

Я активно использую репликацию mysql в своей работе, причем как master-slave, так и master-master. Иногда возникают ошибки, и если за ними не следить, то в определенный момент станет очень грустно, когда окажется, что резервный сервер не имеет актуальную версию базы. Чтобы не грустить по этому поводу, мы настроим мониторинг MySQL репликации с помощью Zabbix.

Если у вас есть желание научиться строить и поддерживать высокодоступные и надежные системы, рекомендую познакомиться с **онлайн-курсом «Администратор Linux»** в OTUS. Курс не для новичков, для поступления нужно пройти .

Содержание:

- 1 Введение
- 2 Добавление пользователя mysql
- 3 Создание скрипта для мониторинга mysql репликации
- 4 Настройка zabbix agent
- 5 Настройка мониторинга репликации mysql на zabbix server
- 6 Проверка работы триггеров
- 7 Заключение
- 8 Дополнительные материалы по Zabbix

Введение

Если у вас еще нет своего сервера для мониторинга, то рекомендую материалы на эту тему. Для тех, кто предпочитает систему CentOS:

1. Установка CentOS 8.
2. Настройка CentOS 8.

3. Установка и настройка zabbix сервера.

То же самое на Debian 10, если предпочитаете его:


1. Установка Debian 10.
2. Базовая настройка Debian.
3. Установка и настройка zabbix на debian.

Несколько слов о том, что именно мы будем мониторить. Посмотреть статус репликации можно с помощью простой команды в консоли mysql:

```
MariaDB [(none)]> show slave status\G;
```

```
MariaDB [(none)]> show slave status\G;
***** 1. row *****
      Slave_IO_State: Waiting for master to send event
      Master_Host: 10.8.0.10
      Master_User: root
      Master_Port: 3306
      Connect_Retry: 60
      Master_Log_File: mysql-bin.000012
      Read_Master_Log_Pos: 245
      Relay_Log_File: mariadb-relay-bin.000008
      Relay_Log_Pos: 529
      Relay_Master_Log_File: mysql-bin.000012
      Slave_IO_Running: Yes
      Slave_SQL_Running: Yes
      Replicate_Do_DB:
      Replicate_Ignore_DB:
      Replicate_Do_Table:
      Replicate_Ignore_Table:
      Replicate_Wild_Do_Table:
      Replicate_Wild_Ignore_Table:
      Last_Errno: 0
      Last_Error:
      Skip_Counter: 0
      Exec_Master_Log_Pos: 245
      Relay_Log_Space: 1109
      Until_Condition: None
      Until_Log_File:
      Until_Log_Pos: 0
      Master_SSL_Allowed: No
      Master_SSL_CA_File:
      Master_SSL_CA_Path:
      Master_SSL_Cert:
      Master_SSL_Cipher:
      Master_SSL_Key:
      Seconds_Behind_Master: 0
      Master_SSL_Verify_Server_Cert: No
      Last_IO_Errno: 0
      Last_IO_Error:
      Last_SQL_Errno: 0
      Last_SQL_Error:
      Replicate_Ignore_Server_Ids:
      Master_Server_Id: 2
1 row in set (0.00 sec)

ERROR: No query specified

MariaDB [(none)]> 
```

serveradmin.ru

Нас будут интересовать три параметра:

- **Seconds_Behind_Master** — то, насколько слейв сервер отстает от мастера в репликации.
- **Slave_IO_Running** — индикатор работы демона по сбору бинарного лога с мастера и записи его в локальный relay лог.
- **Slave_SQL_Running** — индикатор выполнения команд из локального relay лога.

Первый параметр в идеале должен равняться нулю, то есть slave сервере следует за мастером непрерывно. Если значение начинает увеличиваться, срабатывает триггер и оповещает о том, что реплика начинает отставать. Два других параметра должны выдавать значение **Yes**. Если это не так, то тоже срабатывает триггер и шлет оповещение о том, что репликация не работает.

Реализовывать будем так же как и в случае с мониторингом nginx и php-fpm через скрипт и UserParameter. Если у вас репликация master-master, то настраиваете мониторинг на обоих серверах.

Я буду настраивать мониторинг на сервере CentOS 7, но в данном случае это не имеет принципиального значения. Настройки будут идентичны практически на любом linux дистрибутиве.

Добавление пользователя mysql

Прежде чем начать настраивать сам мониторинг, выполним необходимые предварительные действия.

Создаем в mysql юзера, у которого будет доступ к информации о репликации:

```
# mysql -uroot -ppassword
```

```
MariaDB [(none)]> CREATE USER 'zabbix'@'localhost' IDENTIFIED BY 'parol';  
MariaDB [(none)]> GRANT REPLICATION CLIENT ON *.* TO 'zabbix'@'localhost' IDENTIFIED BY 'parol';
```

Создали учетную запись zabbix с паролем *parol* и правами replication client. Её мы будем использовать в дальнейшей работе.

Создание скрипта для мониторинга mysql репликации

Теперь создадим скрипт для мониторинга. Скачиваем его отсюда [mysql-slave.sh](#) и вставляем содержимое в папку `/etc/zabbix/scripts`:

```
# mcedit /etc/zabbix/scripts/mysql-slave.sh
# chown zabbix:zabbix /etc/zabbix/scripts/mysql-slave.sh
# chmod 550 /etc/zabbix/scripts/mysql-slave.sh
```

Проверяем его работу:

```
/etc/zabbix/scripts/mysql-slave.sh Master_Host zabbix parol
```

Скрипт должен вернуть в консоль имя мастера.

Если вы будете проверять работу скрипта от пользователя `root`, обязательно удалите временный файл из папки `/tmp`, который создает скрипт. Если этого не сделать, то потом заббикс не сможет его прочитать, так как ему не хватит для этого прав.

Данный скрипт анализирует вывод `show slave status\G` и парсит 3 необходимых нам значения. Он передает агенту информацию о задержке репликации через параметр `Seconds_Behind_Master` и анализирует значения `Slave_IO_Running` и `Slave_SQL_Running`. Если их значения равны `Yes`, он передает агенту `1`, если там что-то другое то `0`.

Настройка zabbix agent

Добавляем новый параметр в `zabbix_agentd.conf`

```
# mcedit /etc/zabbix/zabbix_agentd.conf  
UserParameter=mysql-slave[*],/etc/zabbix/scripts/mysql-slave.sh "$1" zabbix parol
```

Сохраняем конфиг и перезапускаем агента:

```
# systemctl restart zabbix-agent
```

Настройка мониторинга репликации mysql на zabbix server

Здесь все как обычно. Скачиваем шаблон [mysql-slave.xml](#) импортируем его на сервер. Для этого идем в **Configuration -> Templates** и нажимаем **Import**:

Monitoring | Inventory | Reports | **Configuration** | Administration

Host groups | **Templates** | Hosts | Maintenance | Actions | Screens | Slide shows | Maps | Discovery | IT services

History: Configuration of templates » Dashboard » Host screens » Configuration of hosts » Dashboard

CONFIGURATION OF TEMPLATES Create template **Import**

Group all ▼

Templates
Displaying 1 to 41 of 41 found

<input type="checkbox"/>	Templates ↑	Applications	Items	Triggers	Graphs	Screens	Discovery	Web	Linked templates	Linked to
<input type="checkbox"/>	Template App FTP Service	Applications (1)	Items (1)	Triggers (1)	Graphs (0)	Screens (0)	Discovery (0)	Web (0)	-	-
<input type="checkbox"/>	Template App HTTP Service	Applications (1)	Items (1)	Triggers (1)	Graphs (0)	Screens (0)	Discovery (0)	Web (0)	-	-
<input type="checkbox"/>	Template App HTTPS Service	Applications (1)	Items (1)	Triggers (1)	Graphs (0)	Screens (0)	Discovery (0)	Web (0)	-	-
<input type="checkbox"/>	Template App IMAP Service	Applications (1)	Items (1)	Triggers (1)	Graphs (0)	Screens (0)	Discovery (0)	Web (0)	-	-

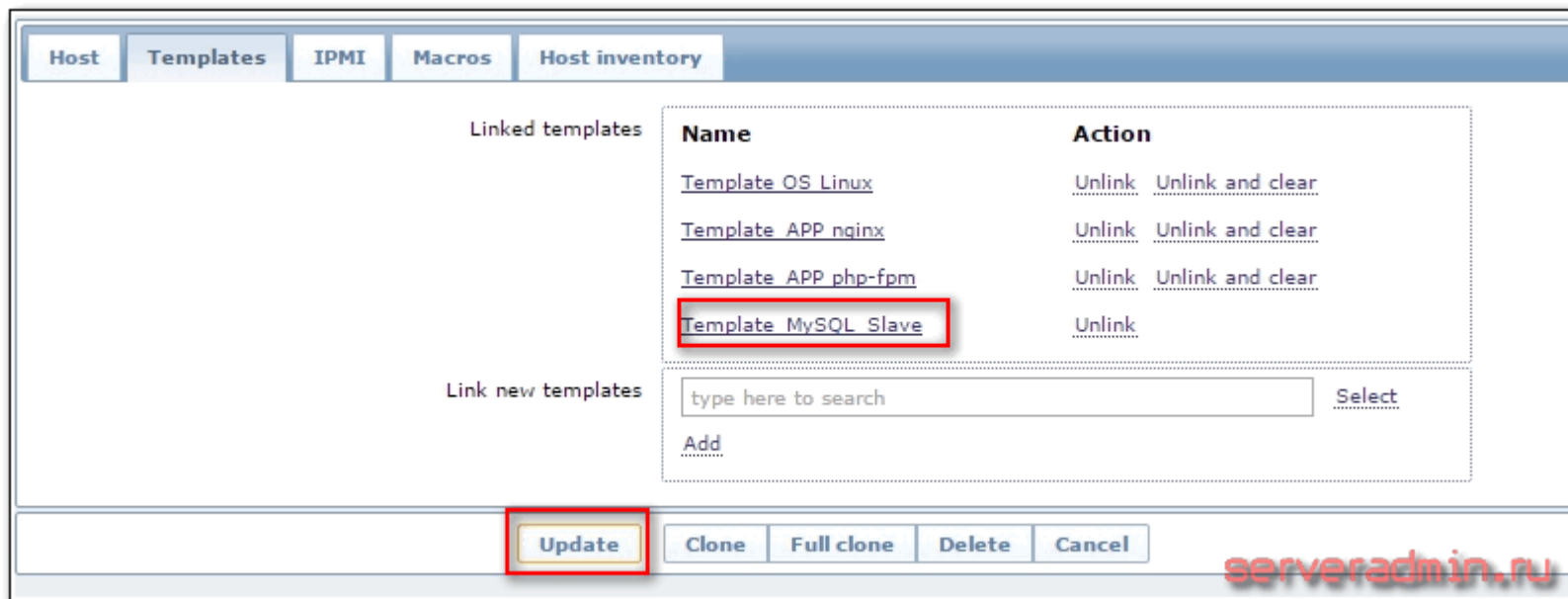
serveradmin.ru

Выбираем скачанный шаблон и жмем [Import](#):

The screenshot shows the Zabbix Configuration interface. The top navigation bar includes 'Monitoring', 'Inventory', 'Reports', 'Configuration', and 'Administration'. Below this, there are sub-menus for 'Host groups', 'Templates', 'Hosts', 'Maintenance', 'Actions', 'Screens', 'Slide shows', 'Maps', 'Discovery', and 'IT services'. A breadcrumb trail reads: 'History: Dashboard » Host screens » Configuration of hosts » Dashboard » Configuration of templates'. The main content area is titled 'Import' and features an 'Import file' field with the text 'Выберите файл' and 'mysql-slave.xml'. Below this is a table of rules with columns for 'Update existing', 'Create new', and 'Delete missing'. The 'Import' button at the bottom left is highlighted with a red box.

Rules	Update existing	Create new	Delete missing
Groups		<input checked="" type="checkbox"/>	
Hosts	<input type="checkbox"/>	<input type="checkbox"/>	
Templates	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Template screens	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Template linkage		<input checked="" type="checkbox"/>	
Applications		<input checked="" type="checkbox"/>	<input type="checkbox"/>
Items	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Discovery rules	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Triggers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Graphs	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Screens	<input type="checkbox"/>	<input type="checkbox"/>	
Maps	<input type="checkbox"/>	<input type="checkbox"/>	
Images	<input type="checkbox"/>	<input type="checkbox"/>	

Дальше отправляемся к списку хостов в **Configuration -> Hosts**, выбираем нужный хост и назначаем ему новый шаблон:



Жмем **Update** для применения настроек. Ждем несколько минут и идем проверять поступление новых данных репликации mysql. Открываем **Monitoring -> Latest Data**, настраиваем фильтр и проверяем значения:

The screenshot shows the Zabbix web interface with the 'Latest data' page selected. The navigation menu includes Monitoring, Inventory, Reports, Configuration, and Administration. The breadcrumb trail is: Configuration of templates » Configuration import » Configuration of templates » Configuration of hosts » Latest data. The 'LATEST DATA' section is active, showing a table of items for MySQL replication. The table has columns for Name, Last check, Last value, and Change. Three items are listed: 'MySQL: Seconds Behind Master' with a last value of 0, 'MySQL: Slave IO Running' with a last value of 1, and 'MySQL: Slave SQL Running' with a last value of 1. The 'Last value' column is highlighted with a red box. The footer of the interface shows 'Zabbix 2.4.6 Copyright 2001-2015 by Zabbix SIA' and the 'serveradmin.ru' logo.

Name	Last check	Last value	Change
MySQL (3 Items)			
MySQL: Seconds Behind Master	2015-11-26 22:56:58	0	-
MySQL: Slave IO Running	2015-11-26 22:56:59	1	-
MySQL: Slave SQL Running	2015-11-26 22:57:00	1	-

В данном случае мы видим, что значение Seconds Behind Master = 0, отставания от мастера нет. Два других значения равны единице, это значит, что наш скрипт проверки состояния репликации получает статусы Slave_IO_Running и Slave_SQL_Running равные Yes и поэтому возвращает значения 1. То есть наша репликация работает в штатном режиме, все в порядке.

Проверка работы триггеров

Попробуем нарушить работу репликации mysql и проверим работу триггеров. Для этого я просто отключу vpn соединение, по которому доступны сервера. После разрыва связи на slave сервере следующая картинка статуса репликации:


```
MariaDB [(none)]> show slave status\G;
***** 1. row *****
      Slave_IO_State: Reconnecting after a failed master event read
        Master_Host: 10.8.0.10
         Master_User: root
         Master_Port: 3306
        Connect_Retry: 60
        Master_Log_File: mysql-bin.000012
    Read_Master_Log_Pos: 245
         Relay_Log_File: mariadb-relay-bin.000008
         Relay_Log_Pos: 529
    Relay_Master_Log_File: mysql-bin.000012
      Slave_IO_Running: Connecting
      Slave_SQL_Running: Yes
    Replicate_Do_DB:
    Replicate_Ignore_DB:
    Replicate_Do_Table:
    Replicate_Ignore_Table:
    Replicate_Wild_Do_Table:
    Replicate_Wild_Ignore_Table:
          Last_Errno: 0
          Last_Error:
         Skip_Counter: 0
    Exec_Master_Log_Pos: 245
         Relay_Log_Space: 1109
    Until_Condition: None
         Until_Log_File:
         Until_Log_Pos: 0
    Master_SSL_Allowed: No
    Master_SSL_CA_File:
    Master_SSL_CA_Path:
         Master_SSL_Cert:
         Master_SSL_Cipher:
         Master_SSL_Key:
    Seconds_Behind_Master: NULL
Master_SSL_Verify_Server_Cert: No
          Last_IO_Errno: 2003
          Last_IO_Error: error reconnecting to master 'root@10.8.0.10:3306'
          Last_SQL_Errno: 0
          Last_SQL_Error:
    Replicate_Ignore_Server_Ids:
         Master_Server_Id: 2
1 row in set (0.01 sec)

ERROR: No query specified
```

serveradmin.ru

Проверяем данные мониторинга репликации:

The screenshot shows the Zabbix web interface for monitoring MySQL replication. The 'Items' section is active, displaying a table of monitoring items. The table has columns for Name, Last check, Last value, and Change. The 'MySQL: Slave IO Running' item is highlighted with a red box, showing a last value of 0 and a change of -1. The interface also includes navigation tabs, a search bar, and filter options.

Name	Last check	Last value	Change
MySQL (3 Items)			
MySQL: Seconds Behind Master	2015-11-26 23:05:58	0	-
MySQL: Slave IO Running	2015-11-26 23:06:59	0	-1
MySQL: Slave SQL Running	2015-11-26 23:07:00	1	-

Значение Slave_IO_Running сменилось с Yes на Connecting и скрипт проверки вернул значение 0 вместо 1. Этого достаточно, чтобы сработал триггер и пришло оповещение о том, что репликация mysql сервера нарушена:



The screenshot shows a Zabbix interface window titled "Last 20 issues". It contains a table with the following columns: Host, Issue, Last change, Age, Info, Ack, and Actions. The first row is highlighted in red and shows the issue "MySQL: Replication not running" with a last change time of "2015-11-26 23:06:59" and an age of "2m 36s". The Ack column for this row is "No" and the Actions column shows "1". Below the table, a watermark "serveradmin.ru" and the text "6 of 6 issues are shown" are visible.

Host	Issue	Last change	Age	Info	Ack	Actions
	MySQL: Replication not running	2015-11-26 23:06:59	2m 36s		No	1
		2015-11-26 13:34:50	9h 34m 45s		No	1
		2015-11-25 02:25:04	1d 20h 44m		No	1
		2015-11-24 18:40:03	2d 4h 29m		No	1
		2015-11-20 19:37:47	6d 3h 31m		No	1
		2015-10-31 22:50:46	26d 18m		No	

На почту пришло оповещение:

PROBLEM: MySQL: Replication not running zabbix zeroxed x

Zabbix Zeroxed Trigger: MySQL: Replication not running Trigger status: PROBLEM Trigger sever...

Zabbix Zeroxed <zabbix.zeroxed@gmail.com>
кому: мне ▾

Trigger: MySQL: Replication not running
Trigger status: PROBLEM
Trigger severity: Disaster
Trigger URL:

Item values:

1. MySQL: Slave IO Running ([redacted] mysql-slave[Slave_IO_Running]): 0
2. MySQL: Slave SQL Running ([redacted] mysql-slave[Slave_SQL_Running]): 1
3. *UNKNOWN* (*UNKNOWN*: *UNKNOWN*): *UNKNOWN*


Original event ID: 362


serveradmin.ru

По-умолчанию мониторинг не умеет отправлять оповещения на сторонние серверы с авторизацией по smtp. Для настройки отправки оповещений на email можно воспользоваться моей подробной инструкцией на эту тему.

Восстанавливаем связь между серверами и ждем новой работы триггера и уведомления:

OK: MySQL: Replication not running zabbix zeroxzed x

 **Zabbix Zeroxzed** Trigger: MySQL: Replication not running Trigger status: OK Trigger severity: ...

 **Zabbix Zeroxzed** <zabbix.zeroxzed@gmail.com>
кому: мне ▾

Trigger: MySQL: Replication not running
Trigger status: OK
Trigger severity: Disaster
Trigger URL:

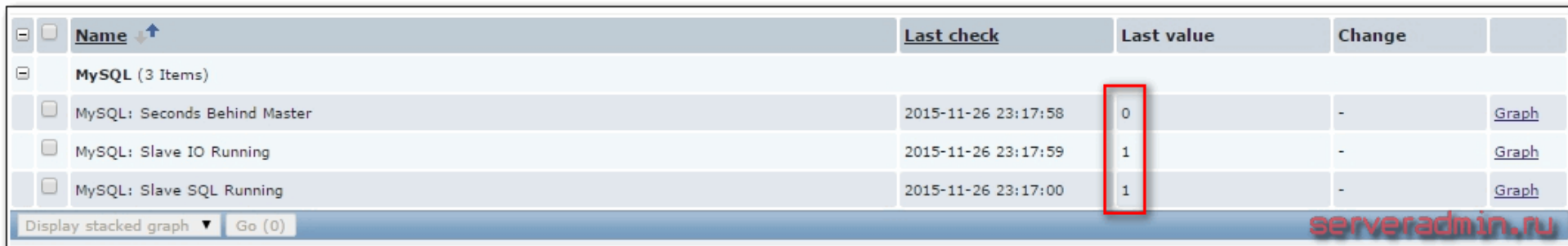
Item values:

1. MySQL: Slave IO Running ([REDACTED] mysql-slave[Slave_IO_Running]): 1
2. MySQL: Slave SQL Running ([REDACTED] mysql-slave[Slave_SQL_Running]): 1
3. *UNKNOWN* (*UNKNOWN*: *UNKNOWN*): *UNKNOWN*

Original event ID: 362

serveradmin.ru

Проверяем Latest Data:



Name	Last check	Last value	Change	
MySQL (3 Items)				
MySQL: Seconds Behind Master	2015-11-26 23:17:58	0	-	Graph
MySQL: Slave IO Running	2015-11-26 23:17:59	1	-	Graph
MySQL: Slave SQL Running	2015-11-26 23:17:00	1	-	Graph

Display stacked graph ▼ Go (0) serveradmin.ru

Все в порядке, мониторинг нормально отработал нарушение mysql репликации. Больше тут настраивать нечего, графики и экраны не нужны, в них нет необходимости. На этом работа по настройке мониторинга окончена.

Заключение

Для мониторинга состояния репликации mysql мы воспользовались самописным скриптом, который парсит вывод значения `show slave status\G` и анализирует необходимые нам параметры реплики. На основе этих параметров он передает zabbix агенту необходимые значения для отправки на сервер.

На сервер мониторинга мы импортировали готовый шаблон для сбора данных и работы триггеров. Подключили этот шаблон к хосту и посмотрели значения, которые он получает. Так же проверили срабатывание триггеров, принудительно нарушив работу репликации. Таким образом убедились, что мониторинг корректно обрабатывает нештатные ситуации.

Помогла статья? Подписывайся на telegram канал автора

Анонсы всех статей, плюс много другой полезной и интересной информации, которая не попадает на сайт.

Дополнительные материалы по Zabbix

Онлайн курс по Linux

Если вы хотите стать специалистом по отказоустойчивым виртуальным и кластерным средам, рекомендую познакомиться с онлайн-курсом **Администратор Linux. Виртуализация и кластеризация** в OTUS. Курс не для новичков, для поступления нужны хорошие знания по Linux. Обучение длится 5 месяцев, после чего успешные выпускники курса смогут пройти собеседования у партнеров. Что даст вам этот курс:

- Умение строить отказоустойчивые кластера виртуализации для запуска современных сервисов, рассчитанных под высокую нагрузку.
- Будете разбираться в современных технологиях кластеризации, оркестрации и виртуализации.
- Научитесь выбирать технологии для построения отказоустойчивых систем под высокую нагрузку.
- Практические навыки внедрения виртуализации KVM, oVirt, Xen.
- Кластеризация сервисов на базе rasemaker, k8s, nomad и построение дисковых кластеров на базе ceph, gluster, linstore.

Проверьте себя на вступительном тесте и смотрите подробнее программу по .

Рекомендую полезные материалы по Zabbix:

Настройки системы

- Установка 4.0
- Обновление 3.0 -> 3.2
- Обновление 3.4 -> 4.0
- Установка Zabbix Proxy
- Работа на NGINX

Видео и подробное описание установки и настройки Zabbix 4.0, а также установка агентов на linux и windows и подключение их к мониторингу.

Подробное описание обновления системы мониторинга zabbix версии 3.4 до новой версии 4.0.

Пошаговая процедура обновления сервера мониторинга zabbix 2.4 до 3.0. Подробное описание каждого шага с пояснениями и рекомендациями.

Подробное описание установки и настройки zabbix proxy для организации распределенной системы мониторинга. Все показано на примерах.

Подробное описание установки системы мониторинга Zabbix на веб сервер на базе nginx + php-fpm.

Мониторинг служб и сервисов

- Температура процессора
- Nginx и php-fpm
- MySQL репликация
- Службы Linux
- Рейд mdadm
- Транки Asterisk
- Synology

Мониторинг температуры процессора с помощью zabbix на Windows сервере с использованием пользовательских скриптов.

Настройка полноценного мониторинга web сервера nginx и php-fpm в zabbix с помощью скриптов и пользовательских параметров.

Мониторинг репликации mysql с помощью Zabbix. Подробный разбор методики и тестирование работы.

Описание настройки мониторинга tcp служб с помощью zabbix и его инструмента простых проверок (simple checks)

Настройка мониторинга рейда mdadm с помощью zabbix. Подробное пояснение принципа работы и пошаговая инструкция.

Подробное описание мониторинга регистраций транков (trunk) в asterisk с помощью сервера мониторинга zabbix.

Подробная инструкция со скриншотами по настройке мониторинга по snmp дискового хранилища synology с помощью сервера мониторинга zabbix.

Мониторинг различных значений

- Мониторинг сайта
- Мониторинг бэкапов
- Размер бэкапа
- Делегирование домена
- Значения из текстового файла
- Мониторинг логов

Настройка мониторинга web сайта в zabbix. Параметры для наблюдения - доступность сайта, время отклика, скорость доступа к сайту.

Один из способов мониторинга бэкапов с помощью zabbix через проверку даты последнего изменения файла из архивной копии с помощью vfs.file.time.

Подробное описание настройки мониторинга размера бэкапов в Zabbix с помощью внешних скриптов.

Пример настройки мониторинга за временем делегирования домена с помощью Zabbix и внешнего скрипта. Все скрипты и готовый шаблон представлены.

Пример распознавания и мониторинга за изменением значений в обычных текстовых файлах с помощью zabbix.

Описание мониторинга лог файлов в zabbix на примере анализа лога программы arcpusd. Отправка оповещений по событиям из лога.