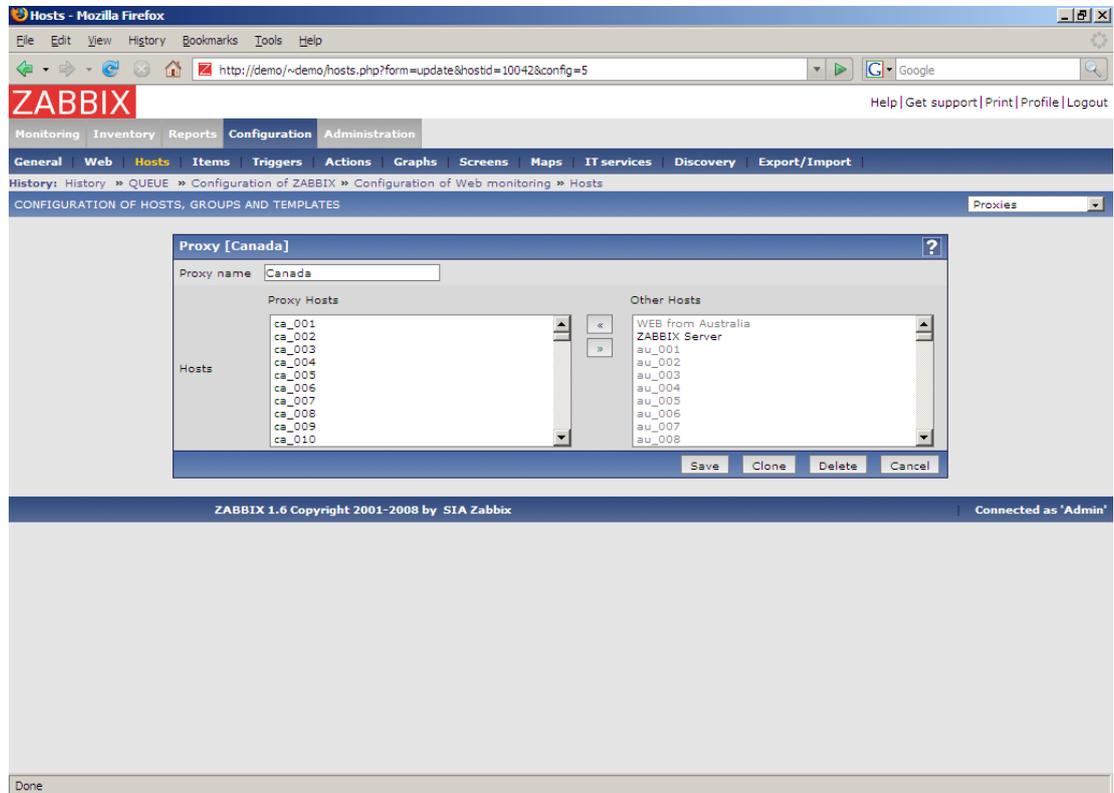


表示されるデータを以下に示します。

パラメータ	説明
[名前]	一意なプロキシ名。
[最新データ受信時刻(経過時間)]	プロキシからハートビートメッセージまたはデータを受信した最終時刻。
[メンバー]	対象のプロキシによって監視されているホストのリスト。

プロキシの設定

このスクリーンには、プロキシの詳細が表示されます。



設定パラメータを以下に示します。

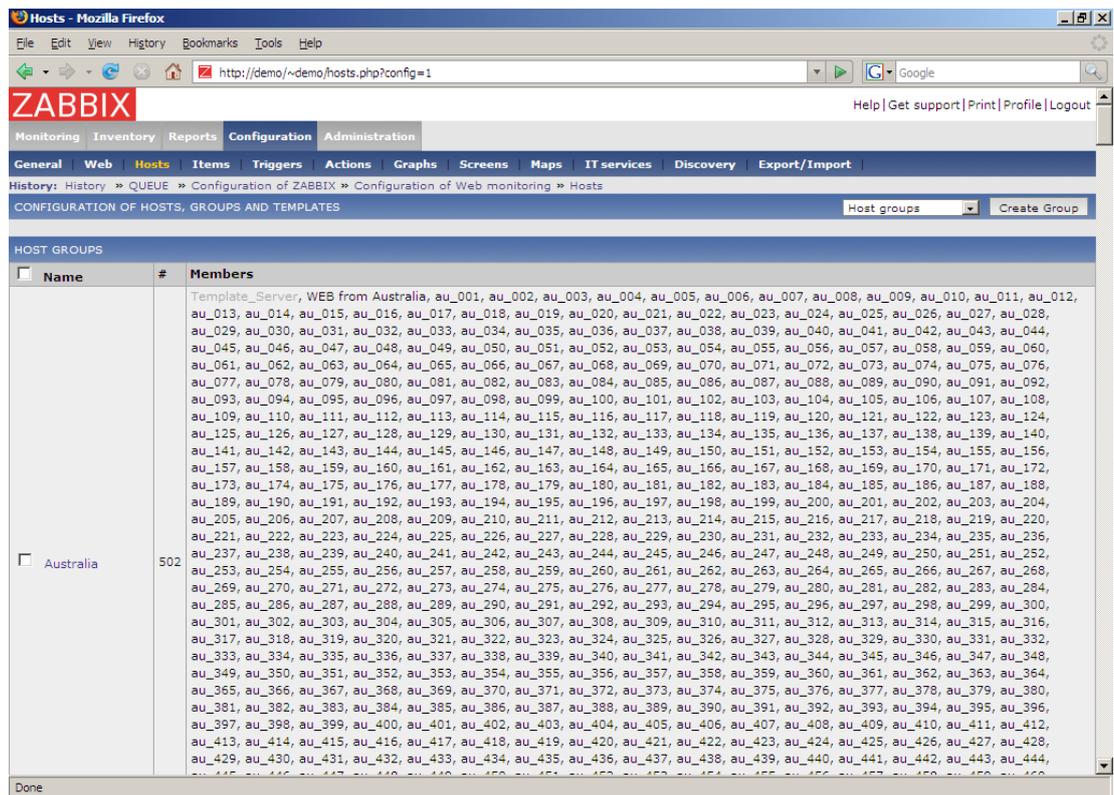
パラメータ	説明
[プロキシ名]	一意なプロキシ名。
[ホスト]	対象のプロキシによって監視されているホストのリスト。

17.2.3.4. ホストグループ

このスクリーンは、ホストグループを管理するために使用します。

ホストグループのリスト

このスクリーンには、ホストグループのリストが表示されます。

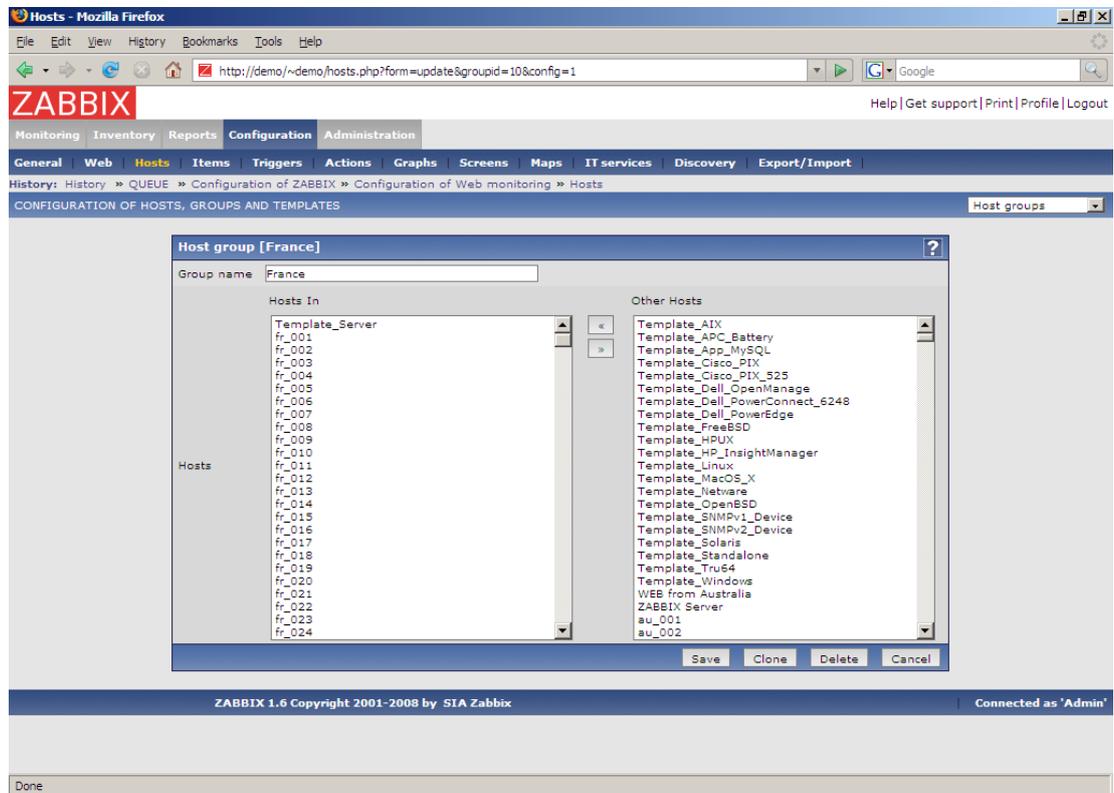


表示されるデータを以下に示します。

パラメータ	説明
[名前]	ホストグループ名。
[#]	グループメンバー(ホスト)の数。
[メンバー]	ホストグループメンバーのリスト。

ホストグループの設定

このスクリーンには、ホストグループの詳細が表示されます。



設定パラメータを以下に示します。

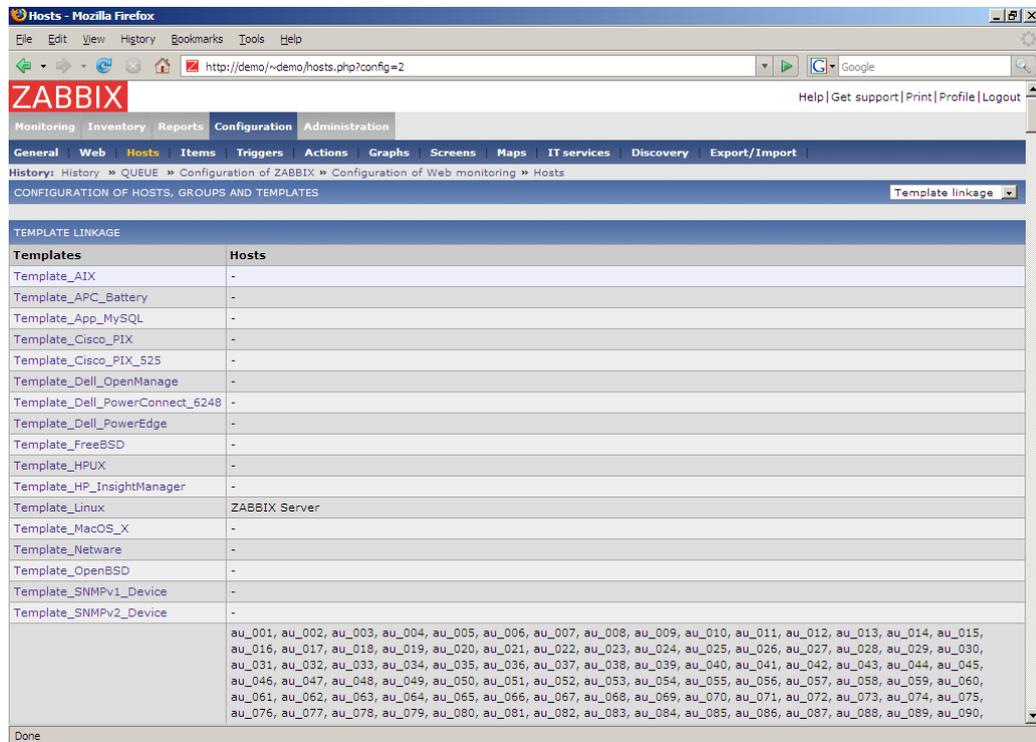
パラメータ	説明
[グループ名]	一意なホストグループ名。
[ホスト]	ホスト(グループのメンバー)のリスト。

17.2.3.5. テンプレートのリンク

このスクリーンは、ホストテンプレートのリンクを管理するために使用します。

テンプレートのリスト

このスクリーンには、テンプレートおよびそれにリンクされているホストのリストが表示されます。

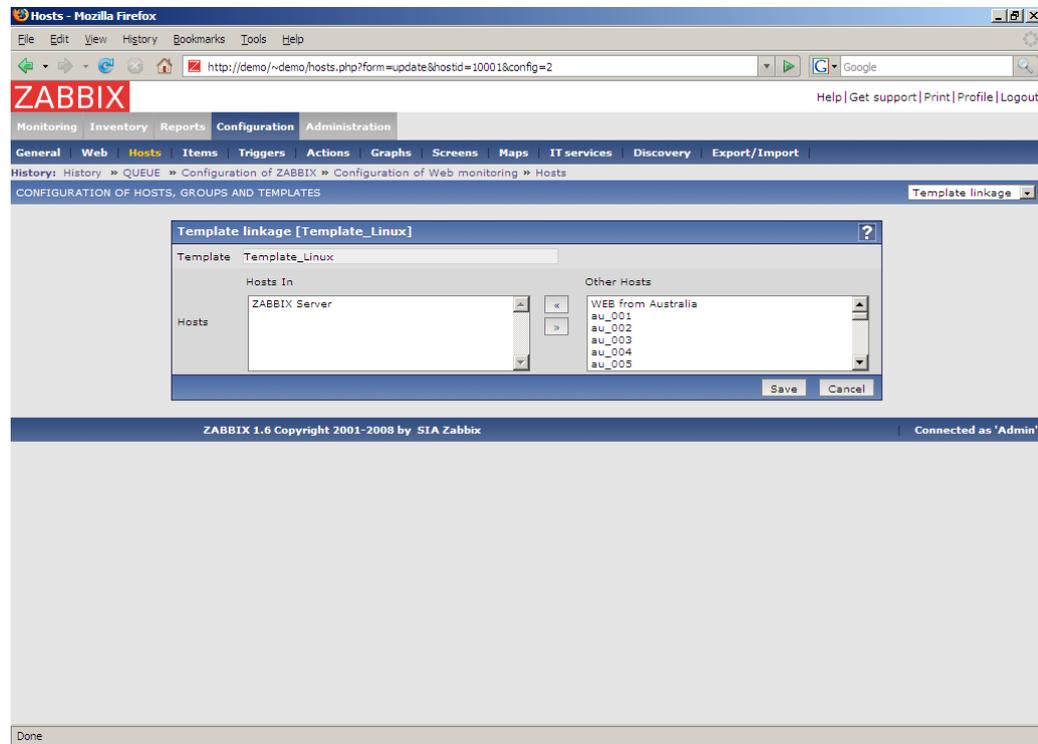


表示されるデータを以下に示します。

パラメータ	説明
[テンプレート]	ホストテンプレート名。
[ホスト]	テンプレートにリンクされているホストのリスト。

テンプレートのリンク

このスクリーンには、ホストテンプレートのリンクの管理情報が表示されます。



設定パラメータを以下に示します。

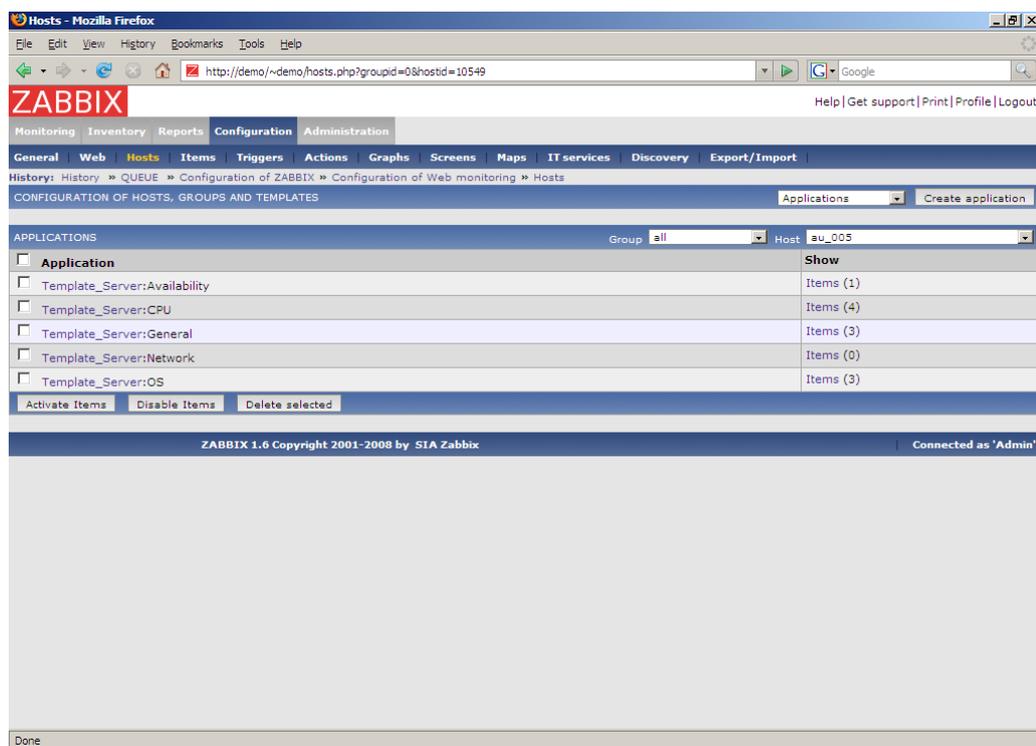
パラメータ	説明
[テンプレート]	テンプレート名。
[ホスト]	テンプレートにリンクされているホストのリスト。

17.2.3.6. アプリケーション

このスクリーンは、アプリケーションを管理するために使用します。

アプリケーションのリスト

このスクリーンには、アプリケーションのリストが表示されます。

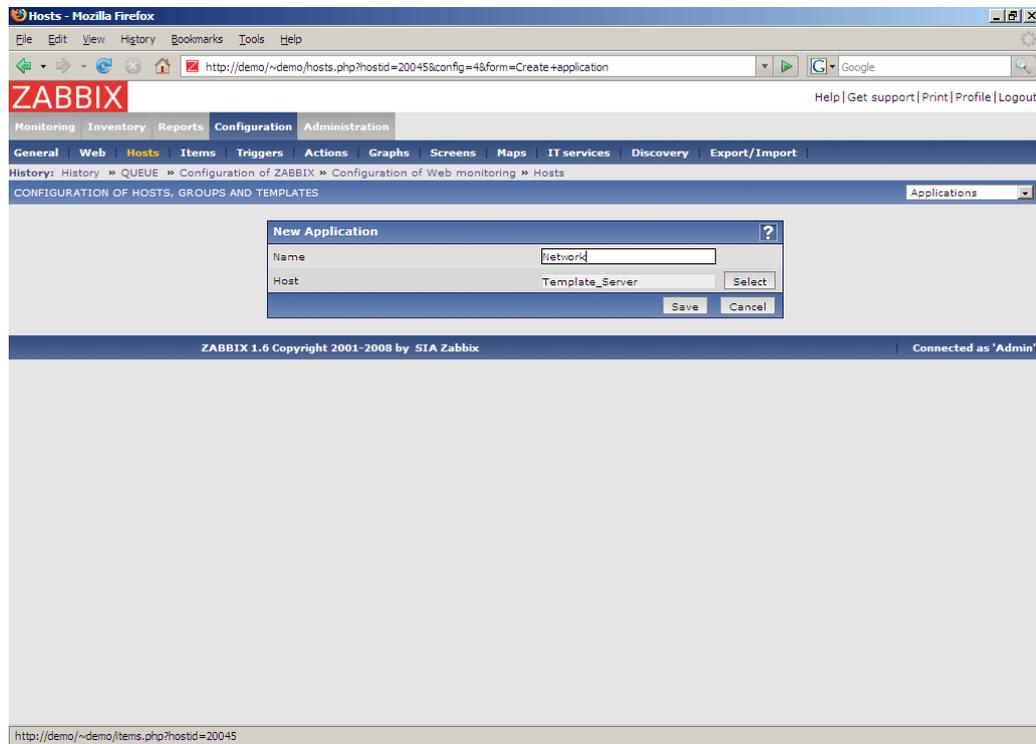


表示されるデータを以下に示します。

パラメータ	説明
[アプリケーション]	アプリケーション名。
[表示]	ホストアイテムへのリンク。アイテム数(アプリケーションのメンバー)も表示されます。

アプリケーションの設定

このスクリーンには、アプリケーションの管理情報が表示されます。



設定パラメータを以下に示します。

パラメータ	説明
[名前]	アプリケーション名。1つのホスト内で一意である必要があります。
[ホスト]	アプリケーションのリンク先のホスト名。

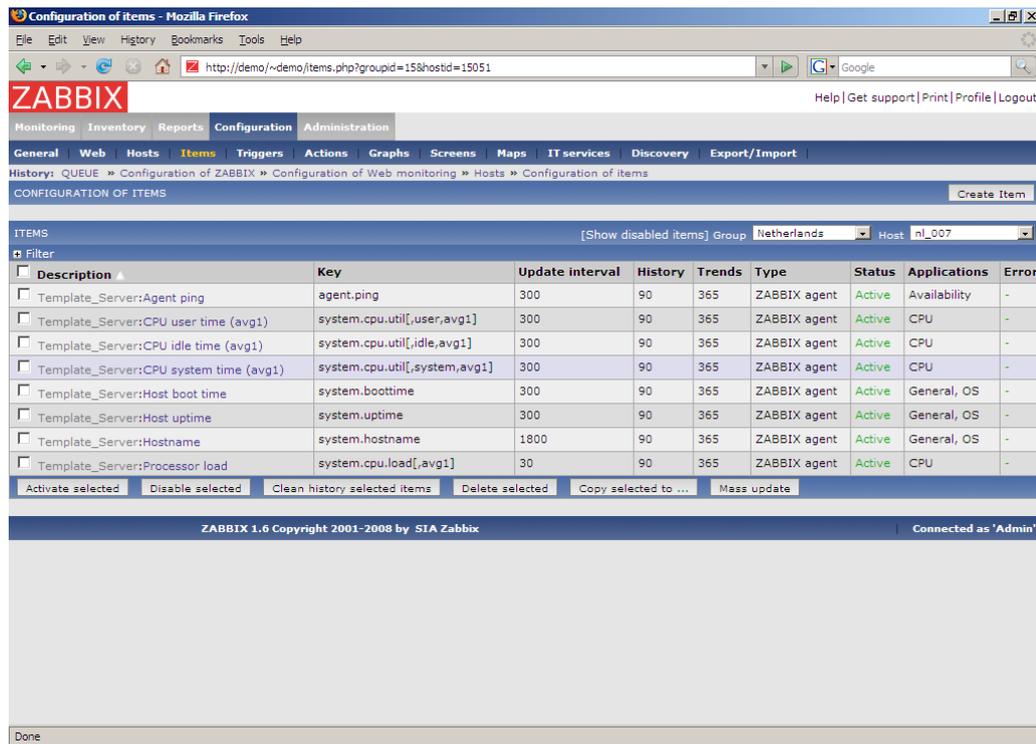
17.2.4. アイテム

17.2.4.1. アイテム

このスクリーンは、アイテム関連の情報を管理するために使用します。

アイテムのリスト

このスクリーンには、ホストにリンクされているアイテムのリストが表示されます。

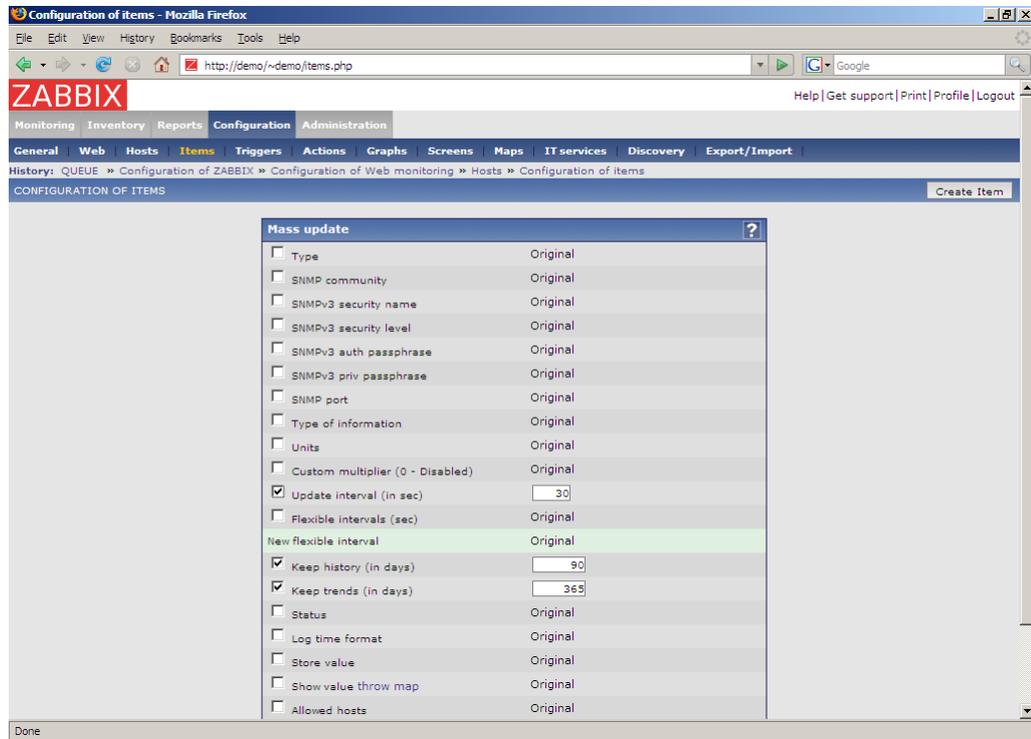


表示されるデータを以下に示します。

パラメータ	説明
[名前]	アイテムの説明(名前)。
[キー]	一意なアイテムキー。
[更新間隔(秒)]	チェックの頻度。
[ヒストリ]	ZABBIXで詳細なヒストリデータを保持する日数。
[トレンド]	ZABBIXでトレンドデータを保持する日数。
[タイプ]	アイテムのタイプ。
[ステータス]	アイテムのステータス。
[アプリケーション]	アイテムが属するアプリケーションのリスト。
[エラー]	対象のアイテムに関連するエラー。

アイテムの一括更新スクリーン

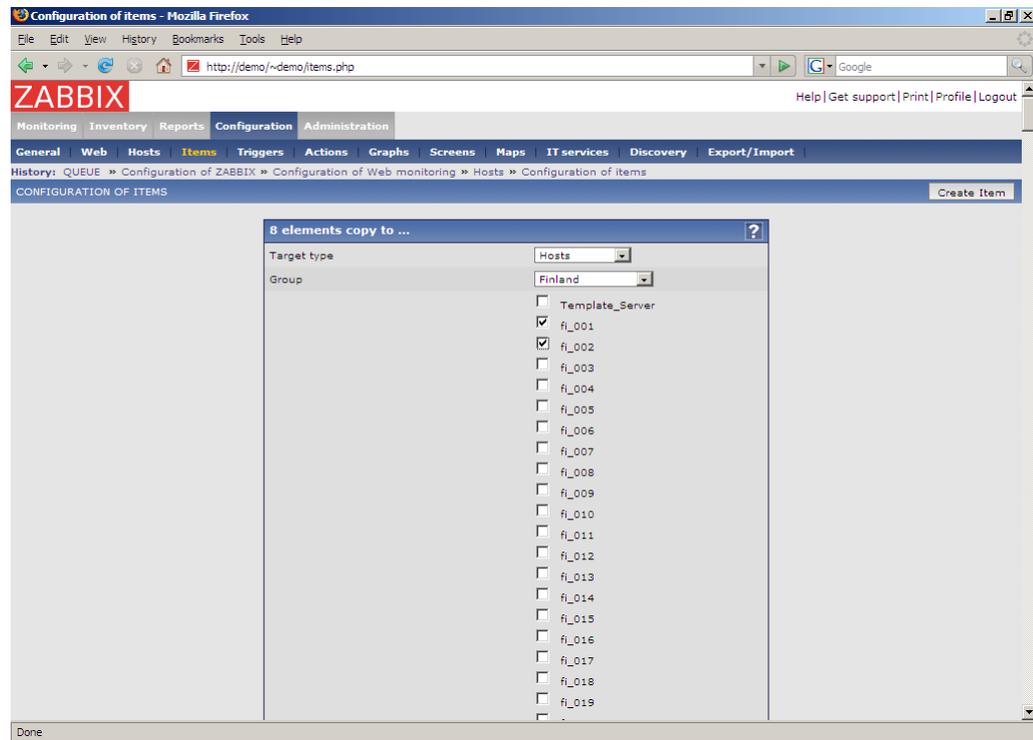
このスクリーンにアクセスするには、アイテムを選択し、[一括更新]ボタンをクリックします。このスクリーンでは、複数のアイテムの属性を非常に効率的に変更できます。



変更するパラメータをクリックし、新しい値を入力して[保存]をクリックします。

選択をコピー

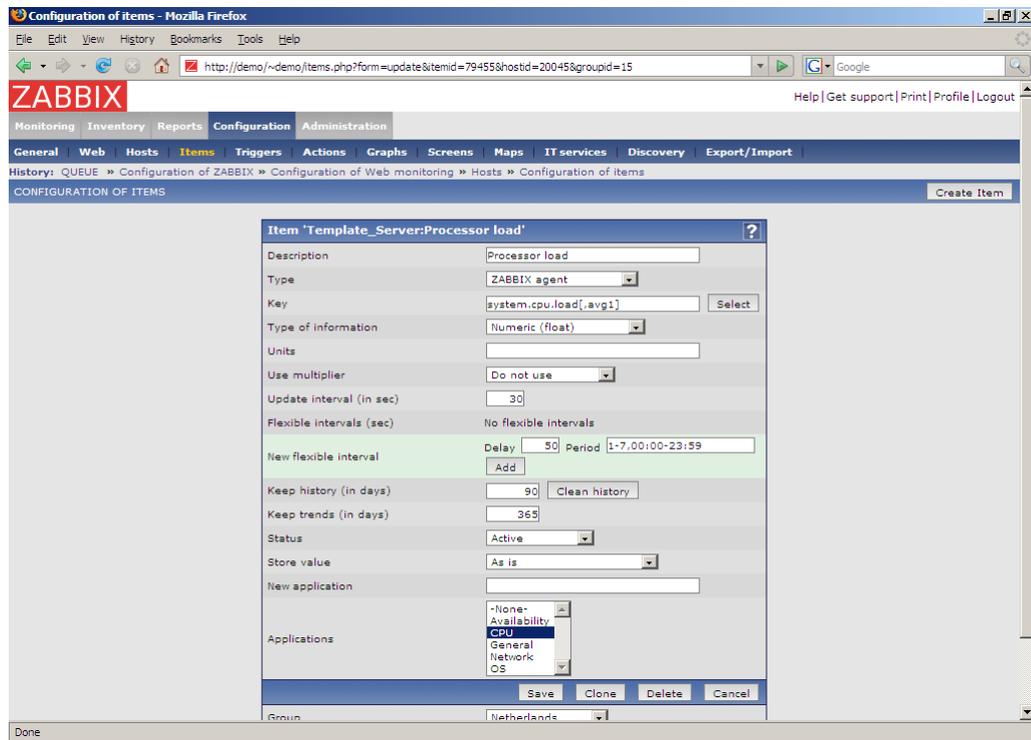
このスクリーンでは、選択したアイテムを複数のホストにコピーできます。



アイテムをコピーするホストを選択し、[コピー]をクリックします。

アイテムの設定

このスクリーンには、単一のアイテムの設定情報が表示されます。



アイテムの属性を以下に示します。

パラメータ	説明
[名前]	アイテムの説明。以下に示すマクロを使用できます。 \$1 - アイテムキーの1番目のパラメータ \$2 - 2番目のパラメータ \$N - N番目のパラメータ 例: Free disk space on \$1 この説明は、アイテムキーが「vfs.fs.size[/,free]」の場合、自動的に「Free disk space on /」に変換されます。
[タイプ]	アイテムのタイプ。各タイプの詳細については、以降のセクションを参照してください。
[キー]	アイテムのキー。単一ホスト内で一意である必要があります。キータイプが[ZABBIXエージェント]、[ZABBIXエージェント(アクティブ)]、[シンプルチェック]、または[ZABBIXアグリゲート]の場合、エージェントまたはZABBIXサーバがキーの値をサポートしている必要があります。
[データ型]	受信データのタイプ。 [数値(整数)] - 64ビット符号なし整数値 [数値(浮動小数)] - 浮動小数点値 [文字] - 最大255バイトの文字(文字列)データ [ログ] - ログファイル。キーlog[]を使用する場合はこの値を設定する必要があります。 [テキスト] - 任意の長さのテキスト
[データタイプ]	期待するデータタイプを指定するために整数アイテムに対して使用されます。

	<p>[10進(Decimal)] - 10進形式のデータ [8進(Octal)] - 8進形式のデータ [16進(Hexadecimal)] - 16進形式のデータ Zabbixでは、数値への変換が自動的に実行されます。 このパラメータは、バージョン1.8以降でサポートされていません。</p>
[単位]	<p>値が設定されている場合、受信した全ての値にポストフィクスとして単位が追加され、必要に応じてプリフィクス「K」、「M」、または「G」が追加されます(1K=1024で換算)。 たとえば、単位に「B」が設定されている場合、以下のように表示されます。 1->1B 1024->1KB 1536->1.5KB 以下に示す単位が設定された場合、特別な処理が行われ ます。 b、bps - ビットの場合は特別に「1K=1000」で換算されます。 unixtime - 「yyyy.mm.dd hh:mm:ss」に変換されます。 uptime - 「hh:mm:ss」または「N days, hh:mm:ss」に変換されます。パラメータは1970年1月1日からの経過秒数として処理されます。 s - 「yyymmddhhmm」に変換されます。パラメータは1970年1月1日からの経過秒数として処理されます。例: 2y10m14d3h54m1s</p>
[乗数を使用]	<p>受信した値に対する前処理を指定します。 [使用しない] - 前処理を行いません。 [乗数] - 受信した値に[乗数]で定義された値を乗じます。KBやMbpsの単位で受信した値をそれぞれBやbps単位に変換する場合にこのオプションを指定します。このオプションを指定しないと、プリフィクス(「K」、「M」、「G」)が正しく設定されません。</p>
[乗数]	<p>受信した全ての値にこの整数値または浮動小数点値を乗じます。</p>
[更新間隔(秒)]	<p>このアイテムを更新する間隔(秒単位)を指定します。</p>
[例外の更新間隔(秒)]	<p>更新間隔の例外のリスト。例: 「10 sec, 1-5,09:00-18:00」と設定すると、ワーキングタイム中は10秒間隔で更新され、それ以外の時間帯はデフォルトの更新間隔で更新されます。 期間の指定形式: dd-dd, hh:mm-hh:mm; dd-dd, hh:mm-hh:mm 例: 1-5,09:00-18:00;6-7,10:00-12:00 1 - 月曜日、...、7 - 日曜日</p>
[履歴の保存期間(日)]	<p>詳細履歴をデータベースに保存する期間を設定します。保存期間を超えたデータはhousekeeperによって削除されます。</p>
[トレンドの保存期間(日)]	<p>累計(毎時の最小、最大、平均、個数)の詳細履歴をデータベースに保存する期間を設定します。保存期間を超えたデータはhousekeeperによって削除されます。</p>
[ステータス]	<p>[有効] - 有効(通常)状態。ZABBIXはこのアイテムを処理します。 [無効] - 無効状態。ZABBIXはこのアイテムを無視します。 [取得不可] - ZABBIXエージェントまたはSNMPエージェントは</p>

	このアイテムを取得しません。このアイテムは処理されませんが、ZABBIXが定期的にステータスを[有効]にしようと試みるように設定することができます。
[保存時の計算]	<p>[なし] - 前処理なし</p> <p>[差分/時間] - $(value - prev_value) / (time - prev_time)$と評価されます。</p> <p>value - 現在値</p> <p>prev_value - 前回受信した値</p> <p>time - 現在のタイムスタンプ</p> <p>prev_time - 前回受信した値のタイムスタンプ</p> <p>この設定は、定常的に増え続ける値について1秒あたりの増加量を計算する場合に非常に便利です。</p> <p>[差分] - $(value - prev_value)$と評価されます。</p> <p>value - 現在値</p> <p>prev_value - 前回受信した値</p>
[値のマッピングの使用]	<p>このアイテムに値のマッピングを適用します。値のマッピングはデータ表示にのみ適用され、受信した値が変更されることはありません。</p> <p>使用できるのは、整数値アイテムだけです。</p> <p>例: 「Windows service states」</p>
[アプリケーション]	アイテムを1つまたは複数のアプリケーションにリンクします。

アイテムの詳細については、本書の他のセクションを参照してください。

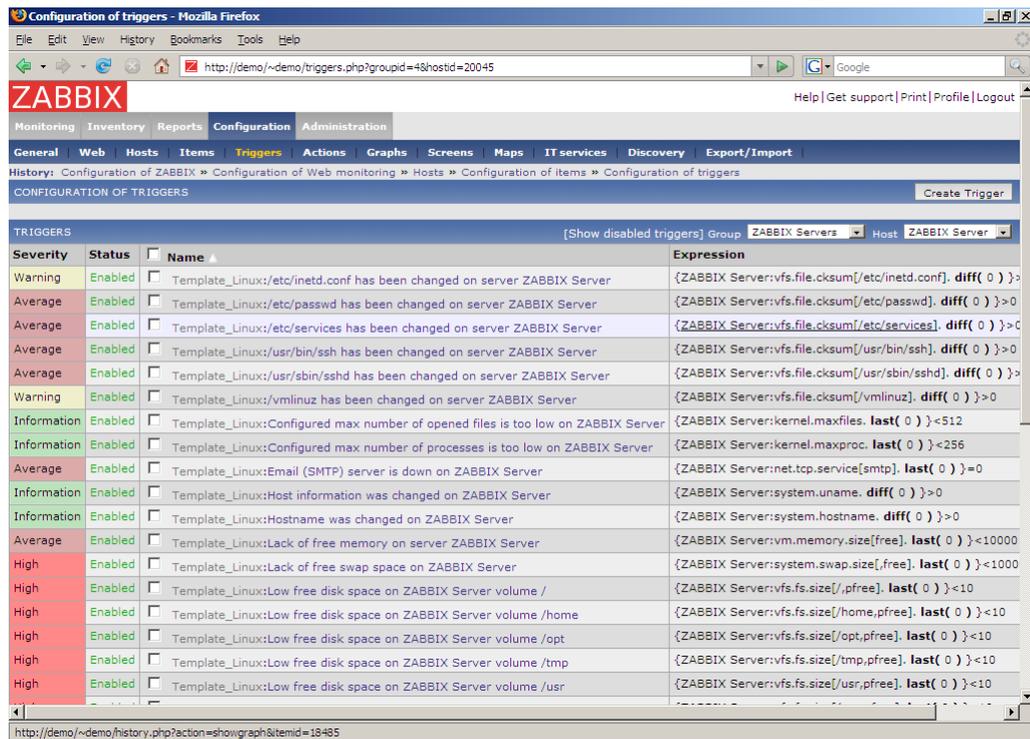
17.2.5. トリガー

17.2.5.1. トリガー

このスクリーンは、トリガーを管理するために使用します。

トリガーのリスト

このスクリーンには、ホストにリンクされているトリガーのリストが表示されます。

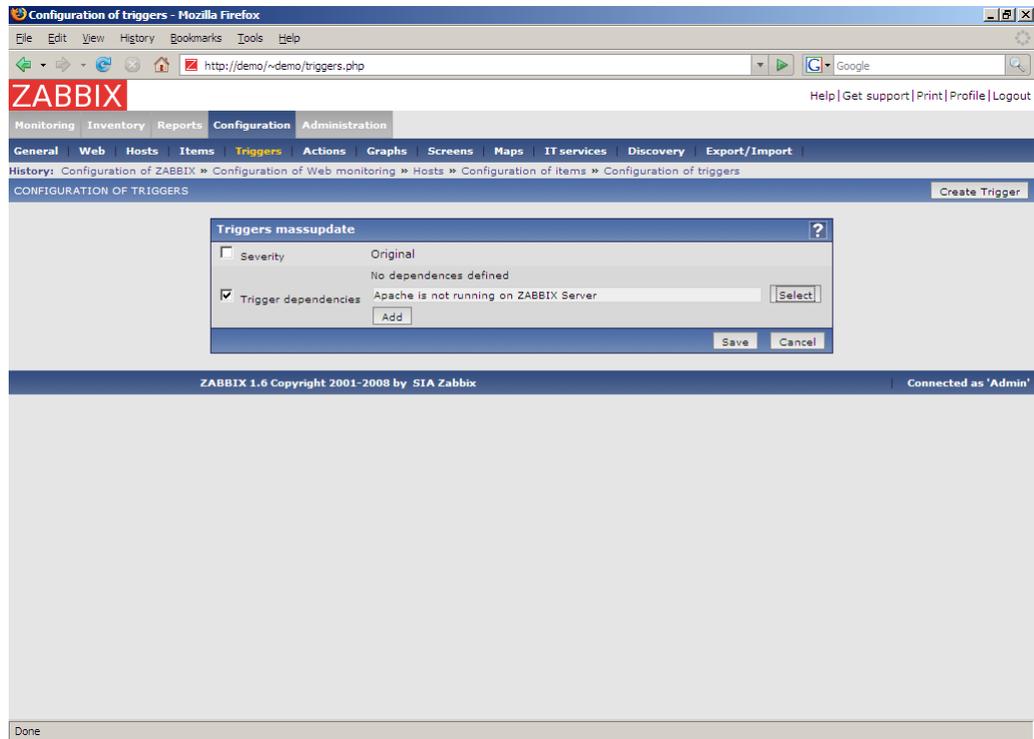


表示されるデータを以下に示します。

パラメータ	説明
[深刻度]	トリガーの深刻度(色分け)。
[ステータス]	トリガーのステータス。デフォルトでは、[無効]ステータスは表示されません。
[名前]	トリガーの名前。
[条件式]	トリガー条件式。

トリガーの一括更新画面

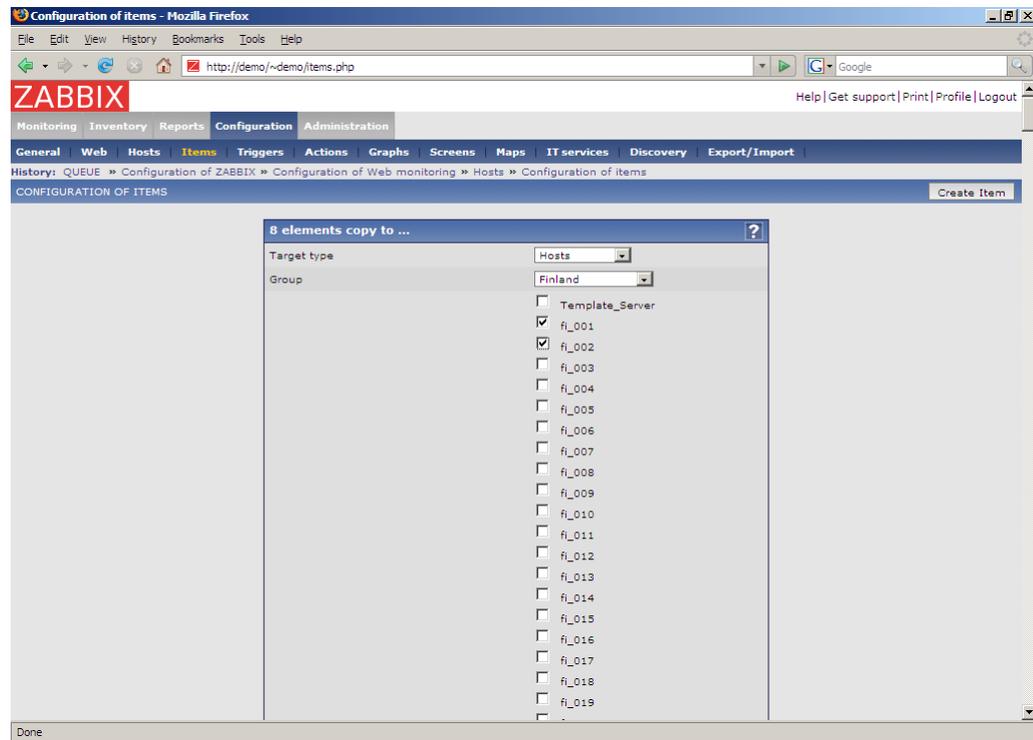
この画面にアクセスするには、トリガーを選択し、[一括更新]ボタンをクリックします。この画面では、複数のトリガーの属性を非常に効率的に変更できます。



変更するパラメータをクリックし、新しい値を入力して[保存]をクリックします。

選択をコピー

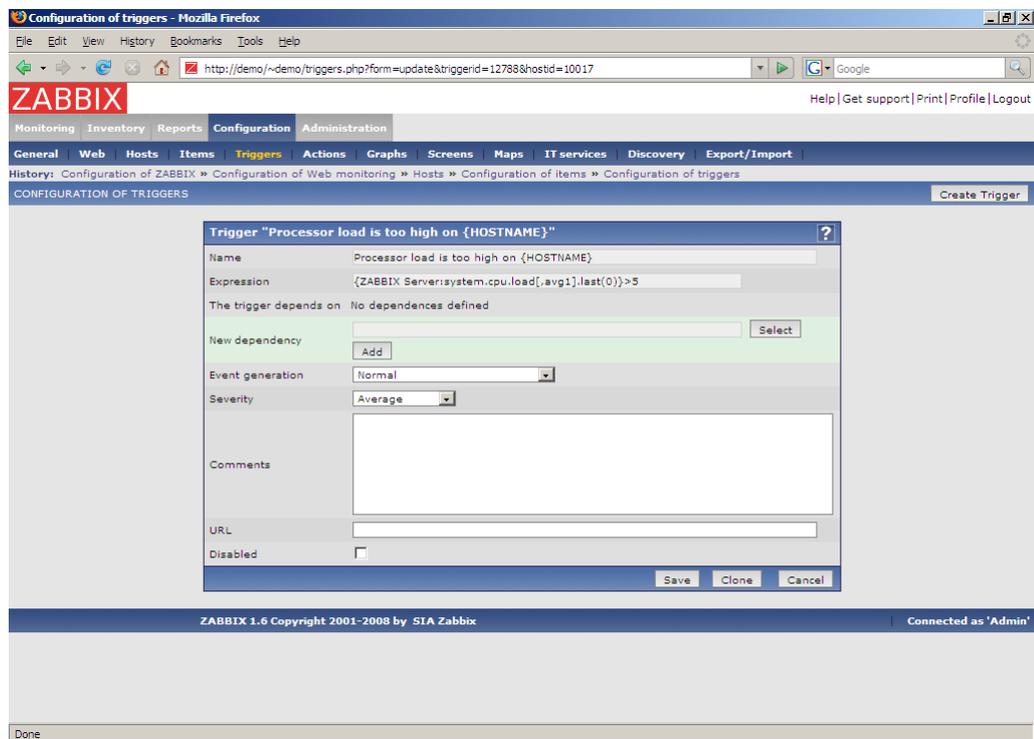
このスクリーンでは、選択したトリガーを複数のホストにコピーできます。



トリガーをコピーするホストを選択し、[コピー]をクリックします。

トリガーの設定

このスクリーンには、単一のトリガーの設定情報が表示されます。



トリガーの属性を以下に示します。

パラメータ	説明
[名前]	トリガーの名前。名前にはマクロも使用できます。
[条件式]	トリガー状態の計算に使用する論理条件式。
[次のトリガーに依存]	トリガーが依存するトリガーのリスト。
[依存関係の作成]	新しい依存関係を追加します。
[イベント生成]	[ノーマル] - トリガーのステータスが変更されると、イベントが標準で生成されます。 [ノーマル] + [障害イベントを継続して生成] - トリガーが障害と評価されるたびにイベントが生成されます。
[深刻度]	トリガーの深刻度。
[コメント]	このトリガーに関する詳細情報を指定するテキストフィールド。特定の問題に対する修正手順、担当者の連絡先などを入力します。
[URL]	入力された値は、[トリガー]スクリーンに表示されます。
[無効]	必要に応じてトリガーを無効にすることができます。

トリガーの詳細については、本書の他のセクションを参照してください。

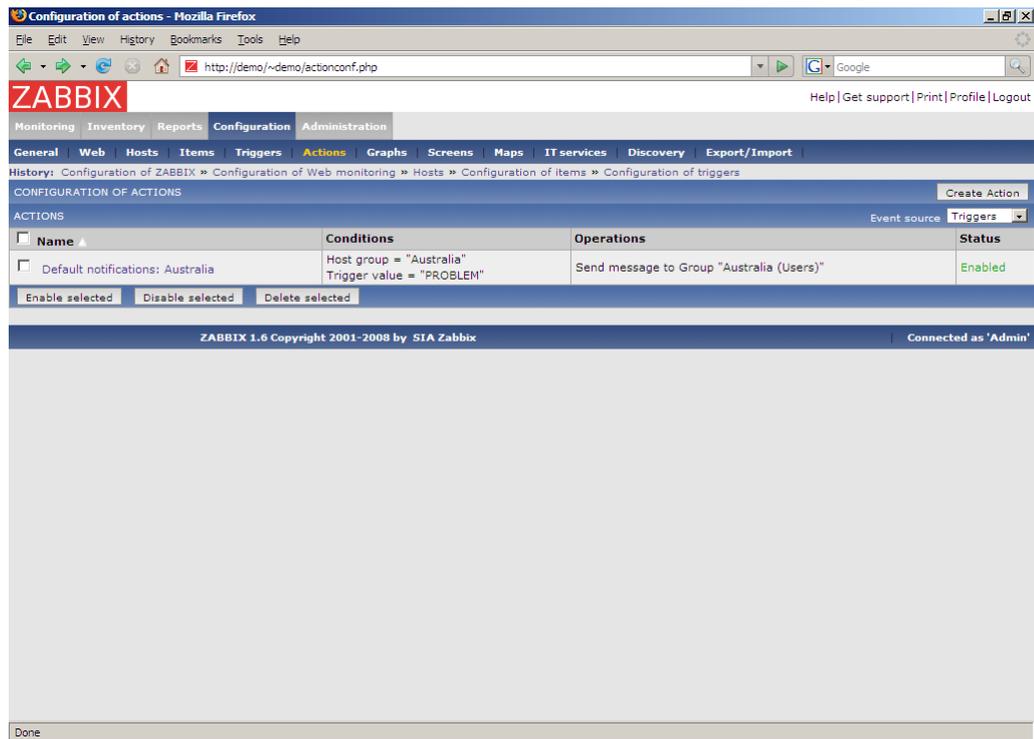
17.2.6. アクション

17.2.6.1. アクション

このスクリーンは、アクションを管理するために使用します。

アクションのリスト

このスクリーンには、アクションのリストが表示されます。

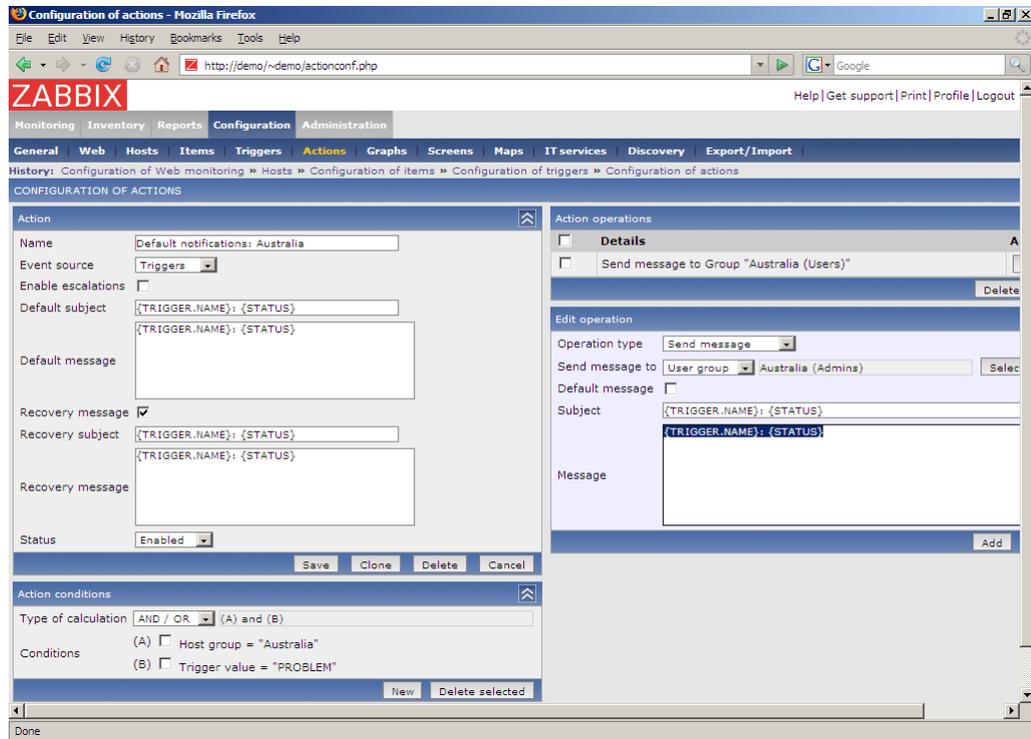


表示されるデータを以下に示します。

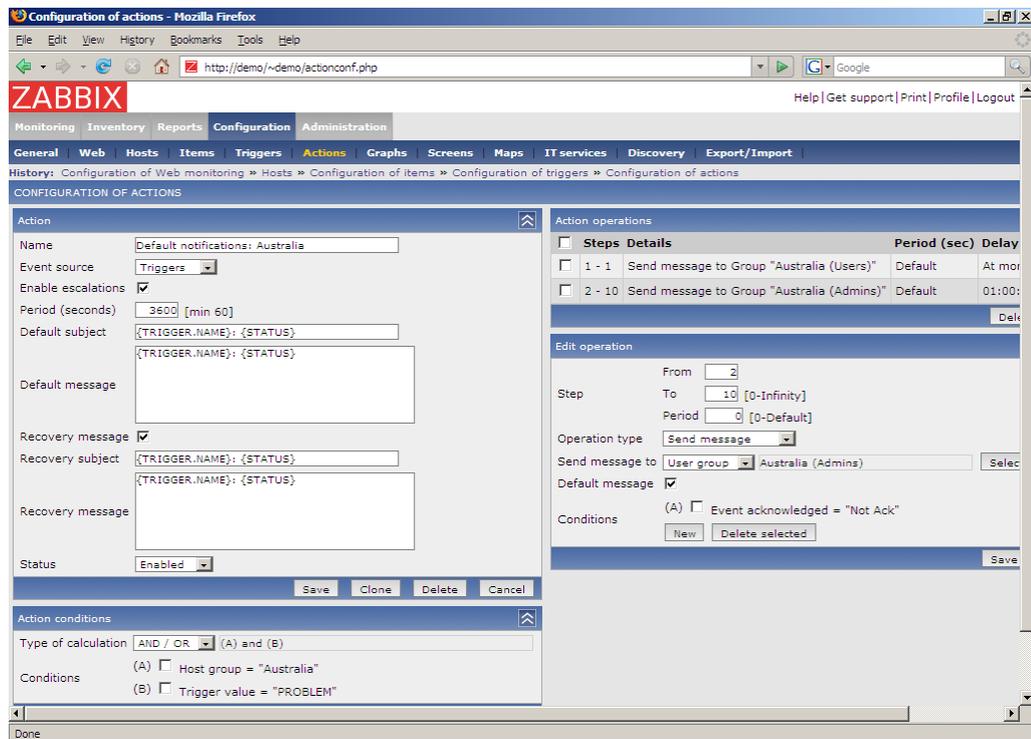
パラメータ	説明
[名前]	アクション名。
[コンディション]	対象のアクションに対するコンディションのリスト。
[オペレーション]	実行用のオペレーションのリスト。
[ステータス]	アクションのステータス。

アクションの設定

このスクリーンには、単一のアクションの設定情報が表示されます。



エスカレーションを有効にしている場合は、他の設定オプションも使用できます。



アクション、コンディション、およびオペレーションの設定の詳細については、本書の他のセクションを参照してください。

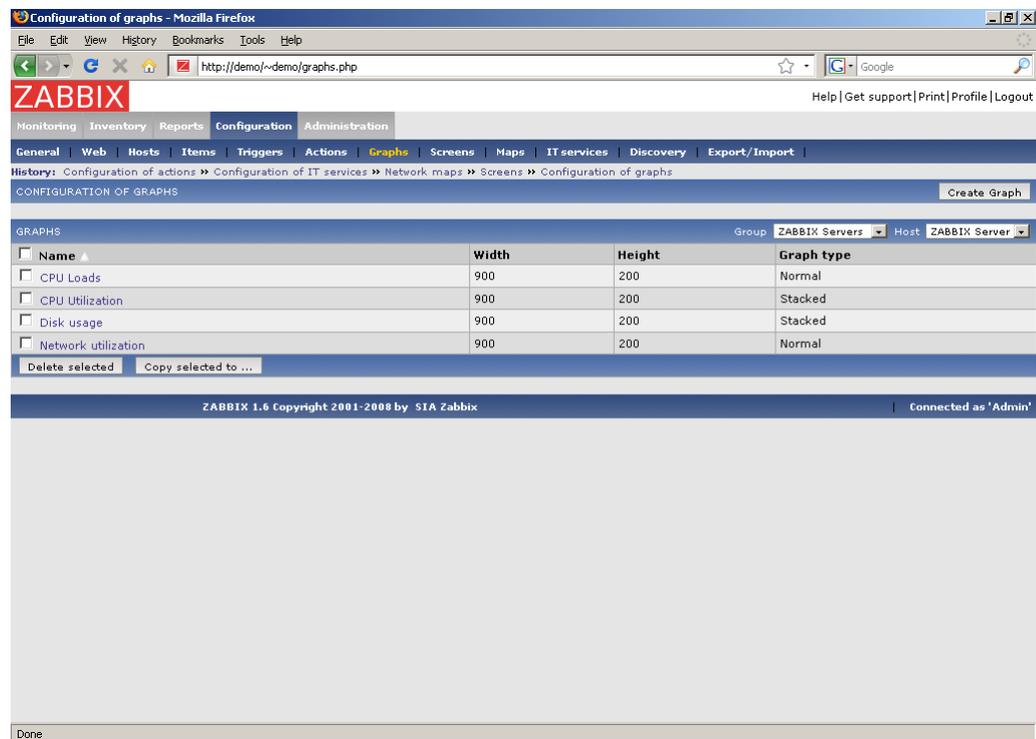
17.2.7. グラフ

17.2.7.1. グラフ

このスクリーンは、カスタムグラフを管理するために使用します。

グラフのリスト

このスクリーンには、グラフのリストが表示されます。

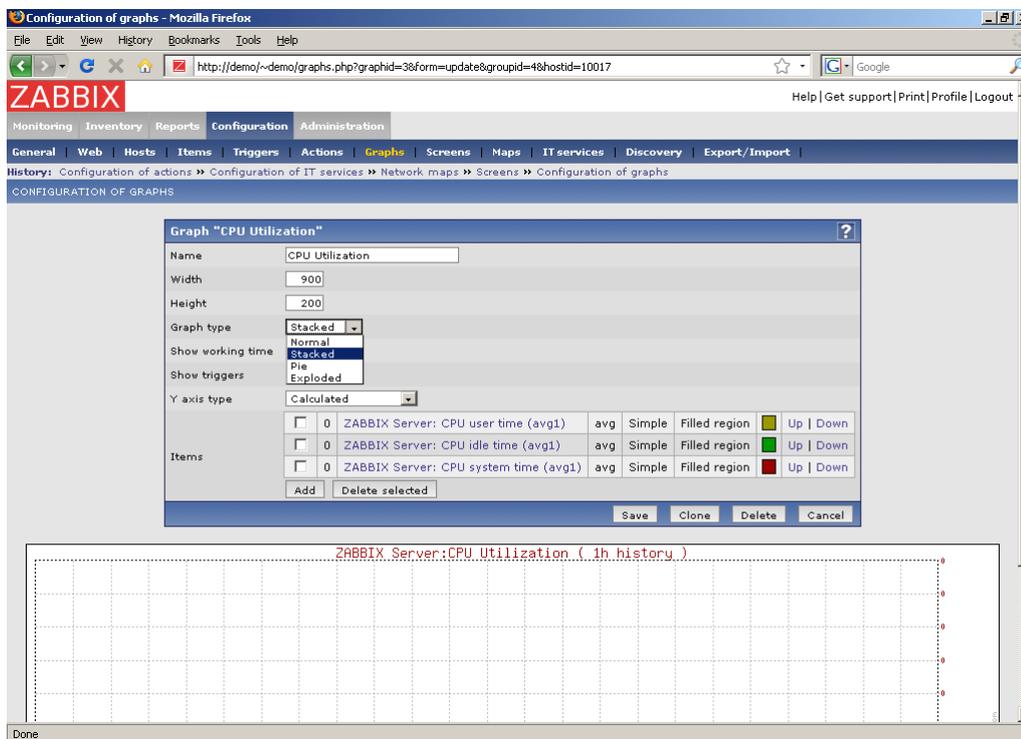


表示されるデータを以下に示します。

パラメータ	説明
[名前]	グラフ名。
[幅]	グラフの幅(ピクセル単位)。
[高さ]	グラフの高さ(ピクセル単位)。
[グラフのタイプ]	グラフのタイプ。 [ノーマル] [積算グラフ] [円グラフ] [分解円グラフ(Pie exploded)]

グラフの設定

このスクリーンには、単一のカスタムグラフの設定情報が表示されます。

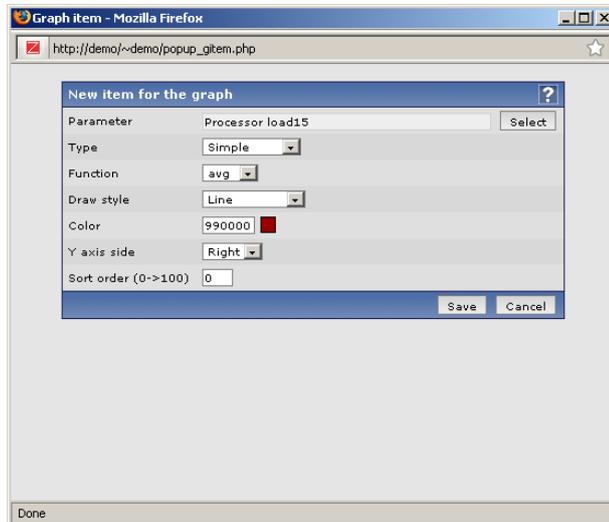


グラフの属性を以下に示します。

パラメータ	説明
[名前]	一意なグラフ名。
[幅]	グラフの幅(ピクセル単位)。
[高さ]	グラフの高さ(ピクセル単位)。
[グラフのタイプ]	グラフのタイプ。 [ノーマル] - ノーマルグラフ。値は線で表示されます。 [積算グラフ] - 積算グラフ。 [円グラフ] - 円グラフ。 [分解円グラフ] - 分解円グラフ。
[ワーキングタイムの表示]	選択すると、非ワーキングタイムは灰色背景で表示されます。
[トリガーを表示]	選択すると、単純なトリガーが赤線で表示されます。
[パーセントライン](左)	左Y軸にパーセンタイルが表示されます。通常、95%のパーセンタイルを表示するために使用されます。
[パーセントライン](右)	右Y軸にパーセンタイルが表示されます。通常、95%のパーセンタイルを表示するために使用されます。
[コメント]	このトリガーに関する詳細情報を指定するテキストフィールド。特定の問題に対する修正手順、担当者の連絡先などを入力します。
[Y軸のタイプ]	Y軸のタイプ。 [自動計算] - Y軸の値が自動的に計算されます。 [自動計算 [最小=0]] - Y軸の最小値が0に設定され、最大値は自動的に計算されます。 [固定] - Y軸の最小値と最大値に固定値が使用されます。

[3D表示]	3D形式を有効にします。円グラフ専用です。
[凡例]	凡例を表示します。円グラフ専用です。

グラフ項目



グラフ項目の属性を以下に示します。

パラメータ	説明
[パラメータ]	表示するホストアイテムを選択します。
[タイプ]	データ型: [標準] [累計]
[機能]	表示する値の種類。単一ピクセル(X軸)に複数の値が存在する場合に使用します。 [全て] - 全て(最小値、平均値、および最大値) [最小] - 最小値のみ [平均] - 平均値のみ [最大] - 最大値のみ
[グラフの形式]	グラフの形式。 [線] - 線を描画 [面] - 面を描画 [太線] - 太線を描画 [点線] - 点線を描画 [破線] - 破線を描画
[色]	RGB色(HEX表記)。
[累計データ数]	
[Y軸]	項目を割り当てる側のY軸。
[ソート順(0->100)]	描画順序。0が最初に処理されます。

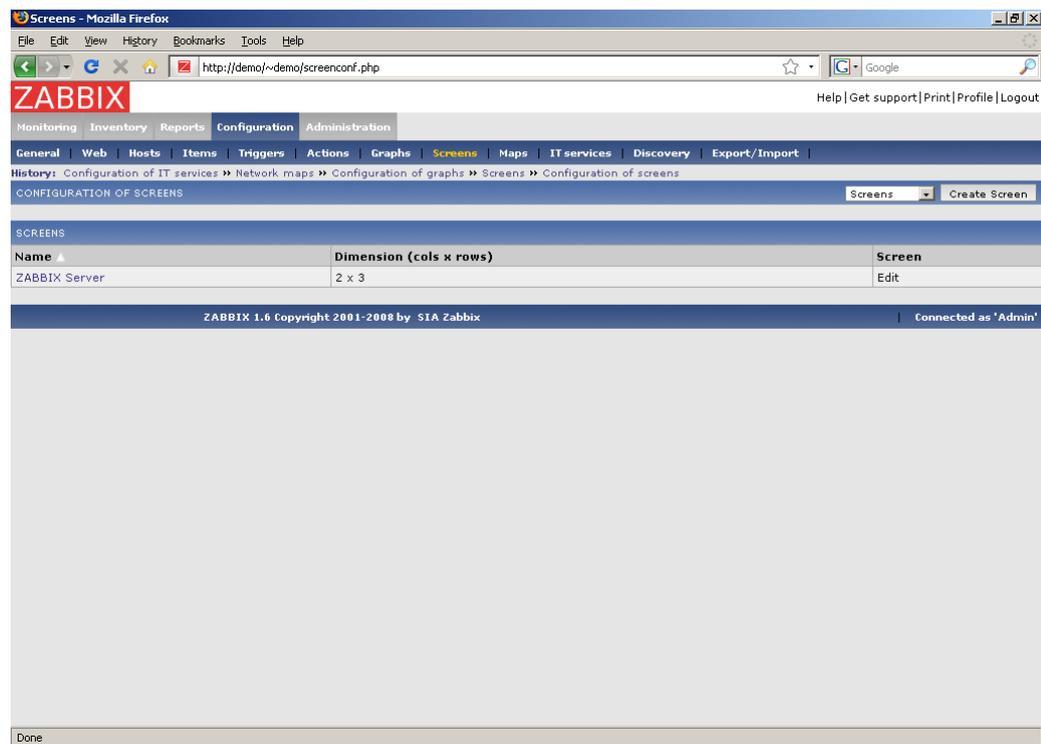
17.2.8. スクリーン

17.2.8.1. スクリーン

このスクリーンは、スクリーンを管理するために使用します。

スクリーンのリスト

このスクリーンには、スクリーンのリストが表示されます。

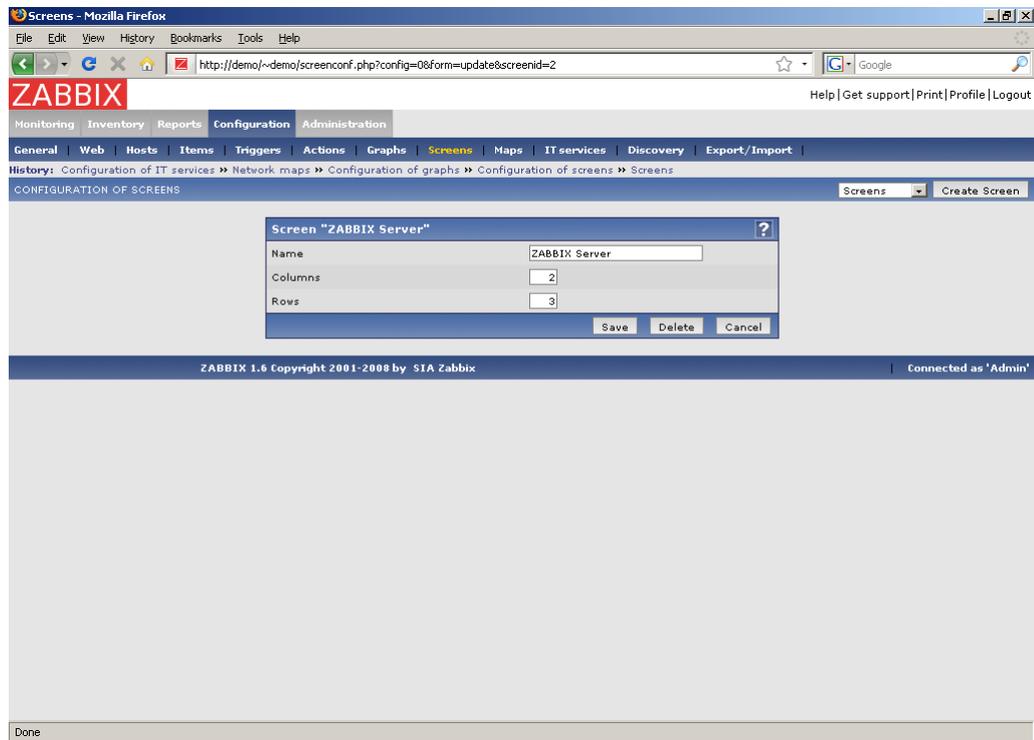


表示されるデータを以下に示します。

パラメータ	説明
[名前]	スクリーン名。
[大きさ(列x行)]	スクリーンサイズ(列数と行数)。

スクリーンの設定(高度)

このスクリーンには、単一のスクリーンの設定情報が表示されます。

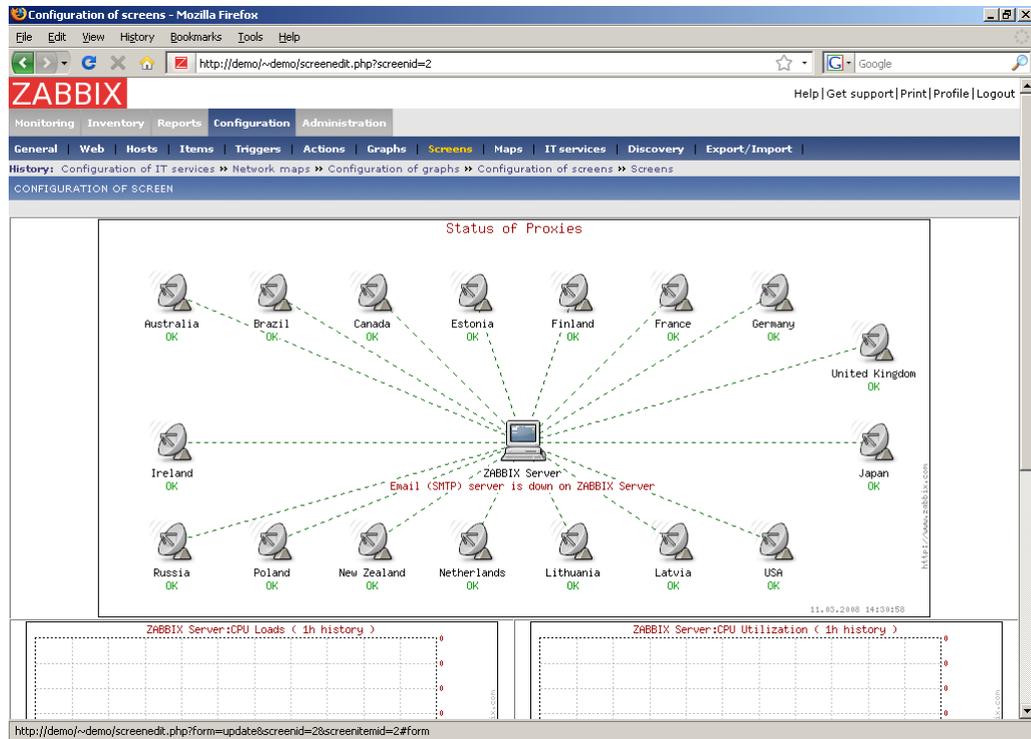


スクリーンの高度な属性を以下に示します。

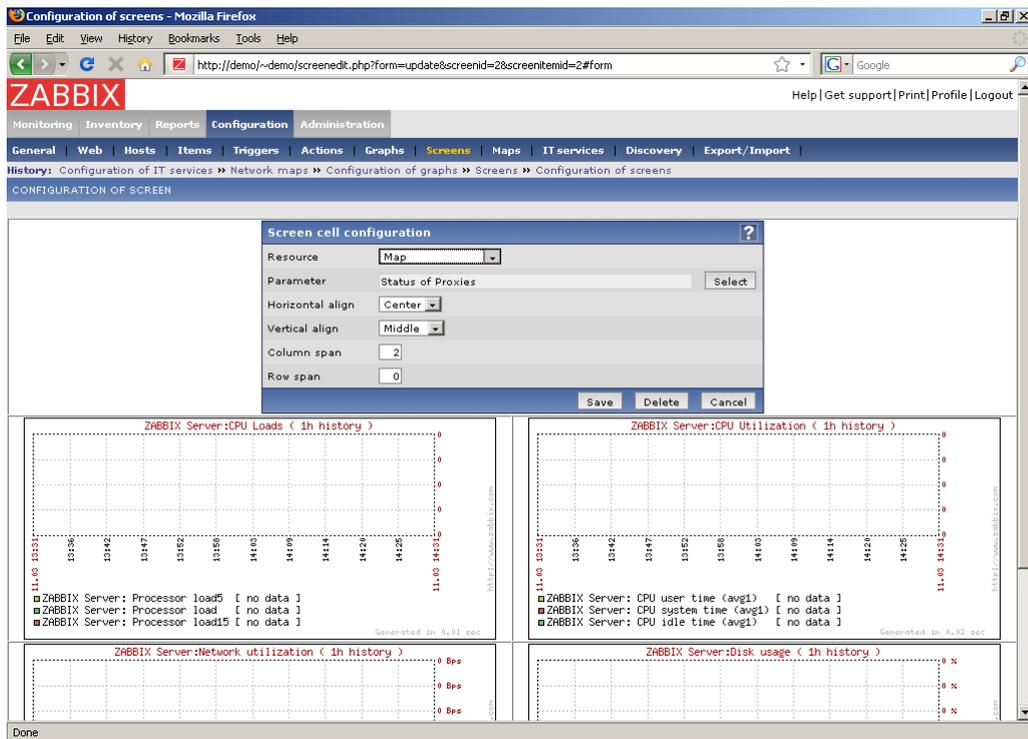
パラメータ	説明
[名前]	一意なスクリーン名。
[列]	スクリーン内の列数。
[行]	スクリーン内の行数。

スクリーンの設定(スクリーン項目)

このスクリーンには、全ての項目の設定にアクセスできる単一のスクリーンの設定情報が表示されます。



スクリーン項目(セル)をクリックして、スクリーンのセルに表示する情報を変更します。



スクリーンの高度な属性を以下に示します。

パラメータ	説明
[リソース]	セルに表示される情報を指定します。 [時刻] - 現在のサーバ時刻またはローカル時刻を表示するデジタル時計またはアナログ時計 [データの概要] - ホストのグループに関する最新データ [グラフ] - 単一のカスタムグラフ [アクションの履歴] - 最新のアクションの履歴 [イベントの履歴] - 最新のイベント [ホスト情報] - ホスト関連の高度な情報 [マップ] - 単一のマップ [プレーンテキスト] - プレーンテキストデータ [スクリーン] - スクリーン(1つのスクリーン内に他のスクリーンを含めることができます) [サーバ情報] - サーバ関連の高度な情報 [シンプルグラフ] - 単一の簡易グラフ [トリガー情報] - トリガー関連の高度な情報 [トリガーの概要] - ホストグループのトリガーのステータス [URL] - 外部リソースからのコンテンツを含むURL
[横位置]	指定できる値は以下のとおりです。 [中央] [左] [右]
[縦位置]	指定できる値は以下のとおりです。 [中央] [上] [下]

[列の結合]	HTMLの列を拡張する場合と同じように、セルを複数の列に拡張します。
[行の結合]	HTMLの行を拡張する場合と同じように、セルを複数の行に拡張します。

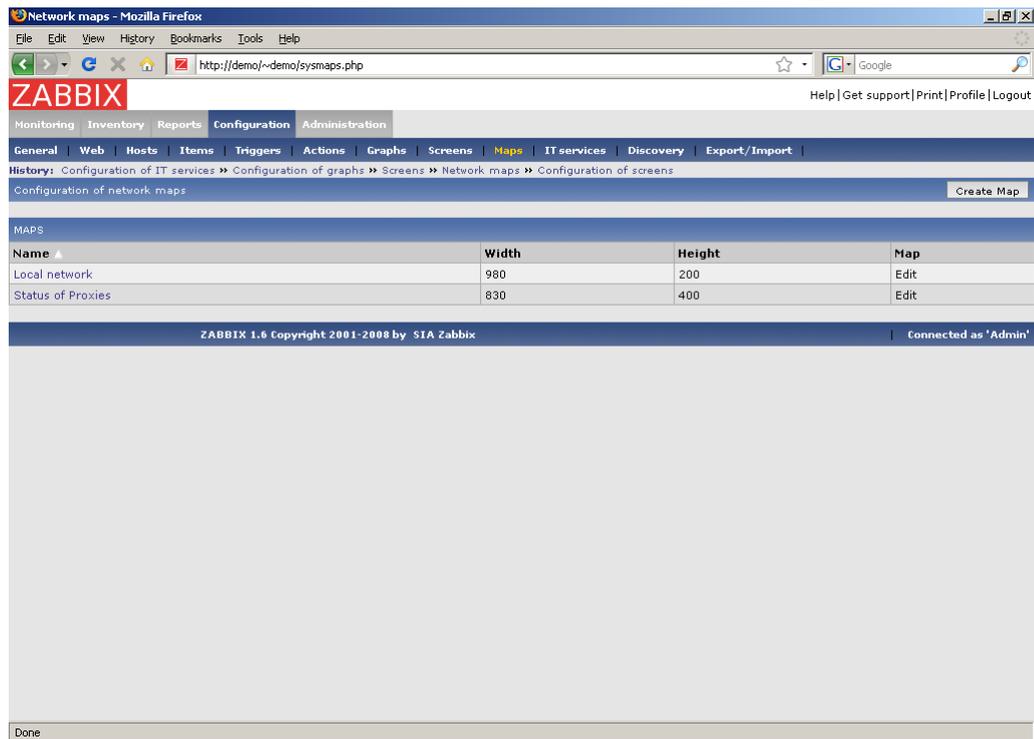
17.2.9. マップ

17.2.9.1. マップ

このスクリーンは、ユーザ定義のマップを管理するために使用します。

マップのリスト

このスクリーンには、マップのリストが表示されます。

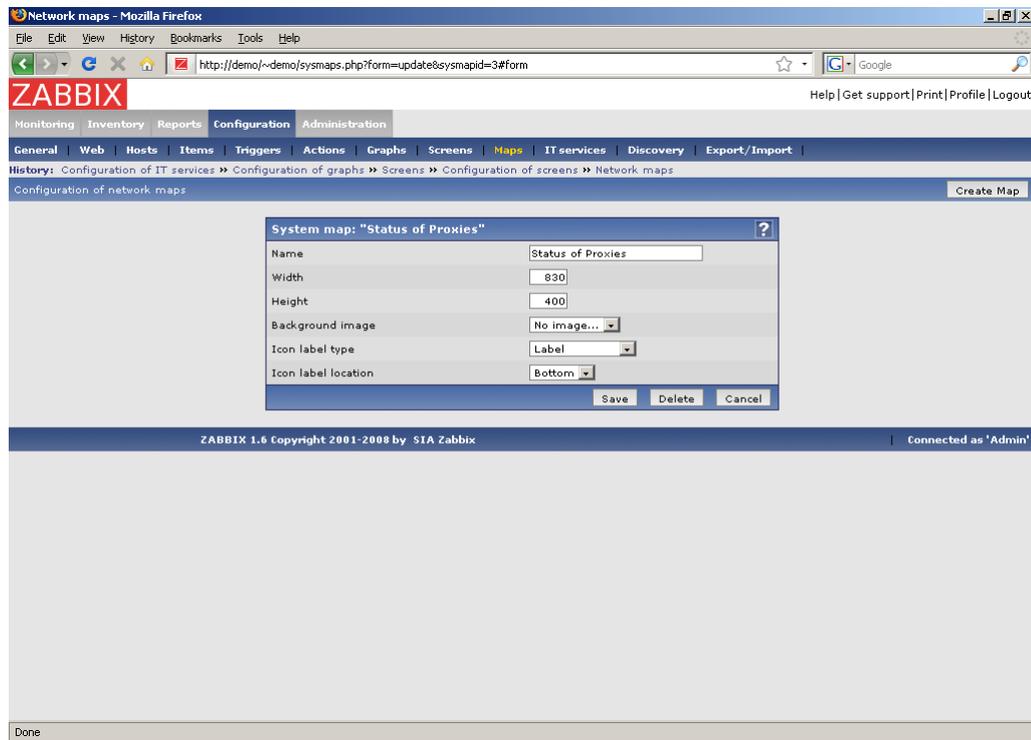


表示されるデータを以下に示します。

パラメータ	説明
[名前]	マップ名。
[幅]	マップの幅(ピクセル単位)。
[高さ]	マップの高さ(ピクセル単位)。

マップの設定(高度)

このスクリーンには、ユーザ定義のスクリーンの設定情報が表示されます。



マップの高度な属性を以下に示します。

パラメータ	説明
[名前]	一意なマップ名。
[幅]	マップの幅(ピクセル単位)。
[高さ]	マップの高さ(ピクセル単位)。
[背景のイメージ]	使用する背景イメージ。 [イメージなし(No image)] - 背景イメージは使用されません(白背景)。 [イメージ] - 選択したイメージが背景イメージとして使用されます。拡大縮小は行われません。
[アイコンラベルのタイプ]	全てのマップアイコンに使用されるラベルタイプ。 [ラベル] - アイコンラベルのみ [IPアドレス] - IPアドレスのみ [要素名] - 要素名(ホスト名など) [ステータスのみ] - ステータスのみ([正常]または[障害]) [無] - アイコンラベルは表示されない
[アイコンラベルの位置]	アイコンラベルの表示位置。 [下] - 下(アイコンの下部) [左] - 左側 [右] - 右側 [上] - アイコンの上部

マップの設定(マップ項目の設定)

このスクリーンには、マップアイコンとリンクの設定情報が表示されます。

マップ項目(アイコン)のリスト

Configuration of network maps - Mozilla Firefox

http://demo/~demo/sysmap.php?sysmapid=3

ZABBIX Help | Get support | Print | Profile | Logout

Monitoring Inventory Reports Configuration Administration

General Web Hosts Items Triggers Actions Graphs Screens Maps IT services Discovery Export/Import

History: Configuration of graphs » Screens » Configuration of screens » Configuration of network maps » Network maps

CONFIGURATION OF NETWORK MAPS

DISPLAYED ELEMENTS Add element

Label	Type	X	Y	Icon (ok)	Icon (problem)	Icon (unknown)	Icon (disabled)
Australia	Host	50	50				
Brazil	Host	150	50				
Canada	Host	250	50				
Estonia	Host	350	50				
Finland	Host	450	50				
France	Host	550	50				
Germany	Host	650	50				
Ireland	Host	50	200				
Japan	Host	750	200				
Latvia	Host	550	300				
Lithuania	Host	450	300				
Netherlands	Host	350	300				
New Zealand	Host	250	300				
Poland	Host	150	300				
Russia	Host	50	300				
United Kingdom	Host	750	100				
USA	Host	650	300				

Done

リンクのリスト

Configuration of network maps - Mozilla Firefox

http://demo/~demo/sysmap.php?sysmapid=3

ZABBIX Server Host 400 200

CONNECTORS Create connection

Link	Element 1	Element 2	Link status indicator
link 1	ZABBIX Server	Australia	-
link 2	ZABBIX Server	Brazil	-
link 3	ZABBIX Server	Canada	-
link 4	ZABBIX Server	Estonia	-
link 5	ZABBIX Server	Finland	-
link 6	ZABBIX Server	France	-
link 7	ZABBIX Server	Germany	-
link 8	ZABBIX Server	Ireland	-
link 9	ZABBIX Server	Japan	-
link 10	ZABBIX Server	Latvia	-
link 11	ZABBIX Server	Lithuania	-
link 12	ZABBIX Server	Netherlands	-
link 13	ZABBIX Server	New Zealand	-
link 14	ZABBIX Server	Poland	-
link 15	ZABBIX Server	Russia	-
link 16	ZABBIX Server	United Kingdom	-
link 17	ZABBIX Server	USA	-

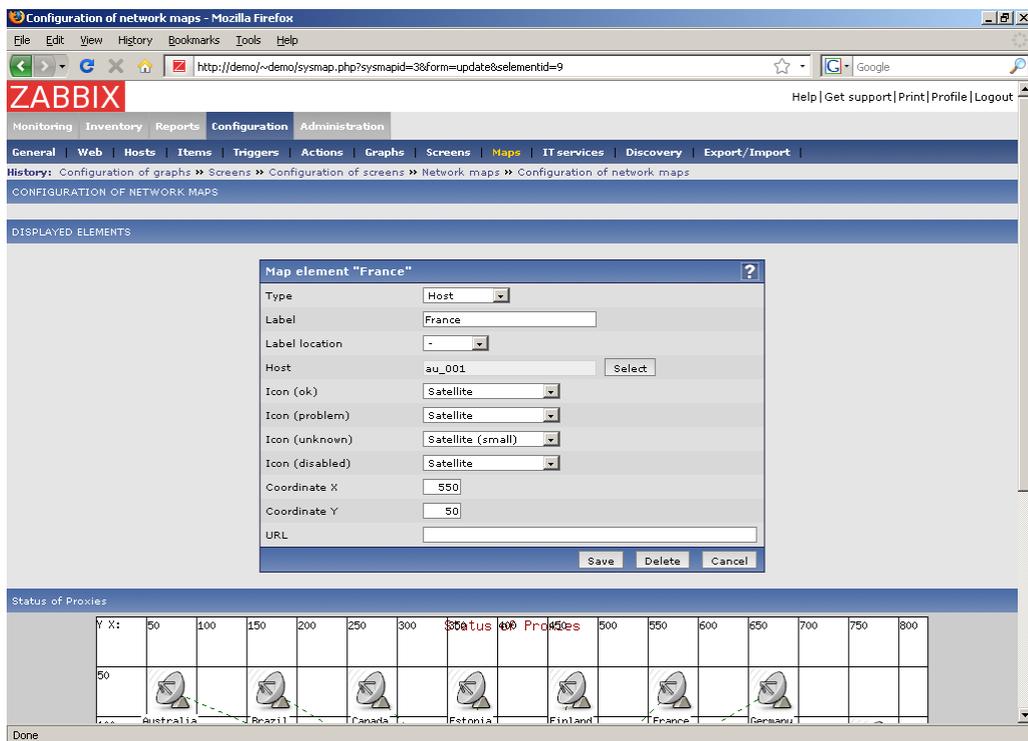
Status of Proxies

Y	X	50	100	150	200	250	300	350	400	450	500	550	600	650	700	750	800
50																	
100		Australia OK	Brazil OK	Canada OK	Estonia OK	Finland OK	France OK	Germany OK	United Kingdom								

Done

マップ項目の設定

このスクリーンには、単一のマップ項目の設定情報が表示されます。



マップ項目の属性を以下に示します。

パラメータ	説明
[タイプ]	項目のタイプ。 [ホスト] - 選択したホストの全てのトリガーのステータスを表すアイコン [マップ] - マップの全ての項目のステータスを表すアイコン [トリガー] - 単一のトリガーのステータスを表すアイコン [ホストグループ] - 対象のホストグループに属する全てのホストの全てのトリガーのステータスを表すアイコン [イメージ] - リソースにリンクされていない単なるアイコン
[ラベル]	アイコンラベル(任意の文字列)。 バージョン1.8以降では、ラベルにマクロと複数行の文字列を使用できます。
[ラベルの位置]	ラベルの位置。 デフォルト - マップのデフォルトのラベルの位置 [下] - 下(アイコンの下部) [左] - 左側 [右] - 右側 [上] - アイコンの上部
[ホスト]	対象のホストのトリガーのステータスが使用されます。
[マップ]	対象のマップの全ての項目のステータスが使用されます。
[トリガー]	対象のトリガーのステータスが使用されます。
[ホストグループ]	対象のホストグループの全てのトリガーのステータスが使用されます。

[アイコン (正常)]	問題が存在しない場合に使用されるアイコン。
[アイコン(障害)]	(1つまたは複数の)問題が存在する場合に使用されるアイコン。
[アイコン(不明)]	(1つまたは複数の)問題が存在する場合に使用されるアイコン。
[アイコン(無効)]	選択したホストが無効な場合に使用されるアイコン。
[X座標]	マップ項目のX座標。
[Y座標]	マップ項目のY座標。
[URL]	設定すると、ユーザがスクリーン項目をクリックしたときにそのURLが使用されます。

リンクの設定

このスクリーンには、リンクの設定情報が表示されます。

マップリンクの属性を以下に示します。

パラメータ	説明
[アイコン1]	一意なスクリーン名。
[アイコン2]	スクリーン内の列数。
[状態識別用トリガー]	リンクにリンクされているトリガーのリスト。トリガーのステータスが[障害]の場合、その形式がリンクに適用されます。
[線種(正常)(Type (OK))]	デフォルトのリンク形式。 [線] - 単線 [太線] - 太線 [点線] - 点線 [破線] - 破線
[色(正常)(Color (OK))]	デフォルトのリンク色。

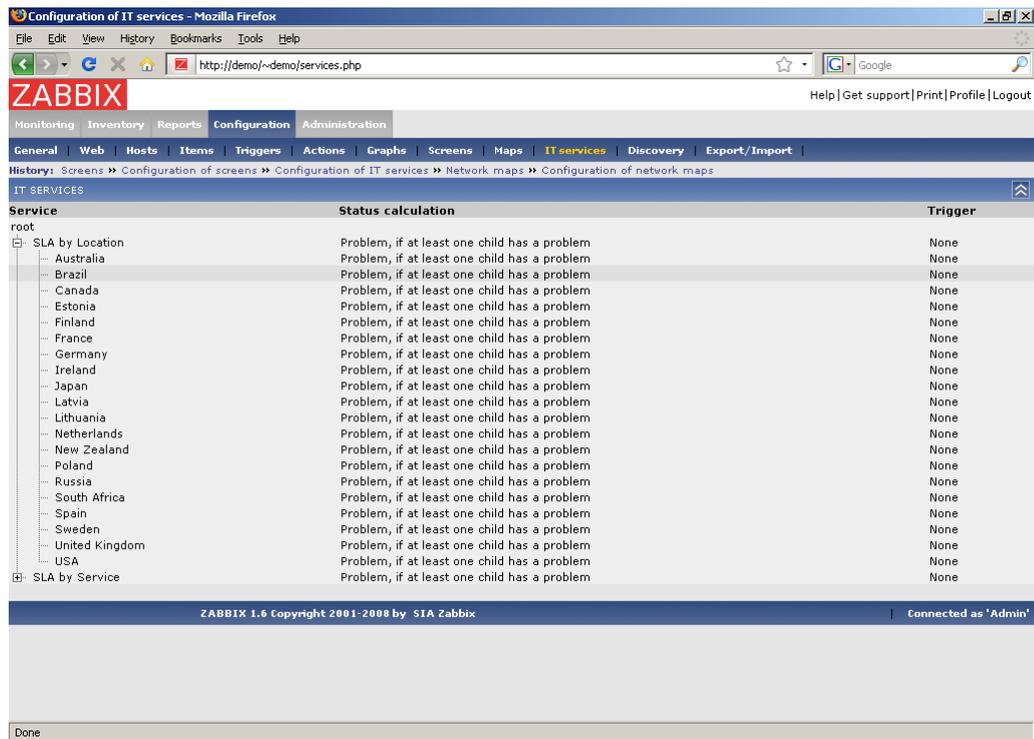
17.2.10. IT サービス

17.2.10.1. IT サービス

このスクリーンは、IT サービスを管理するために使用します。

IT サービスのリスト

このスクリーンには、IT サービスのリストが表示されます。



表示されるデータを以下に示します。

パラメータ	説明
[サービス]	サービス名。
[ステータスの計算]	サービスによるそのステータスの更新方法。
[トリガー]	トリガーへのリンク。 [なし] - リンクされていません。 [トリガー名] - トリガーにリンクされています(トリガーのステータスに依存)。

IT サービスの設定

このスクリーンには、ユーザ定義のスクリーンの設定情報が表示されます。

The screenshot shows a web browser window titled "IT services - Mozilla Firefox" displaying the configuration form for a service named "Australia". The form includes the following fields and options:

- Name:** Australia
- Parent Service:** SLA by Location (with a "Change" button)
- Depends on:** A table with columns for "Services", "Soft", and "Trigger", and buttons for "Add" and "Remove".
- Status calculation algorithm:** Problem, if at least one child has a problem
- Calculate SLA:**
- Acceptable SLA (in %):** 99.05
- Service times:** No times defined
- New service time:** Uptime (dropdown), From: Sunday (dropdown), H:i (input), Till: Sunday (dropdown), H:i (input), and an "add" button.
- Link to trigger?:**
- Trigger:** Mysql is not running on ZABBIX Server (with a "Select" button)
- Sort order (0->999):** 1

At the bottom right of the form are "Save", "Delete", and "Cancel" buttons. The status bar at the bottom of the browser window shows "Done".

IT サービスの属性を以下に示します。

パラメータ	説明
[名前]	サービス名。
[親サービス(Parent service)]	親サービス。参照専用であり、変更できません。
[依存先]	サービスが依存する子サービスのリスト。
[ステータス計算アルゴリズム]	サービスのステータスの計算方法。 [計算しない] - サービスのステータスは計算されません。 [子に一つでも障害が合った場合に障害として検知] - 少なくとも1つの子サービスに障害があった場合に障害とみなされます。 [すべての子に障害があった場合に障害として検知] - 全ての子に障害があった場合に障害とみなされます。
[算出SLA]	SLAデータを表示する場合に選択します。
[SLAの許容値(%)]	対象のサービスのSLAパーセント値。レポート用に使われます。
[サービスタイム]	デフォルトでは、全てのサービスが24×7×365体制で稼働します。例外を設定するには、新しいサービスタイムを追加します。
[新規サービスタイム]	サービスタイム。 [一時的なダウンタイム] - 1回のダウンタイム。この期間中のサービスの状態はSLAに影響しません。

	[アップタイム] - サービスのアップタイム。 [ダウンタイム] - この期間中のサービスの状態はSLAに影響しません。
[トリガーとリンク]	最下位レベルのサービスがトリガーにリンクされている必要があります。
[ソート順]	表示のソート順。最下位レベルが最初に表示されます。

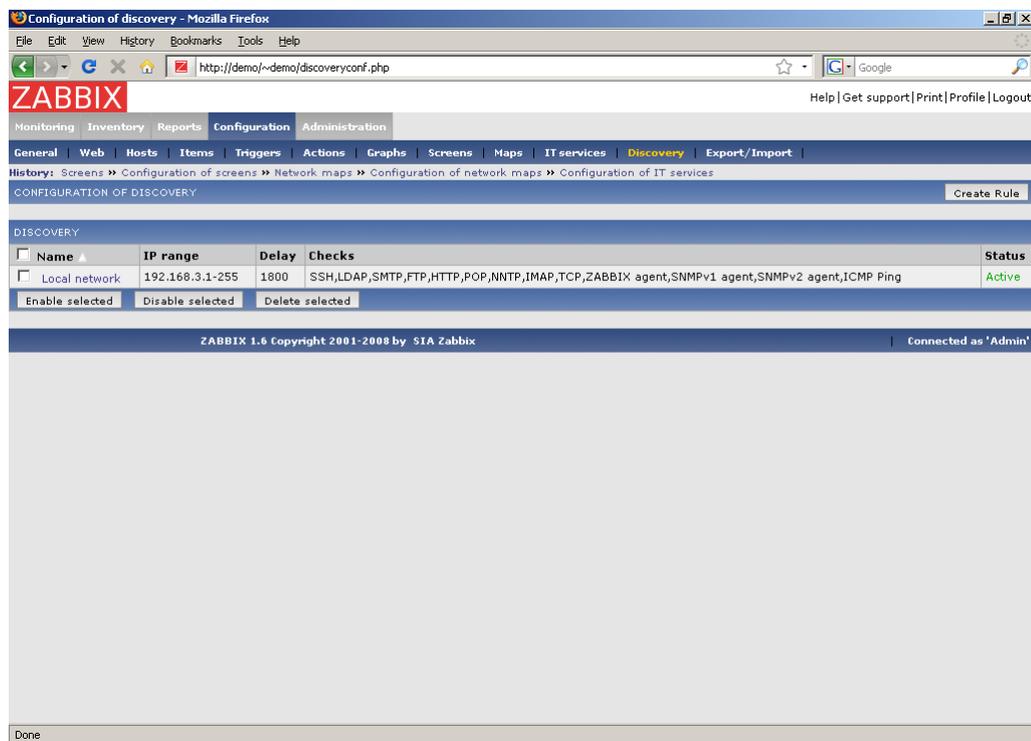
17.2.11. ディスカバリ

17.2.11.1. ディスカバリ

このスクリーンは、ディスカバリルールを管理するために使用します。

ディスカバリルールのリスト

このスクリーンには、ディスカバリルールのリストが表示されます。

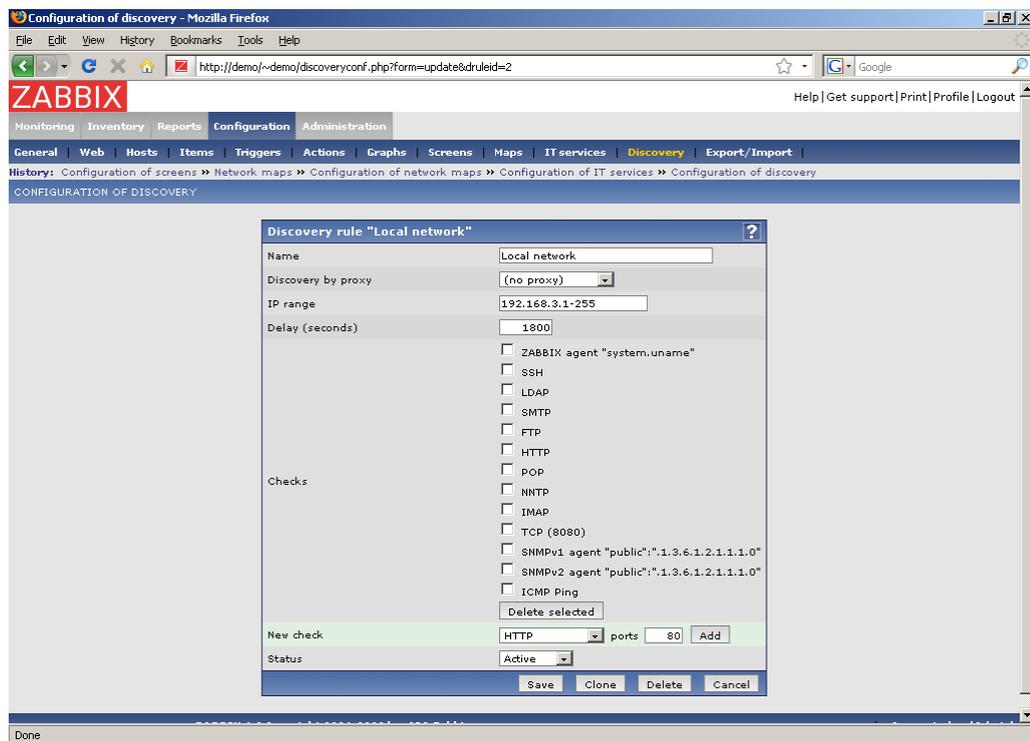


表示されるデータを以下に示します。

パラメータ	説明
[名前]	ディスカバリルールの名前。
[IPアドレスの範囲]	ディスカバリルールの影響を受けるIPアドレスの範囲。
[間隔]	頻度(秒単位)。
[チェック]	ディスカバリルールによって実行されるチェックのリスト。
[ステータス]	ディスカバリルールのステータス。 [有効] - ルールは有効です。 [無効] - ルールは無効です。

ディスカバリルールの設定

このスクリーンには、ディスカバリルールの設定情報が表示されます。



ディスカバリルールの属性を以下に示します。

パラメータ	説明
[名前]	ディスカバリルールの一意な名前。
[プロキシによるディスカバリ]	ディスカバリの実行担当。 [(プロキシなし)] - ZABBIXサーバによってディスカバリが実行されます。 [プロキシ名] - ここで指定したプロキシによってディスカバリが実行されます。
[IPアドレスの範囲]	チェックするIPアドレスの範囲。以下の形式を使用します。 単一IPアドレス: 192.168.1.33 IPアドレスの範囲: 192.168.1.1-255 リスト: 192.168.1.1-255,192.168.2.1-100,192.168.2.200
[間隔(秒)]	対象のルールの実行頻度(秒単位)を定義します。
[チェック]	サポートされているチェックのリスト: SSH、LDAP、SMTP、FTP、HTTP、POP、NNTP、IMAP、TCP、ZABBIXエージェント、SNMPv1エージェント、SNMPv2エージェント
[新規チェック]	対象のサービスのSLAパーセント値。レポート用に使用されません。
[ポート]	このパラメータは以下のいずれかの形式で指定できます。 単一ポート: 22 ポートの範囲: 22-45 リスト: 22-45,55,60-70
[ステータス]	ディスカバリルールのステータス。 [有効] - ルールは有効です。

	[無効] - ルールは無効です。
[新規サービスタイム]	サービスタイム。 [一時的なダウンタイム] - 1回のダウンタイム。この期間中のサービスの状態はSLAに影響しません。 [アップタイム] - サービスのアップタイム。 [ダウンタイム] - この期間中のサービスの状態はSLAに影響しません。
[トリガーとリンク]	最下位レベルのサービスがトリガーにリンクされている必要があります。
[ソート順]	表示のソート順。最下位レベルが最初に表示されます。

17.2.12. エクスポート/インポート

17.2.12.1. エクスポート

このスクリーンは、ホスト、アイテム、トリガー、およびグラフをエクスポートするために使用します。

エクスポート

このスクリーンには、エクスポート対象のホストとその項目のリストが表示されます。

The screenshot shows the Zabbix web interface for the 'Export/Import' function. The main content area displays a table of templates with the following columns: Name, DNS, IP, Port, Status, Templates, Items, Triggers, and Graphs. The 'Name' column is sorted in ascending order. The table lists various templates such as 'Template_APC_Battery', 'Template_App_MySQL', 'Template_Cisco_P1X_S25', etc. Each row has checkboxes for 'Templates', 'Items', 'Triggers', and 'Graphs' to select items for export. At the bottom of the table, there are 'Preview' and 'Export' buttons.

Name	DNS	IP	Port	Status	Templates	Items	Triggers	Graphs
<input type="checkbox"/> Template_APC_Battery	-	-	-	Template	-	<input checked="" type="checkbox"/> 7	<input checked="" type="checkbox"/> 6	-
<input type="checkbox"/> Template_App_MySQL	-	-	-	Template	-	<input checked="" type="checkbox"/> 6	-	-
<input type="checkbox"/> Template_Cisco_P1X_S25	-	-	-	Template	-	<input checked="" type="checkbox"/> 35	-	-
<input type="checkbox"/> Template_Dell_OpenManage	-	-	-	Template	-	<input checked="" type="checkbox"/> 15	<input checked="" type="checkbox"/> 15	-
<input type="checkbox"/> Template_Dell_PowerConnect_6248	-	-	-	Template	-	<input checked="" type="checkbox"/> 832	-	<input checked="" type="checkbox"/> 52
<input type="checkbox"/> Template_Dell_PowerEdge	-	-	-	Template	-	<input checked="" type="checkbox"/> 2	<input checked="" type="checkbox"/> 2	<input checked="" type="checkbox"/> 1
<input type="checkbox"/> Template_HP_InsightManager	-	-	-	Template	-	<input checked="" type="checkbox"/> 12	<input checked="" type="checkbox"/> 10	-
<input type="checkbox"/> Template_Linux	-	-	-	Template	-	<input checked="" type="checkbox"/> 102	<input checked="" type="checkbox"/> 44	-
<input type="checkbox"/> Template_Server	-	-	-	Template	-	<input checked="" type="checkbox"/> 10	<input checked="" type="checkbox"/> 1	<input checked="" type="checkbox"/> 3
<input type="checkbox"/> Template_SNMPV2_Device	-	-	-	Template	-	<input checked="" type="checkbox"/> 207	<input checked="" type="checkbox"/> 207	-
<input type="checkbox"/> Template_Standalone	-	-	-	Template	-	<input checked="" type="checkbox"/> 8	<input checked="" type="checkbox"/> 7	-
<input type="checkbox"/> Template_Windows	-	-	-	Template	-	<input checked="" type="checkbox"/> 29	<input checked="" type="checkbox"/> 13	-

エクスポートする項目を選択し、[プレビュー]または[エクスポート]をクリックします。

表示されるデータを以下に示します。

パラメータ	説明
[名前]	ホスト名。
[DNS]	ホストDNS名。
[IPアドレス]	ZABBIXエージェントのIPアドレス。
[ポート]	ZABBIXエージェントのポート番号。
[ステータス]	ホストのステータス。
[テンプレート]	テンプレート関連の情報をエクスポートする場合に選択し ます。
[アイテム]	ホストのアイテムをエクスポートする場合に選択し ます。
[トリガー]	ホストのトリガーをエクスポートする場合に選択し ます。
[グラフ]	ホストのグラフをエクスポートする場合に選択し ます。

プレビューページ

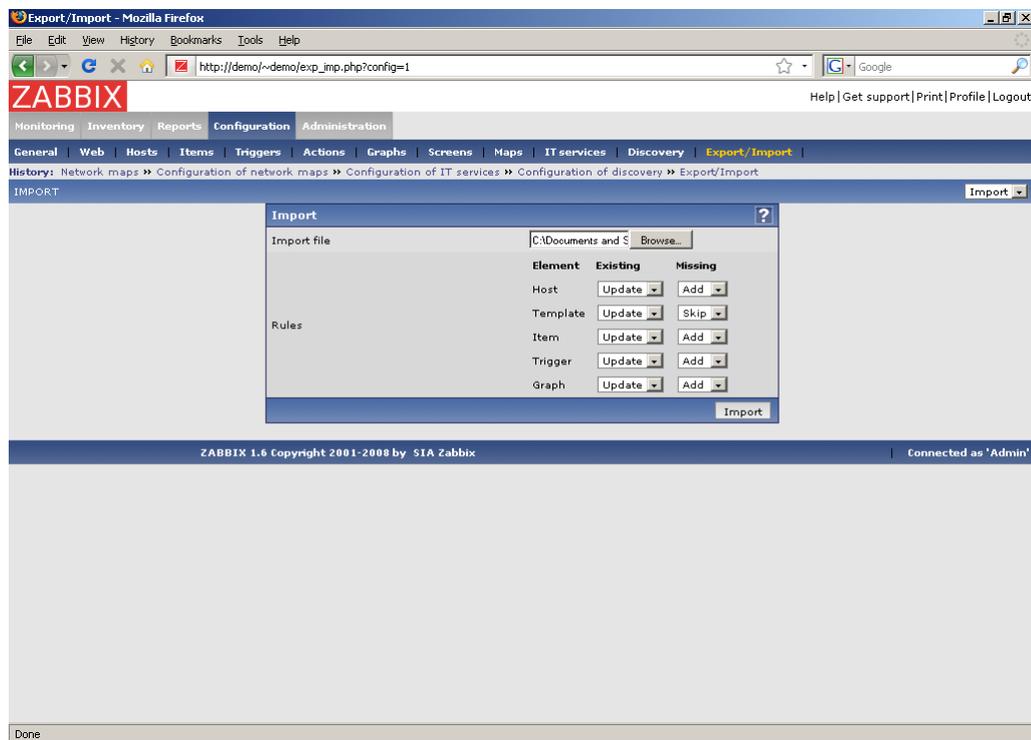
The screenshot shows the Zabbix web interface in a Mozilla Firefox browser. The browser's address bar shows the URL `http://demo/~demo/exp_imp.php`. The Zabbix logo is visible in the top left, and navigation links like 'Help', 'Get support', 'Print', 'Profile', and 'Logout' are in the top right. The main navigation menu includes 'Monitoring', 'Inventory', 'Reports', 'Configuration', and 'Administration'. The 'Configuration' menu is expanded, showing 'General', 'Web', 'Hosts', 'Items', 'Triggers', 'Actions', 'Graphs', 'Screens', 'Maps', 'IT services', 'Discovery', and 'Export/Import'. The 'Export/Import' page is active, displaying a table of elements for the host 'Template_Server'.

Host	Elements	
Template_Server	Item	Agent ping
	Item	Incoming traffic on interface \$1
	Item	Outgoing traffic on interface \$1
	Item	Host boot time
	Item	Processor load
	Item	CPU \$2 time (\$3)
	Item	CPU \$2 time (\$3)
	Item	CPU \$2 time (\$3)
	Item	Hostname
	Item	Host uptime
	Trigger	Processor load is too high on {HOSTNAME}
	Graph	Network stats
	Graph	CPU times
	Graph	Processor load

At the bottom of the table, there are buttons for 'Back', 'Refresh', and 'Export'. The footer of the page displays 'ZABBIX 1.6 Copyright 2001-2008 by SIA Zabbix' and 'Connected as 'Admin''. The status bar at the very bottom shows 'Done'.

17.2.12.2. インポート

このスクリーンは、ホスト関連のデータの XML インポートを実行するために使用します。



ディスカバリルールの属性を以下に示します。

パラメータ	説明
[ファイルをインポート]	インポートするXMLファイル。
[ルール]	各タイプの項目に対するルールセット。 [存在する場合] - 項目がすでに存在する場合にアクションを実行します。 [存在しない場合] - 項目が存在しない場合にアクションを実行します。 指定できるアクションは以下のとおりです。 [更新] - 既存の項目を更新します。 [追加] - 項目を追加します。 [スキップ] - 新しいデータを処理しません。

選択したファイルをインポートするには、[インポート]をクリックします。

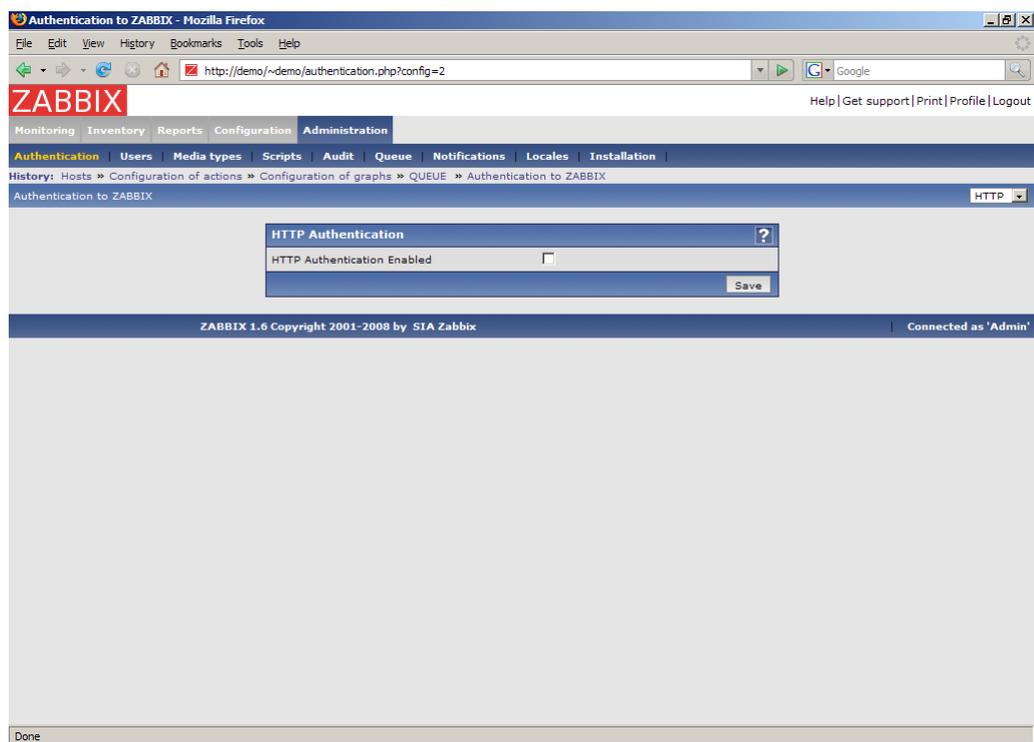
17.3. 管理

[管理]タブは、特権管理者のユーザのみが使用できます。

17.3.1. 認証

17.3.1.1. HTTP

このスクリーンは、Apache ベースの(HTTP)認証を有効にするために使用します。認証では、ユーザ名とパスワードがチェックされます。ユーザはZABBIXにも存在する必要がありますが、そのユーザの ZABBIX パスワードは使用されません。



設定パラメータを以下に示します。

パラメータ	説明
[HTTP認証の有効化]	Apacheベースの認証を有効にするかどうかを定義します。

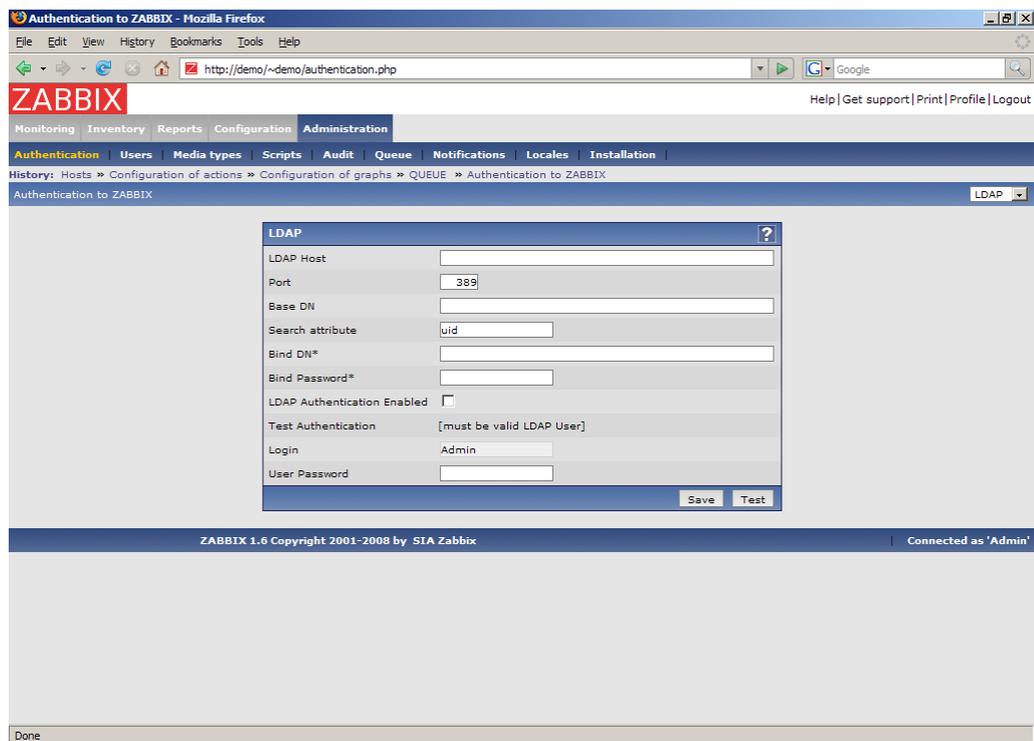
注: このパラメータをオンにする前に、Apache認証が設定され、適切に動作していることを必ず確認してください。

注: Apache認証では、全てのユーザ([GUIアクセス]が[インターナル]に設定されているユーザを含む)がZABBIXではなくApacheによって承認されます。

17.3.1.2. LDAP

このスクリーンは、外部 LDAP 認証を有効にするために使用します。認証では、ユーザ名とパスワードがチェックされます。ユーザは ZABBIX にも存在する必要がありますが、そのユーザの ZABBIX パスワードは使用されません。

ZABBIX LDAP 認証を動作させるには、少なくとも Microsoft Active Directory と OpenLDAP が必要です。



設定パラメータを以下に示します。

パラメータ	説明
[LDAPホスト]	LDAPサーバの名前。例: ldap://ldap.zabbix.com セキュリティで保護されたLDAPサーバの場合は、以下のよう にldaps//...を使用します。 ldaps://ldap.zabbix.com
[ポート]	LDAPサーバのポート。デフォルトは389です。 セキュリティで保護されたLDAP接続のポート番号は通常636 です。
[Base DN]	ou=Users、ou=system
[Search attribute]	uid
[Bind DN]	uid=Admin、ou=system
[Bind Password]	LDAPサーバへのバインド用のパスワード。
[LDAP認証の有効化]	LDAP認証を有効にします。
[認証のテスト]	-
[ログイン]	テストユーザの名前。ユーザはLDAPに存在する必要があります。
[ユーザパスワード]	テストユーザのLDAPパスワード。テストユーザを認証できな い場合は、LDAP認証は有効になりません。

注: 一部のユーザグループは引き続きZABBIXによって承認できます。これらのグループに対しては、[GUIアクセス]が[インターナル]に設定されている必要があります。

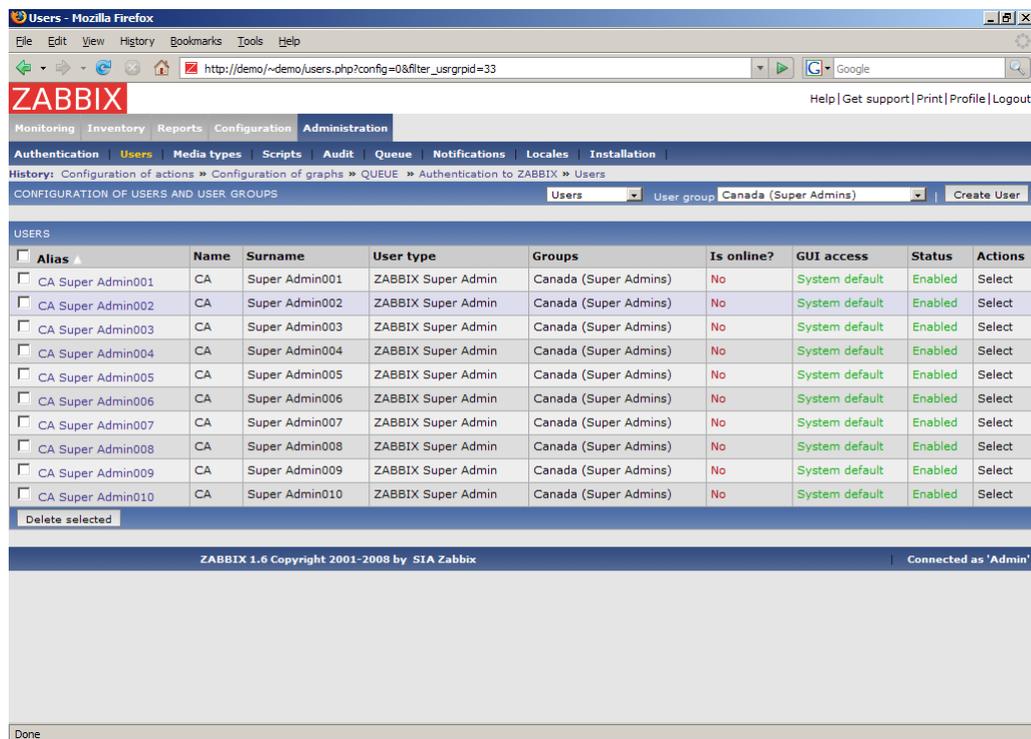
17.3.2. ユーザ

17.3.2.1. ユーザ

このスクリーンは、ZABBIX ユーザを管理するために使用します。

ユーザのリスト

ユーザのリストが表示されます。



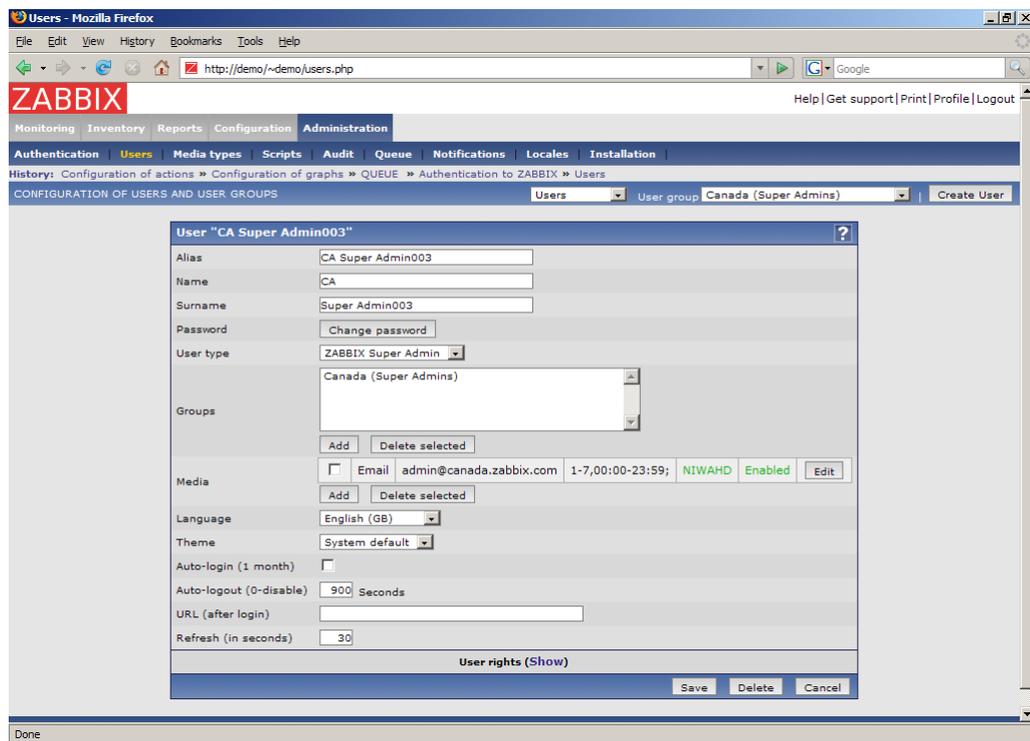
表示されるデータを以下に示します。

パラメータ	説明
[エイリアス]	ユーザの省略名(つまり、ログイン名)。
[名前]	ユーザ名。
[名字]	ユーザの名字。
[ユーザの種類]	ユーザの種類。以下のいずれかです。 ZABBIXユーザ ZABBIX管理者 ZABBIX特権管理者
[グループ]	ユーザが属する全てのグループのリスト。
[ログイン状態]	ユーザのログイン状態。
[GUIアクセス]	GUIへのアクセス(ユーザグループの設定に依存)。 [システムデフォルト] - ZABBIX、HTTP認証、LDAP認証が使

	用されます。 [インターナル] - ユーザはシステム設定に関係なくZABBIXによって認証されます。 [無効] - GUIアクセスは対象のユーザに制限されます。
[ステータス]	ユーザのステータス(ユーザグループの設定に依存)。 [有効] - ユーザは有効です。 [無効] - ユーザは無効です。ユーザはZABBIXで無視されません。
[アクション]	

ユーザの設定

このスクリーンにはユーザの詳細が表示され、ユーザ属性を変更できます。



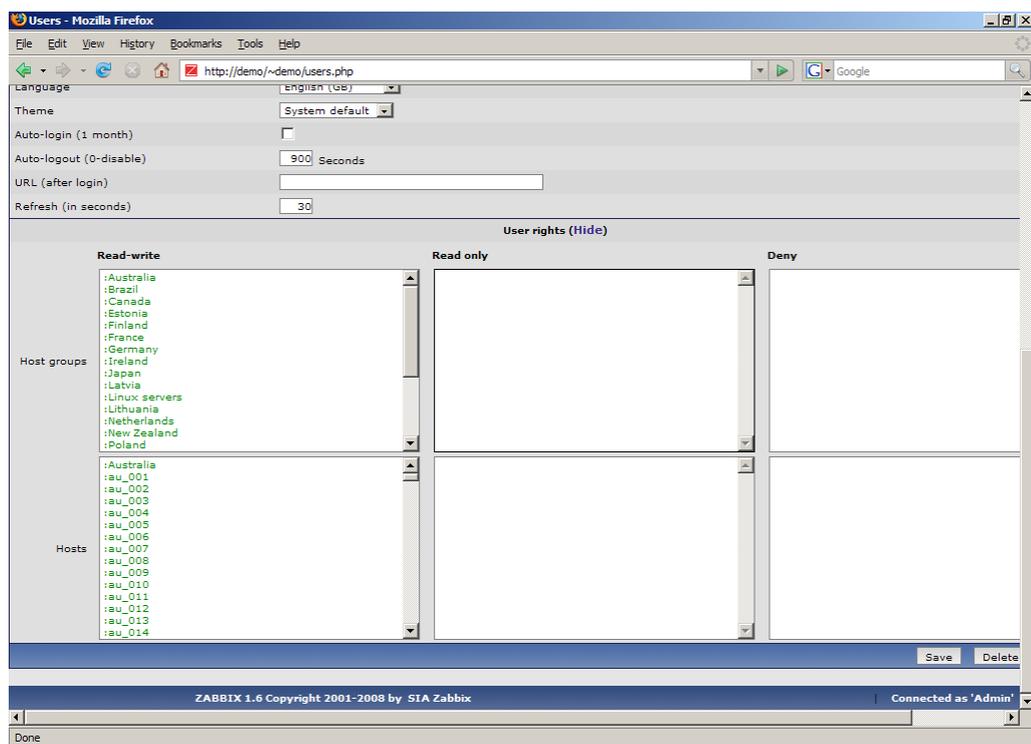
設定パラメータを以下に示します。

パラメータ	説明
[エイリアス]	ユーザの省略名(つまり、ログイン名)。一意である必要があります。
[名前]	ユーザ名。
[名字]	ユーザの名字。
[ユーザの種類]	ユーザの種類。以下のいずれかです。 [ZABBIXユーザ] - [監視データ]タブにのみアクセスできます。 [ZABBIX管理者] - [監視]および[設定]タブにアクセスできます。 [ZABBIX特権管理者] - [管理]タブを含む全ての情報にアクセスできます。
[グループ]	ユーザが属する全てのグループのリスト。
[メディア]	全てのメディアのリスト。メディアは、ZABBIXで通知を送信するために使用されます。

[言語]	ZABBIX GUIの言語。
[テーマ]	GUIの表示形式を定義します。 [システムデフォルト] - システム設定が使用されます。 [Original blue] - 標準の青のテーマが使用されます。 [Black & Blue] - 代替のテーマが使用されます。
[自動ログイン (1ヶ月)]	ZABBIXにログイン情報を記憶させる場合に有効にします。そのためにブラウザのCookieが使用されます。
[自動ログアウト] (0 - 無効)	ユーザがN秒間何も操作しないと、自動的にログアウトします。自動ログアウトを無効にするには、0に設定します。
[ログイン後のURL]	ログインが正常に完了すると、このURLが自動的に表示されます。
[更新(秒)]	グラフ、スクリーン、プレーンテキストデータなどを対象とした更新間隔。 無効にするには、0に設定します。

[ユーザの権限(User Right)]の[表示]をクリックすると、ユーザの権限が表示されます。ここでユーザの権限を変更することはできません。権限はユーザグループのメンバーシップに依存します。

この情報は読み取り専用です。

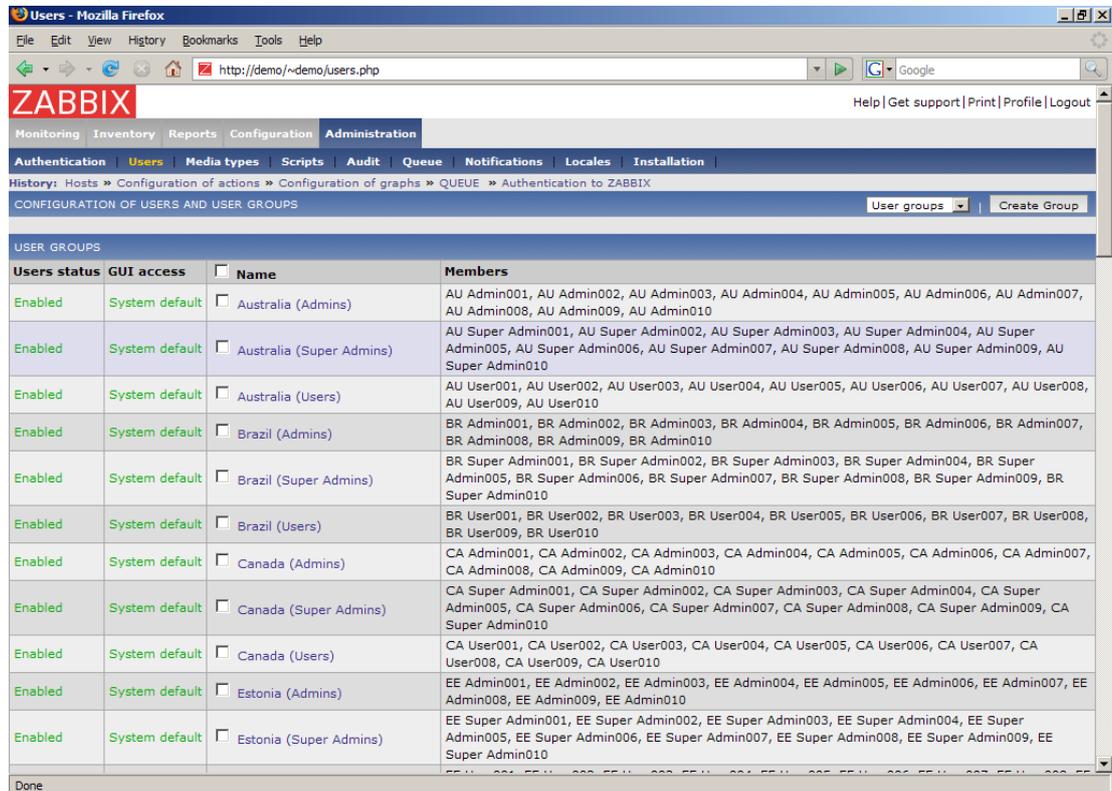


17.3.2.2. ユーザグループ

このスクリーンは、ZABBIX ユーザグループを管理するために使用します。

ユーザグループのリスト

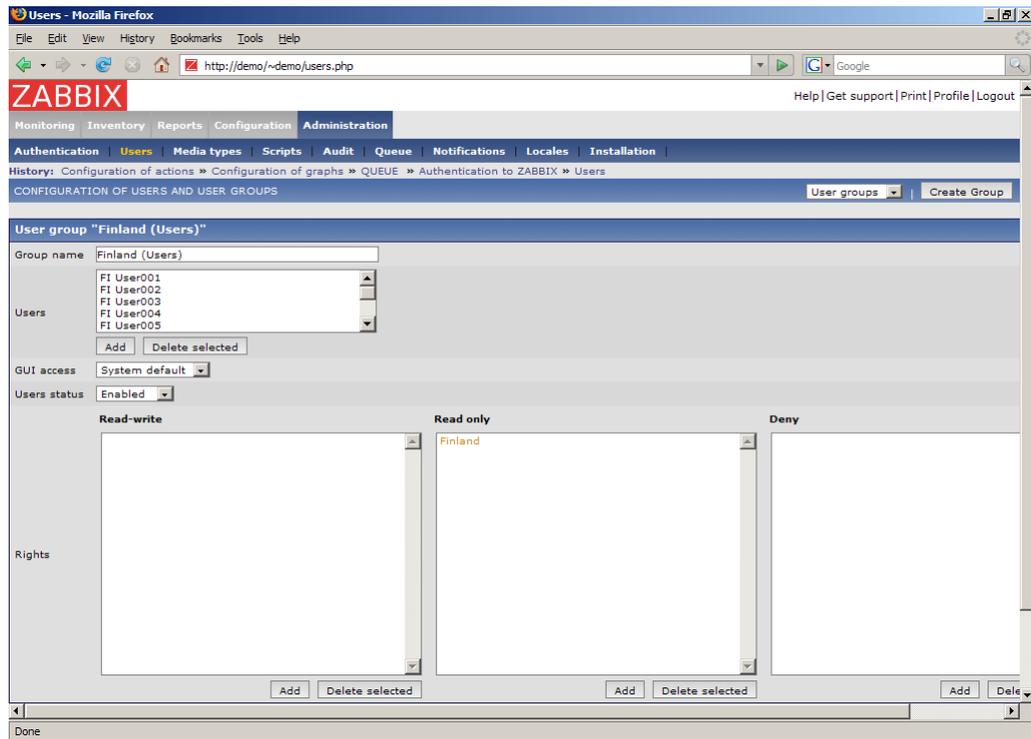
ユーザグループのリストが表示されます。



表示されるデータを以下に示します。

パラメータ	説明
[名前]	ホストグループ名。一意である必要があります。
[ユーザの状態]	[有効] - ユーザは有効です。 [無効] - グループの全てのユーザが無効です。
[GUIアクセス]	ユーザの認証方法が表示されます。 [システムデフォルト] - デフォルトの認証が使用されます。 [インターナル] - ZABBIX認証が使用されます。 [無効] - ZABBIX GUIへのアクセスは禁止されます。
[メンバー]	グループメンバーのリスト。

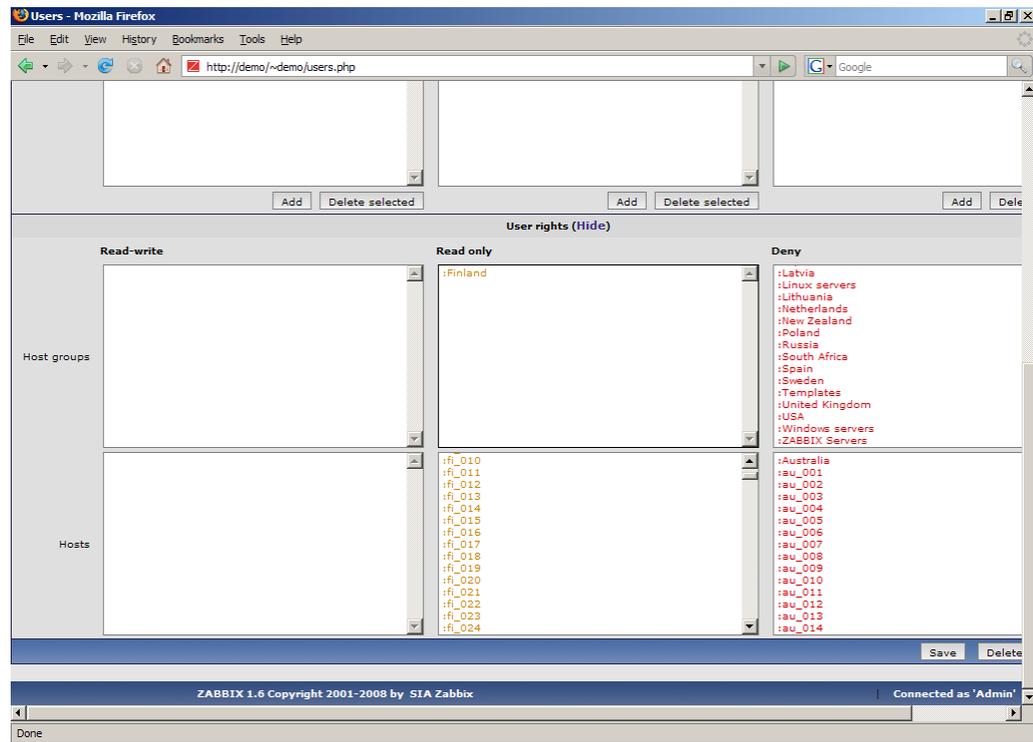
ユーザグループの設定



設定パラメータを以下に示します。

パラメータ	説明
[グループ名]	一意なグループ名。
[ユーザ]	このグループに属するメンバーのリスト。
[GUIアクセス]	グループのユーザの認証方法。 [システムデフォルト] - デフォルトの認証が使用されます。 [インターナル] - ZABBIX認証が使用されます。 [無効] - ZABBIX GUI へのアクセスは禁止されます。
[ユーザの状態]	グループメンバーのステータス。 [有効] - ユーザは有効です。 [無効] - ユーザは無効です。
[権限]	各種ホストパーミッションの3つのリスト。 [読書可能] - 読み書き可能なアクセス権を持つホストグループ。 [読込専用] - 読み取り専用のアクセス権を持つホストグループ。 [拒否] - アクセスが拒否されているホストグループ。

[ユーザの権限(User rights)]の[表示]をクリックすると、ユーザグループが持つパーミッションが表示されます。



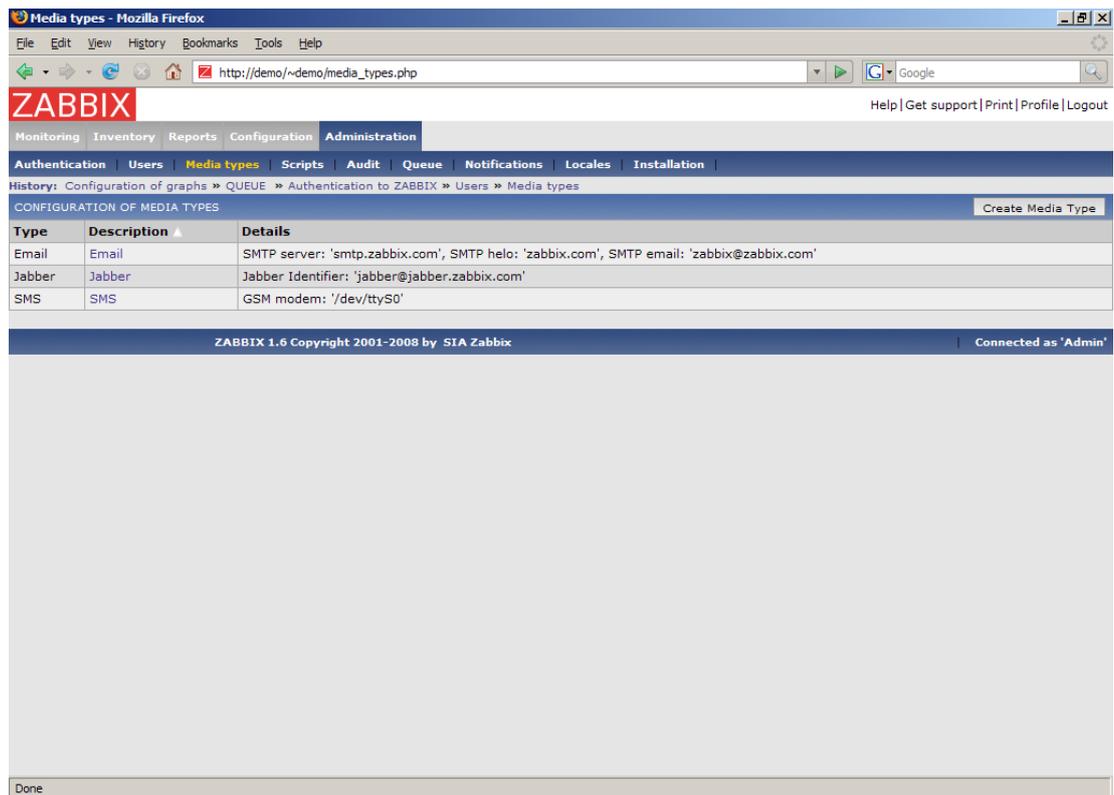
17.3.3. メディアタイプ

17.3.3.1. メディアタイプ

このスクリーンは、ZABBIX メディアを管理するために使用します。

メディアタイプのリスト

メディアタイプのリストが表示されます。メディアタイプは、ユーザ通知の配信方法です。

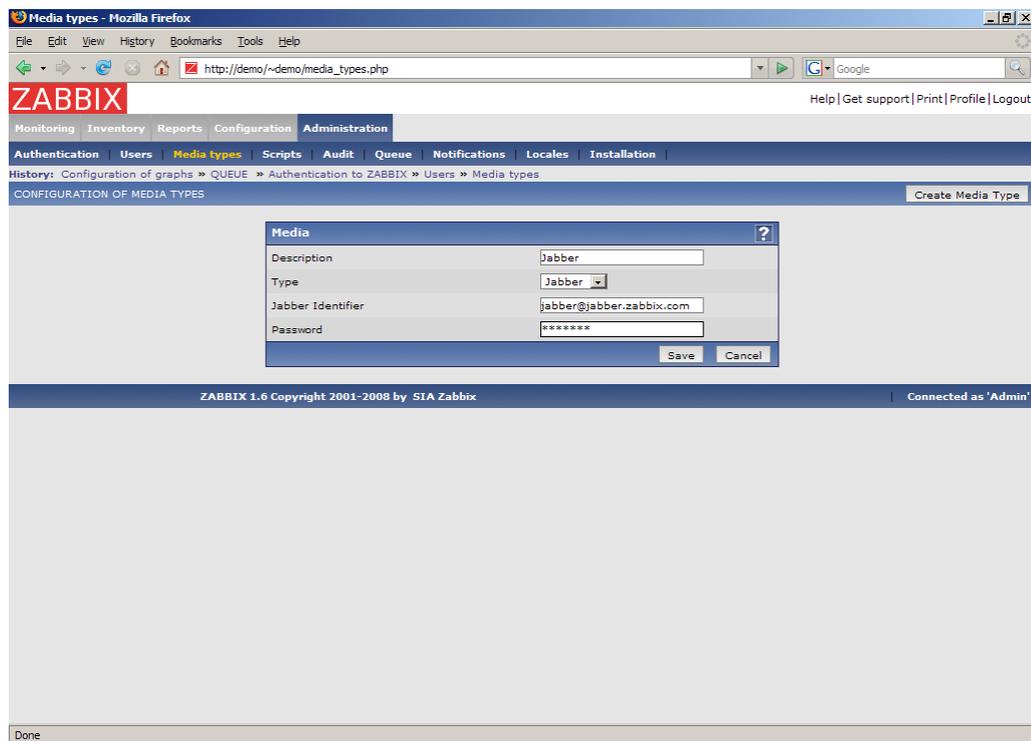


表示されるデータを以下に示します。

パラメータ	説明
[タイプ]	メディアタイプ。 [メール] - メール通知 [SMS] - シリアルGSMモデムを使用して送信されるSMS通知 [Jabber] - Jabber通知 [スクリプト] - スクリプトベースの通知
[名前]	メディアの名前。
[詳細]	設定の詳細(メディアタイプに依存)。

メディアの設定

このスクリーンにはメディアの詳細が表示され、メディア属性を変更できます。



設定パラメータを以下に示します。

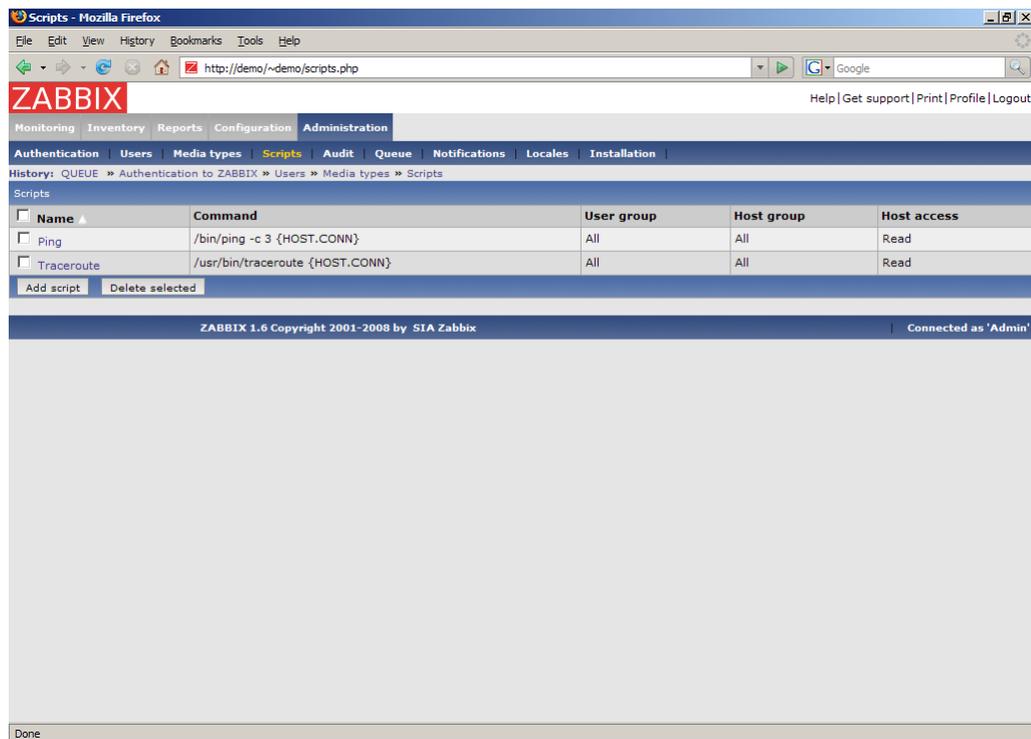
パラメータ	説明
[名前]	一意なメディア名。
[タイプ]	メディアタイプ。 [メール] - メール通知 [SMTPサーバ] - サーバ名 [SMTP helo] - hello文字列(通常はドメイン名) [送信元メールアドレス] - 送信元メールアドレス [SMS] - シリアルGSMモデムを使用して送信されるSMS通知 [GSMモデム] - GSMモデムのシリアルデバイス名 [Jabber] - Jabber通知 [JabberID] - Jabber ID [パスワード] - Jabber IDのパスワード [スクリプト] - スクリプトベースの通知 [スクリプト名] - カスタムスクリプトの名前

17.3.4. スクリプト

このスクリーンは、ユーザ定義のスクリプトを管理するために使用します。スクリプトは ZABBIX サーバサイドで実行されます。ホストがプロキシで監視される場合でも同様です。

スクリプトのリスト

ZABBIX で認識されているスクリプトのリストが表示されます。ZABBIX ユーザは、パーミッションに応じて、特定のスクリーンでホストをクリックすることでフロントエンドからスクリプトを実行できます。

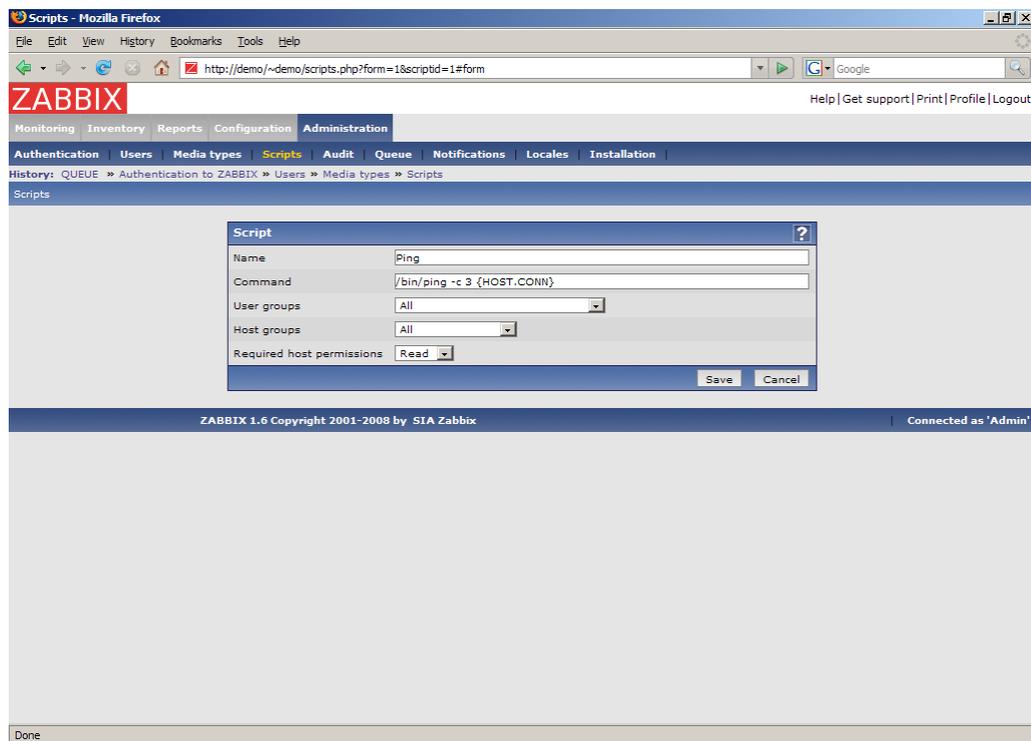


表示されるデータを以下に示します。

パラメータ	説明
[名前]	一意なスクリプト名。
[コマンド]	実行されるコマンド。
[ユーザグループ]	ユーザグループのメンバーのみがスクリプトを使用できます。
[ホストグループ]	ホストグループのホストのみがスクリプトを使用できます。
[ホストアクセス]	[読込] - ユーザには、スクリプトを実行するホストに対する読み取りパーミッションが必要です。 [書込] - ユーザには、スクリプトを実行するホストに対する書き込みパーミッションが必要です。

スクリプトの設定

このスクリーンにはスクリプトの詳細が表示され、スクリプト属性を変更できます。



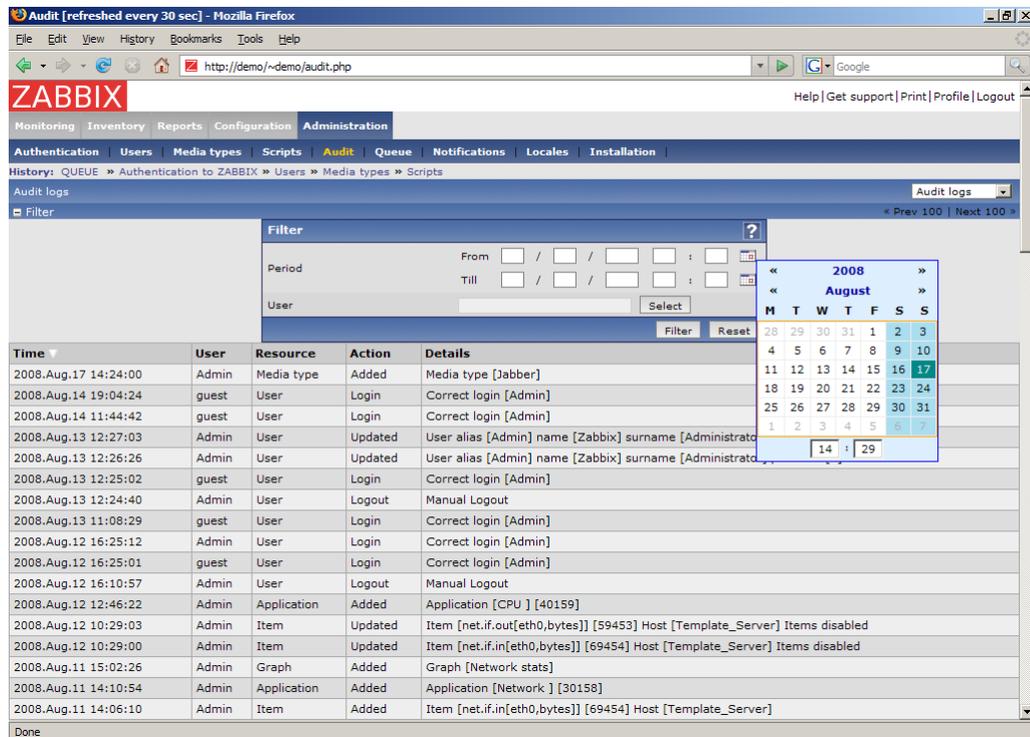
設定パラメータを以下に示します。

パラメータ	説明
[名前]	一意なスクリプト名。
[コマンド]	ユーザの要求によって実行されるコマンドに対する完全なパッチ。コマンドはZABBIXサーバサイドで実行されます。ここでは以下のマクロがサポートされています。 {HOST.CONN} {HOST.DNS} {IPADDRESS} 例: /bin/ping-c 3 {HOST.CONN} IPMIコマンドの場合は、以下の特別な構文を使用する必要があります。 IPMI <ipmiコントロール> [値] 例: IPMI power off
[ユーザグループ]	ユーザグループのメンバーのみがスクリプトを使用できます。
[ホストグループ]	ホストグループのホストのみがスクリプトを使用できます。
[ホストアクセス]	[読込] - ユーザには、スクリプトを実行するホストに対する読み取りパーミッションが必要です。 [書込] - ユーザには、スクリプトを実行するホストに対する書き込みパーミッションが必要です。

17.3.5. 監査

このスクリーンは、フロントエンドの監査レコードとユーザに送信された通知のリストを表示するために使用します。

監査ログ

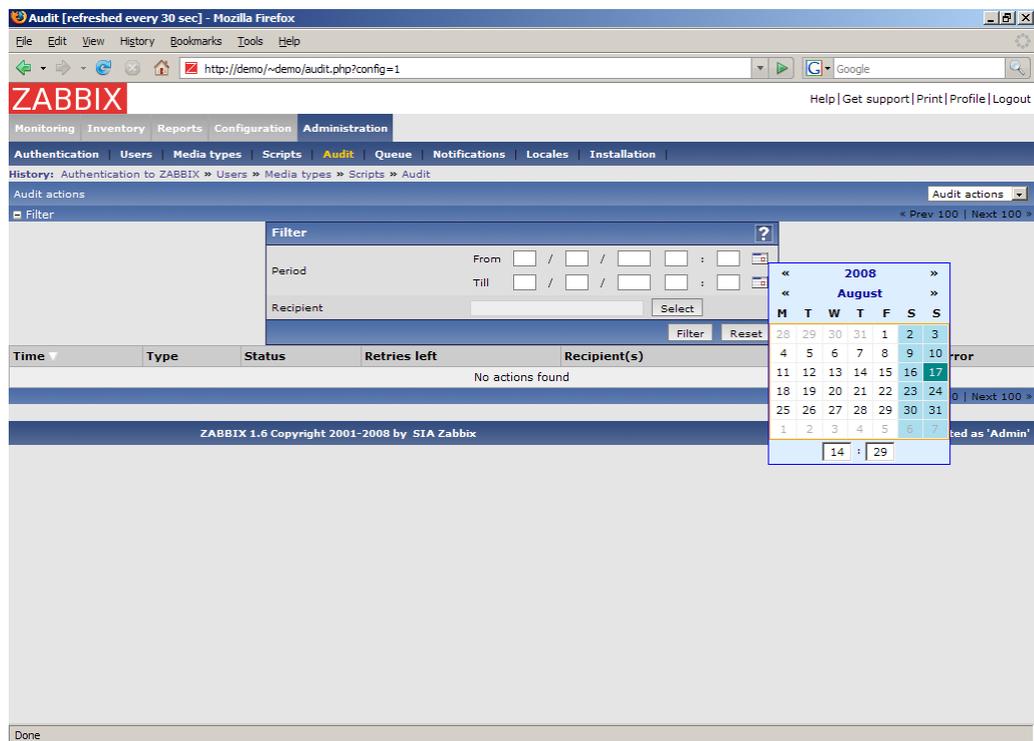


表示されるデータを以下に示します。

パラメータ	説明
[時間]	アクションが実行されたタイムスタンプ。
[ユーザ]	ユーザ名。
[リソース]	影響を受けたオブジェクト。 [アプリケーション] [グラフ] [ホスト] [アイテム] [ユーザ]
[アクション]	実行されたアクション。 [追加] [ログイン] [ログアウト] [削除(Removed)] [更新]
[詳細]	アクションの詳細情報。

監査アクション

このスクリーンには、通知とリモートコマンドの履歴が表示されます。



表示されるデータを以下に示します。

パラメータ	説明
[時間]	アクションが実行されたタイムスタンプ。
[タイプ]	実行されたオペレーションのタイプ。 [通知] [リモートコマンド]
[ステータス]	ステータス。 [未送信] [送信済み]
[リトライの残り回数]	リトライの残り回数。
[送信先]	送信先のリスト。
[メッセージ]	通知に使用されるメッセージ。
[エラー]	通知が送信されなかった場合のエラー。

17.3.6. キュー

キューによって ZABBIX のパフォーマンスに関する情報が提供されます。

概要

このビューには、ZABBIX サーバおよびプロキシを含む ZABBIX の全体的なパフォーマンスに関する情報が表示されます。

Items	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
ZABBIX agent	2025	5028	0	0	0	0
ZABBIX agent (active)	0	0	0	0	0	0
SNMPv1 agent	0	0	0	0	0	0
SNMPv2 agent	0	0	0	0	0	0
SNMPv3 agent	0	0	0	0	0	0
Simple check	0	0	0	0	0	0
ZABBIX internal	0	0	0	0	0	0
ZABBIX aggregate	0	0	0	0	0	0
External check	0	0	0	0	0	0

各アイテムタイプについて、以下のデータが表示されます。

パラメータ	説明
[アイテム]	アイテムのタイプ。
[5秒]	データは5~10秒間遅延しています。
[10秒]	データは10~30秒間遅延しています。
[30秒]	データは 30~60 秒間遅延しています。
[1分]	データは1~5分間遅延しています。
[5分]	データは 5~10 分間遅延しています。
[10分以上]	データは 10 分以上遅延しています。

[プロキシごとの概要]

このビューには、ZABBIX サーバおよびプロキシのパフォーマンスに関する詳細情報が表示されます。

The screenshot shows the ZABBIX web interface. The main content is a table titled "QUEUE OF ITEMS TO BE UPDATED". The table has columns for "Proxy", "5 seconds", "10 seconds", "30 seconds", "1 minute", "5 minutes", and "More than 10 minutes". The data is as follows:

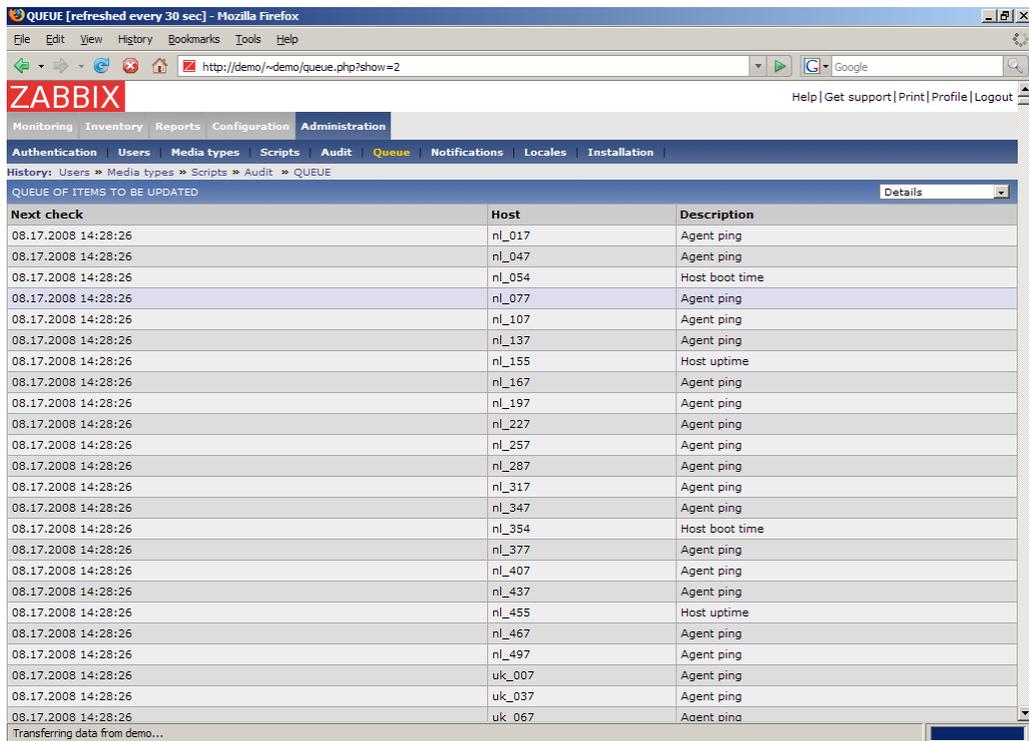
Proxy	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
Australia	101	273	0	0	0	0
Brazil	99	211	0	0	0	0
Canada	105	289	0	0	0	0
Estonia	99	271	0	0	0	0
Finland	101	294	0	0	0	0
France	104	264	0	0	0	0
Germany	96	280	0	0	0	0
Ireland	105	229	0	0	0	0
Japan	101	273	0	0	0	0
Latvia	100	256	0	0	0	0
Lithuania	104	222	0	0	0	0
Netherlands	101	252	0	0	0	0
New Zealand	99	231	0	0	0	0
Poland	105	289	0	0	0	0
Russia	101	231	0	0	0	0
South Africa	99	251	0	0	0	0
Spain	100	313	0	0	0	0
Sweden	109	233	0	0	0	0
United Kingdom	100	237	0	0	0	0
USA	96	320	0	0	0	0
Server	0	0	0	0	0	0
Total: 21						

各プロキシおよびローカル ZABBIX サーバについて、以下のデータが表示されます。

パラメータ	説明
[プロキシ]	プロキシ名または[サーバ]。[サーバ]は最後に表示され、ローカルサーバの統計情報が示されます。

【詳細】

このビューには、遅延アイテムに関する詳細情報が表示されます。



以下の詳細を含むアイテムのリストが表示されます。

パラメータ	説明
[次のチェック]	次のデータ取得の予想タイムスタンプ。常に過去のタイムスタンプが表示されます。
[ホスト]	ホスト名。
[名前]	アイテム名。

17.3.7. 通知

各ユーザに送信された通知の数をメディアタイプ別にグループ化したレポートです。

From	Till	Admin	RU Admin001	RU Admin002	RU Admin003	RU Admin004	RU Admin005	RU Admin006	RU Admin007	RU Admin008	RU Admin009	RU Admin010	RU Super Admin001	RU Super Admin002
31 Dec 2007 00:00	07 Jan 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
07 Jan 2008 00:00	14 Jan 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
14 Jan 2008 00:00	21 Jan 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
21 Jan 2008 00:00	28 Jan 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
28 Jan 2008 00:00	04 Feb 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
04 Feb 2008 00:00	11 Feb 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
11 Feb 2008 00:00	18 Feb 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
18 Feb 2008 00:00	25 Feb 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
25 Feb 2008 00:00	03 Mar 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
03 Mar 2008 00:00	10 Mar 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
10 Mar 2008 00:00	17 Mar 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
17 Mar 2008 00:00	24 Mar 2008 00:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
24 Mar 2008 00:00	31 Mar 2008 01:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
31 Mar 2008 01:00	07 Apr 2008 01:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
07 Apr 2008 01:00	14 Apr 2008 01:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
14 Apr 2008 01:00	21 Apr 2008 01:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)
21 Apr 2008 01:00	28 Apr 2008 01:00	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)	0 (0/0/0)

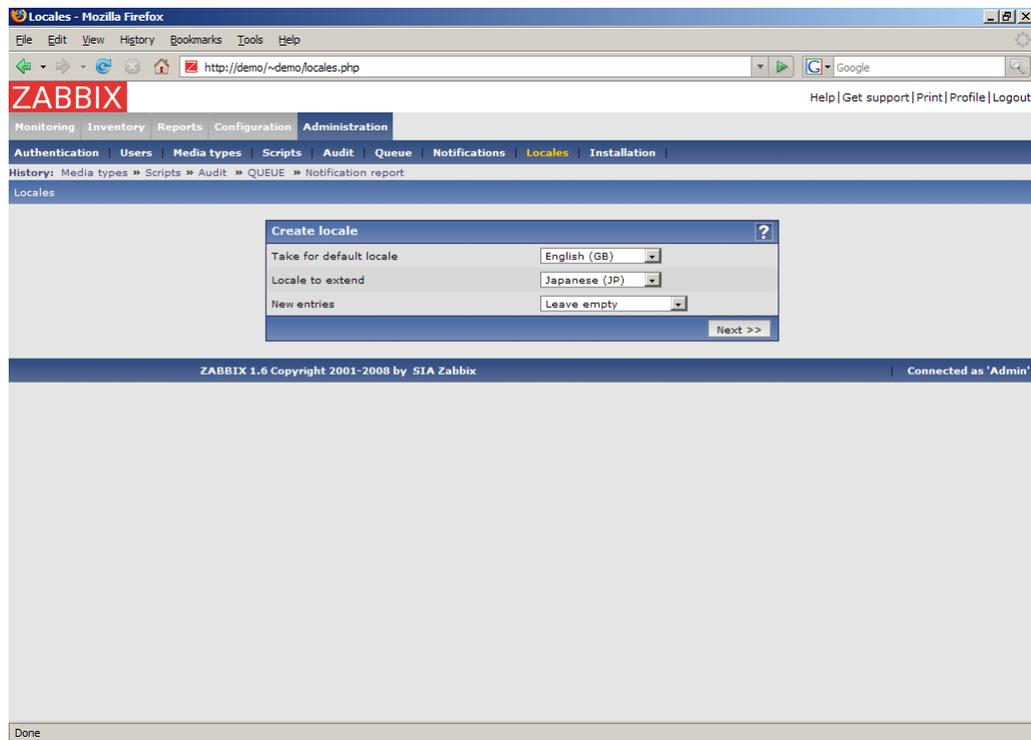
各ユーザに対する通知の数がメディアタイプ別に表示されます。

17.3.8. ロケール

[ロケール]には、ZABBIX フロントエンドの翻訳を簡単に編集できる機能が用意されています。

ロケールの選択

処理が必要なロケールを選択します。

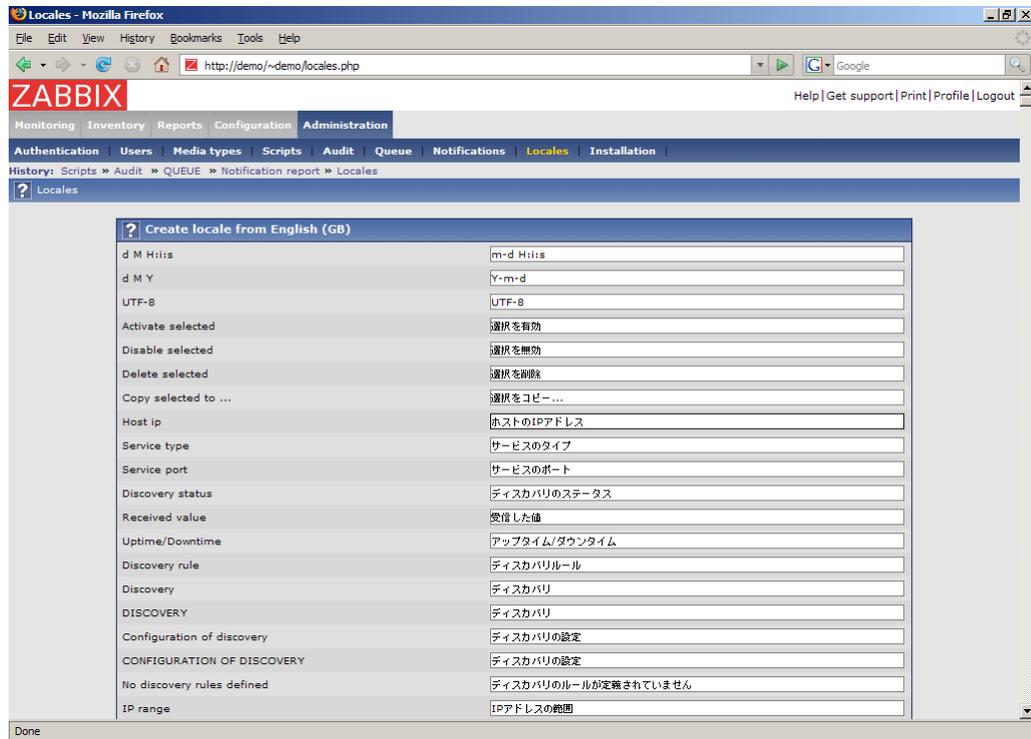


パラメータを以下に示します。

パラメータ	説明
[デフォルトロケールの選択 (Take for default locale)]	選択したロケールが基準として使用されます。
[拡張するロケール(Locale to extend)]	改善する言語を選択します。
[新規エントリ(New entries)]	<p>[追加しない] - 項目が翻訳されていない場合は無視されます。</p> <p>[空白のまま] - 項目が翻訳されていない場合は空白のままになります。</p> <p>[デフォルト値を表示] - 項目が翻訳されていない場合はデフォルト値が翻訳として使用されます。</p>

翻訳フォーム

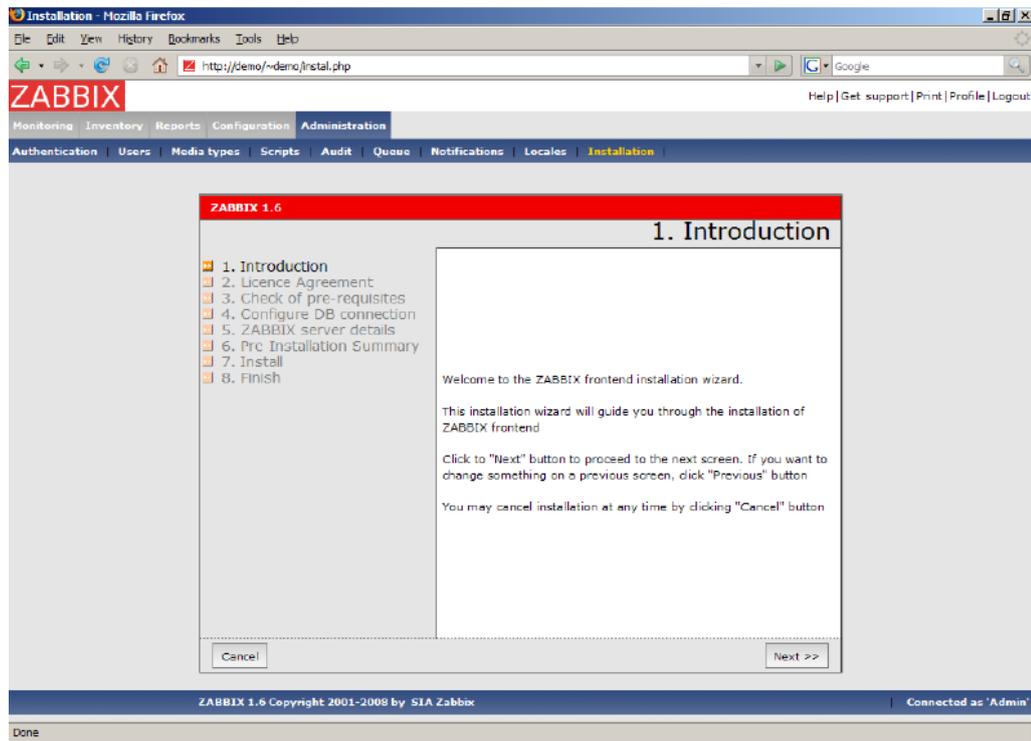
このフォームでは、ZABBIX フロントエンドで使用される語句を翻訳できます。左側にはデフォルトの言語が表示され、右側には翻訳後の語句が表示されます。



翻訳が終了したら、[ダウンロード]ボタンをクリックして翻訳ファイルを取得します。この翻訳ファイルで **include/locales** 内のファイルを置換します。

17.3.9. インストール

このスクリーンでは、ZABBIX フロントエンドの設定ファイルを作成できます。



18. パフォーマンスチューニング

18.1. 実際の設定

ZABBIX 1.0 がインストールされたサーバ(RedHat Linux 8.0、kernel 2.4.18-14、MySQL/MyISAM 3.23.54a-4、Pentium IV 1.5GHz、256MB、IDE)は、監視するサーバから 1 秒あたり 200 個以上のパラメータを収集することができます(ネットワークの遅延はないと仮定)。

このハードウェア上で動作する ZABBIX で監視できるサーバの数は、監視するパラメータの数と ZABBIX がパラメータを収集する頻度によって決まります。監視するサーバ 1 台あたり 10 個のパラメータを収集するものとします。更新間隔は 30 秒です。この場合、単純に計算すると、ZABBIX が処理可能なサーバの数は 600 台(または 6000 個のチェック)になります。パラメータの更新間隔が 1 分の場合は、同じハードウェア設定で 1200 台(=600 台×2)のサーバを処理できます。これらの計算結果は、監視する値を全て、必要なときに遅延なしで取得できることを前提としています。この前提が不要な場合は、監視できるサーバの台数を 5~10 倍に増やすことができます。

18.2. パフォーマンスチューニング

パフォーマンスを最大限に高めるように ZABBIX システムを適切にチューニングすることが非常に重要です。

18.2.1. ハードウェア

ハードウェアに関する一般的なアドバイス:

- できるかぎり高速なプロセッサを使用する
- IDE と SATA より SCSI または SAS を選択する(ただしユーティリティ `hdparm` を使用すると IDE ディスクのパフォーマンスを大幅に高めることが可能)
- 7200RPM よりも 10K RPM、10K RPM よりも 15K RPM を選択する
- 高速 RAID ストレージを使用する
- 高速イーサネットアダプタを使用する
- メモリはできるかぎり増やす

18.2.2. オペレーティングシステム

- OS の最新バージョン(安定版)を使用する
- カーネルから不要な機能を削除する
- カーネルパラメータを調整する

ZABBIX 設定パラメータ

最適なパフォーマンスを得るために、多くのパラメータを調整できます。

`zabbix_server`

`StartPollers`

原則として、このパラメータの値はできるかぎり小さくします。`zabbix_server` のインスタンスを 1 つ追加するたびに既知のオーバーヘッドが増加すると同時に、並列処理数が増加します。最適なインスタンス数になるのは、キューに存在するパラメータの平均数が最も少ない(理想は常に 0)ときです。

この値は、内部チェック `zabbix[queue]` を使用して監視できます。

`DebugLevel`

最適な値は 3 です。

`DBSocket`

これは MySQL のみ利用できます。データベース接続に `DBSocket` を使用することをお奨めします。これが最も速く、最も安全な方法です。

18.2.3. データベースエンジン

ZABBIX のチューニングで最も重要なのはデータベースエンジンでしょう。ZABBIX は、データベースエンジンの可用性とパフォーマンスに大きく依存しています。

- 最も高速なデータベースエンジン(すなわち MySQL)を使用する
- データベースエンジンの安定版リリースを使用する
- 最高のパフォーマンスを得るためにソースから MySQL または PostgreSQL をリビルドする
- MySQL または PostgreSQL のドキュメントに記載されているパフォーマンスチューニング手順を実行する
- MySQL の場合、InnoDB テーブル構造を使用する
- InnoDB を使用すると並行処理度が高まるので、MyISAM を使用した場合の 1.5 倍の速度で ZABBIX は動作するが、大量の CPU パワーを必要とする
- 別個のハードディスクにデータベーステーブルを保持する
- 「history」、「history_str」、「items」、「functions」、「triggers」、「trends」が最も頻繁に使用されるテーブルになる
- 大規模システムでは MySQL の一時ファイルを `tmpfs` に配置することが推奨される

18.2.4. 一般的なアドバイス

- 必要なパラメータだけを監視する
- 全てのアイテムの[更新間隔]を調整する。更新間隔を短く設定すれば詳細なグラフを作成できるが、ZABBIX が過負荷になる可能性がある
- デフォルトテンプレートのパラメータを調整する
- `housekeeper` のパラメータを調整する
- 同じ情報を返すパラメータは監視しない

例: `system[procload]`、`system[procload5]`、および `system[procload15]` は全て `system[procload]` に含まれているので監視する必要はありません。

- 関数の引数として長期間が指定されているトリガーを使用することを避ける。たとえば、`max(3600)` の計算には `max(60)` を大幅に上回る時間がかかる

19. 具体的な設定例

19.1. 一般的な手法

19.1.1. サーバの可用性の監視

サーバの可用性を監視する方法として、少なくとも以下の 3 つ(またはその組み合わせ)が考えられます。

- ICMP ping(キー「icmpping」)
- キー「status」
- アクティブチェックだけを使用してホストの可用性を監視するトリガー関数 nodata()

19.1.2. WinPopUps によるアラートの送信

ZABBIX を Windows 上で実行している場合、WinPopUps を使用すると、イベント発生後すぐに通知を受け取ることができます。メールベースのアラートメッセージと組み合わせると便利な方法です。WinPopUps を有効にする詳細な手順については、https://sourceforge.net/forum/message.php?msg_id=2721722 を参照してください。

19.2. 具体的なアプリケーションの監視

19.2.1. AS/400

IBM AS/400 プラットフォームを監視するには、SNMP を使用します。詳細については、<http://publibb.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg244504.html?Open> を参照してください。

19.2.2. MySQL

設定ファイル misc/conf/zabbix_agentd.conf には、MySQL の監視に使用できるパラメータのリストが記述されています。

```
### Set of parameter for monitoring MySQL server (v3.23.42 and later)
### Change -u and add -p if required
#UserParameter=mysql[ping],mysqladmin -uroot ping|grep alive|wc -l
#UserParameter=mysql[uptime],mysqladmin -uroot status|cut f2 -d":"|cut -f1 -d"T"
#UserParameter=mysql[threads],mysqladmin -uroot status|cut f3 -d":"|cut -f1-d"Q"
#UserParameter=mysql[questions],mysqladmin -uroot status|cut f4 -d":"|cut -f1-d"S"
#UserParameter=mysql[slowqueries],mysqladmin -uroot status|cut f5 -d":"|cut -f1-d"O"
#UserParameter=mysql[qps],mysqladmin -uroot status|cut -f9 d":"
#UserParameter=version[mysql],mysql -V
```

* mysql[ping]

MySQL が動作中かどうかをチェック

戻り値: 0 - 停止中 1 - 動作中

* mysql[uptime]

MySQL の動作時間(秒単位)

* mysql[threads]

MySQL のスレッド数

* mysql[questions]

処理済みクエリの数

* mysql[slowqueries]

時間のかかるクエリの数

* mysql[qps]

1 秒あたりに処理するクエリの数

* mysql[version]

MySQL のバージョン

例: mysql Ver 11.16 Distrib 3.23.49, for pc-linux-gnu (i686)

19.2.3. Mikrotik ルータ

Mikrotik が提供する SNMP エージェントを使用します。詳細については、<http://www.mikrotik.com> を参照してください。

19.2.4. WIN32

ZABBIX ディストリビューションに付属する(コンパイル済み)ZABBIX W32 エージェントを使用します。

19.2.5. Novell

Novell の MRTG Extension Program for NetWare Server(MRTGEXT.NLM)エージェントを使用します。このエージェントは、ZABBIX が使用するプロトコルと互換性があります。入手するには、<http://forge.novell.com/modules/xfmod/project/?mrtgext> を参照してください。

アイテムは、[タイプ]に[ZABBIX エージェント]を指定し、MRTGEXT のマニュアルを参照してキーを指定する必要があります。

例:

* UTIL1

1 分間の平均 CPU 使用率

* CONNMAX

ライセンスされている接続の最大数

* VFKSys

ボリューム Sys の空き容量(バイト単位)

このエージェントがサポートするパラメータの詳細なリストについては、エージェントに付属する readme.txt を参照してください。

19.2.6. Tuxedo

UserParameter の定義で Tuxedo コマンドラインユーティリティ tadmin と qadmin を使用すると、サーバ/サービス/キューごとのパフォーマンスカウンタおよび Tuxedo リソースの可用性を取得できます。

19.2.7. Informix

Informix の標準ユーティリティ onstat を使用すると、Informix データベースのほとんど全ての情報を監視できます。また、Informix SNMP エージェントが提供する情報を取得できます。

19.2.8. JMX

まず jmx を監視できるように jvm を設定する必要があります。この操作が可能かどうかを調べるには、jdk に付属する Sun の jconsole ユーティリティを、jvm が動作するマシンを指定して実行します。接続できれば、jvm を設定できます。

tomcat 環境では、以下の jvm オプションを設定することで jvm を設定可能にできます。

- Dcom.sun.management.jmxremote ¥
- Dcom.sun.management.jmxremote.port=xxxxx ¥
- Dcom.sun.management.jmxremote.ssl=false ¥
- Dcom.sun.management.jmxremote.authenticate=true ¥
- Dcom.sun.management.jmxremote.password.file=/path/java/jre/lib/management/jmxremote.password"

ここでは、jmx サーバをポート XXXXX で実行すること、パスワード認証を使用すること、および jmxremote.password ファイルに保存されているパスワードを参照することを指示しています。詳細については、Sun の jconsole のマニュアルを参照してください。ssl を有効にすると接続の安全性を高めることができます。

必要な設定が完了したら、jconsole を実行して、現在公開されている情報を全て取得することができます。また、正常に接続できることも確認できます。jconsole を使用すると、[information]タブから特定の jmx 属性を照会する際に必要な情報を取得することもできます。

tomcat 環境では、jmx 属性の値を取得する(または jmx の動作を設定する)方法は 2 つあります。1 つは、tomcat が提供するサーブレットを使用する方法であり(jboss については不明)、もう 1 つは jmx コマンドラインツールを使用して正しい構文の要求を送信する方法です。

システムが使用するスレッドのピーク値を取得することを考えます。jconsole で jmx オブジェクトを展開し、[java.lang]->[Threading]を選択します。[Threading]を選択した後、[info]タブをクリックすると、mbean の名前として「java.lang:type=Threading」と表示されていることを確認できます。

tomcat 環境では、以下のコマンドを実行できます。

```
curl -s \
-u<jmxusername>:<jmxpassword>'http://<tomcat_hostname>/manager/jmxproxy/?qry=java.lang:type=Threading'
```

ここで、jmx のユーザ名とパスワードは、前述の jvm オプションで指定したファイルで定義されているユーザ名とパスワードを使用します。qry に渡しているのは jconsole で調べた文字列です。

このコマンドを実行すると、ここで指定した jmx キーで取得できる測定基準が全て出力されます。出力を解析して、必要な数値を取得します。

jmx インタフェースに http 要求を渡すことができるサーブレットが存在しない環境では、以下のようなコマンドラインツールを使用します。

```
./<pathTo>/java -jar /<pathTo>/cmdline-jmxclient.jar<jmxusername>:<jmxpassword> \
<jvmhostname>:<jmxport>java.lang:type=Threading PeakThreadCount
```

コマンドラインクライアントとの違いは、必要な属性を具体的に指定する必要がある点です。属性を指定しない場合、[Threading]の全ての属性のリストが出力されます。

この場合も、出力を解析して、必要な数値を取得します。

必要なデータを確実に取得できるようになったら、そのコマンドを ZABBIX の UserParameter に指

定めます。

例:

```
UserParameter=jvm.maxthreads, /usr/bin/curl
-s-u<jmxusername>:<jmxpassword>'http://<tomcat_hostname>/manager/jmxproxy/?qry=java
.lang:type=Threading' | /bin/awk '/^PeakThreadCount¥:/ { gsub( /^[^0123456789]/, "" ); print
$1 }'
```

または

```
UserParameter=jvm.maxthreads, /<pathTo>/java -jar /<pathTo>/cmdlinejmxclient.jar
<jmxusername>:<jmxhostname> <jvmhostname>:<jmxport>java.lang:type=Threading
PeakThreadCount | <some filter to grab just the number you need - left as an exercise to the
reader>
```

これで完了です。

java コマンドラインクライアントを使用するよりも、http 経由でサブレットを使用するほうが、処理を起動して情報を取得する際の負荷は軽減されます。

コマンドライン jmx クライアントは、以下の URL から入手できます。

<http://crawler.archive.org/cmdline-jmxclient/>

jvm を監視するための jmx の設定については、以下の URL を参照してください。

<http://java.sun.com/j2se/1.5.0/docs...ment/agent.html>

JMX の概要については、以下の URL を参照してください。

<http://java.sun.com/j2se/1.5.0/docs...verviewTOC.html>

追加情報: 1.5 jvm は snmp をサポートしていると考えられるので、選択肢はさらに増えます。

19.3. 統合

19.3.1. HP OpenView

OpenView サーバにメッセージを送信するように ZABBIX を設定するには、以下の手順を実行する必要があります。

ステップ 1 新しいメディアの定義

必要な情報を OpenView に送信するスクリプトを実行するメディアを定義します。

ステップ 2 新しいユーザの定義

定義したユーザはメディアとリンクする必要があります。

ステップ 3 アクションの設定

トリガーの全ての(または選択した)ステータス変更をユーザに送信するアクションを設定します。

ステップ 4 メディアスクリプトの作成

このスクリプトのロジックは以下のとおりです。トリガーが ON の場合、OpenView コマンド「`opcmsg -id application=<application> msg_grp=<msg_grp> object=<object> msg_text=<text>`」を実行します。このコマンドが返す一意なメッセージ ID は、別の場所、できれば ZABBIX データベースの新しいテーブルに保存する必要があります。トリガーが OFF の場合、データベースから取得したメッセージ ID を引数として「`opcmsg <message id>`」を実行する必要があります。

`opcmsg` と `opcmsg` の詳細については、OpenView の公式マニュアルを参照してください。このメディアスクリプトはここには掲載しません。

20. トラブルシューティング

20.1. エラーメッセージと警告メッセージ

何らかの問題が発生した場合、ZABBIX デーモンによってエラーメッセージと警告メッセージが生成されます。これらのメッセージは、設定パラメータに応じて、ログファイルまたは syslog に書き込まれます。

一部のメッセージには番号が付けられます。

以下の表は、番号付けされたメッセージとその詳細の完全なリストです。

エラー	メッセージ	詳細
Z3001	データベース '%s' への接続に失敗しました: [%d] %s(Connection to database '%s' failed: [%d] %s)	ZABBIXデーモンがデータベースへの接続を確立できません。追加情報は以下のとおりです。 データベース名 データベースエラーコード データベースエラー文字列
Z3002	データベース '%s' を作成できません: [%d] %s(Cannot create database '%s': [%d] %s)	ZABBIXデーモンがデータベースを作成できません。追加情報は以下のとおりです。 データベース名 データベースエラーコード データベースエラー文字列
Z3003	データベースへの接続が行われません。(No connection to the database.)	発生してはならない問題です。Zabbixチームに報告してください。
Z3004	データベースを閉じることができません: [%d] %s(Cannot close database: [%d] %s)	ZABBIXデーモンがデータベースへの接続を閉じることができません。追加情報は以下のとおりです。 データベースエラーコード データベースエラー文字列
Z3005	クエリが失敗しました: [%d] %s [%s](Query failed: [%d] %s [%s])	SQLクエリの実行が失敗しました。追加情報は以下のとおりです。 データベースエラーコード データベースエラー文字列 SQLクエリ文字列
Z3006	フェッチが失敗しました: [%d] %s(Fetch failed: [%d] %s)	レコードのフェッチが失敗しました。追加情報は以下のとおりです。 データベースエラーコード データベースエラー文字列

注: 番号付けされたエラーメッセージは、ZABBIX 1.8以降でサポートされています。

21. ライセンス

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its

recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope.

The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

* a) You must cause the modified files to carry prominent notices

stating that you changed the files and the date of any change.

* b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

* c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

* a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

* b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

* c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and

any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of

preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

22.ZABBIX への貢献

このプロジェクトに貢献するには以下のような方法があります。

■ 経験を共有する

ZABBIX の使用経験に関するご意見をお寄せください。ZABBIX を向上させるための非常に有益な情報となります。また、機能の追加要求のプライオリティの正当性の評価にも有効です。

■ レビューを書く

ZABBIX は比較的新しいソフトウェアであり、多くの人がある存在を認識していません。著名な技術メディアでこのプロジェクトが取り上げられることには、大きなメリットがあります。特に、既存のオープンソースや市販されている競合製品との比較レビューを歓迎します。筆者のサポートが必要な場合は、遠慮なくお問い合わせください。責任を持って対応します。

■ バグを報告する

ZABBIX にバグや不具合が見つかった場合は、是非ご報告ください。パッチや回避策を送付していただく必要はありません。

■ コードを作成する

パッチまたはコードの一部を送付する前に、以下のことを確認してください。

- 新しいコードが ZABBIX のコーディング規約に準拠していること
- サポートされている全てのプラットフォームで新しいコードがテストされ、動作すること。互換性に関する問題がある場合はそれを報告すること
- 新しい機能が明確に説明されていること
- 作成物に関して著作権問題が存在しないこと

コードを作成する前には、是非 ZABBIX の開発担当者にご相談ください。

この方針によって、ZABBIX の品質の高さが保証され、サポート内容も充実すると考えられます。

筆者の Amazon.com のウィッシュリスト

ZABBIX の使用にご満足いただけた場合は、Amazon.com の筆者のウィッシュリスト(以下の URL) から商品をご購入いただけますようお願いいたします。

<http://www.amazon.com/exec/obidos/wishlist/2MXT84ZA4ZNN>

以下は Amazon から商品をお送りいただいた方のリストです。

- Charlie Collins, USA
- Henrik Huhtinen, Finland
- Jaroslaw Pioro, Poland
- Julian Pawlowski, Virtual-Planet Group GmbH, Germany
- Ken Smith, USA
- Plushosting B.V., Netherlands
- Abdourahmane SECK, Senegal

コントリビュータ

コントリビュータの全リストについては、ZABBIX マニュアルを参照してください。

ウェブホスティング

ウェブホスティングは、Clearcut Networks によって無料で提供されています。オランダで比較的安価なホスティングをお求めの方にお奨めいたします。

23. クレジット

ZABBIX チームは、このプロジェクトのためにホスティングを提供していただいた <http://sourceforge.net> の方々や、修正や提案を送っていただいた全ての ZABBIX ユーザに感謝いたします。これらのフィードバックは、ZABBIX の品質向上に役立てられています。

23.1. ZABBIX の開発者

■ ALEXEI VLADISHEV

ZABBIX の作成者であり、PHP フロントエンドを含む ZABBIX コードの大部分を作成しています。

■ EUGENY GRIGORJEV

主に PHP フロントエンドと ZABBIX エージェントに関連して、重要な改良を多数手がけています。

23.2. ZABBIX に対するコントリビュータ

ここでは ZABBIX に貢献した方の一部が記載されています。

アルファベット順

■ ALEXANDER KALIMULIN

C や C 関数に関連する様々な問題をサポート

■ ALEXANDER KIRHENSTEIN

SCO で ZABBIX が動作するための修正を提案

■ ARTURS ABOLTINS

UNIX ソケットを使用して MySQL に接続するためのパッチ作成、MySQL サーバが停止した場合のグレースフルシャットダウンのサポート(未実装)、および ZABBIX スクリーンのアイデアと初期コードの提供

■ CHARLIE COLLINS

スタートアップスクリプトおよびマニュアルの大幅な書き直し。ご協力いただき誠にありがとうございます。

■ DENIS USTIMENKO

IP アドレスによる SNMP パラメータ照会のサポート

- DANIEL ESTER

SNMP のタイプ timetick の値のサポート

- DANIEL HIGGINS

メール送信ルーチンの改善を始めとする様々な変更

- ERIK CARLSEEN

様々なアイデアの提供

- EUGENY BACULA

様々な改善の提案

- FRANKY VAN LIEDEKERKE

Solaris における system[uptime]のサポートおよび様々な修正と提案

- HARALD HOLZER

RPM と zabbix.spec

- IGOR MICKO

大規模な監視環境で ZABBIX を実際に使用した経験に基づく様々なアイデアの提供

- JAEN-BAPTISTE MARIOTTE

テストのサポート

- JEFF REDDING

非 GCC コンパイラのサポート

- JOHN CRUNK

RedHat 8.0 のスタートアップスクリプト

- JOSH KONKOL

テストのサポート

■ JURGEN SCHMITZ

check_service_perf[*]のアイデア提供と実装

■ KASPARS CIKMACS

ZABBIX を実際に使用した経験に基づく様々な新しいアイデアの提供

■ LAURIS STIGLICS

[トリガーのステータス]の基準の選択

■ LUKAS MACURA

様々なアイデアの提供

■ MARC LEDENT

Solaris における proc_cnt[*]の最初の実装

■ MARIUSZ ...

Solaris 2.6 における system[procload]のサポート、グラフの改善、およびシステムマップの改善

■ MICHAL SUSZYCKI

autoconf および automake に関する問題のサポート

■ MIKE HOOLEHAN

ZABBIX マニュアルの正確性とわかりやすさの向上をサポート

■ OLIVER SIEGMAR

ウェブフロントエンドの SQL 文の修正

■ RICKARD PLARS

zabbix_suckerd のコアダンプの修正のサポート

■ SEBASTIEN "SLIX" LIENARD

sysmap.php のホストとアイコンの選択の修正を始めとする数々の修正

- SHAWN MARRIOTT

マニュアルの校正

- VICTOR KIRHENSTEIN

WIN32 プラットフォームにおけるネイティブ ZABBIX エージェント

ZABBIX

ZABBIX SIA
Dzelzavas Street 117
Office #417
LV-1021
Riga, Latvia

電話: +371 6 7784742
Fax: +371 6 7784744

メール: sales@zabbix.com

ウェブ: www.zabbix.com

Copyright © 2008 by
ZABBIX SIA.

ZABBIX は ZABBIX SIA の
登録商標です。その他の全
ての名前および製品は、その
所有者の商標または登録商
標です。