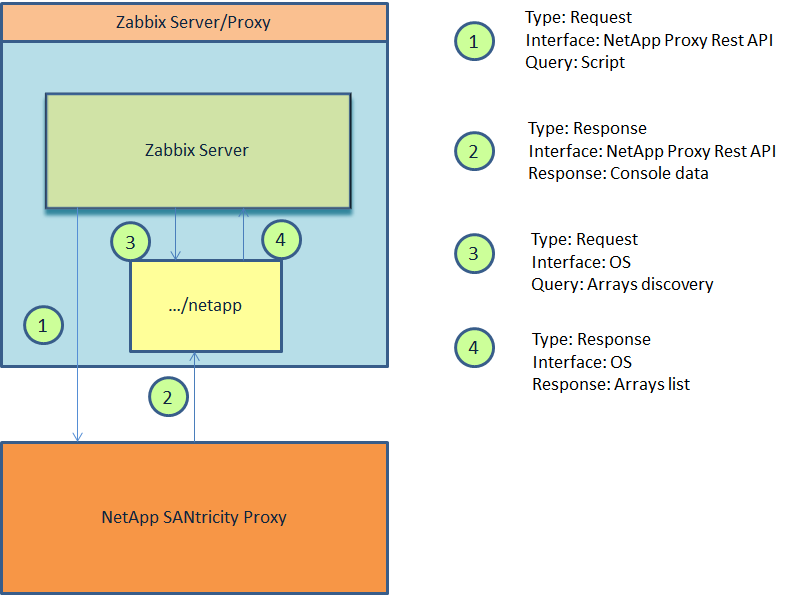
## NetApp E-Series

### Метод сбора информации от объекта мониторинга



Для сбора необходимых метрик и статусов система мониторинга использует скрипт netapp.py с различными параметрами, подключаемый через механизм Rest API к виртуальной машине NetApp SANtricity Proxy. Информация, полученная от NetApp SANtricity Proxy, записывается во временные файлы в директорию /usr/local/share/zabbix/externalscripts/netapp, которая должна быть **предварительно создана** на сервере мониторинга. Параметры подключения к NetApp SANtricity Proxy задаются в фале скрипта в разделе “Credentials”. Например:

**# Credentials**

**hostIP = '10.46.64.14'**

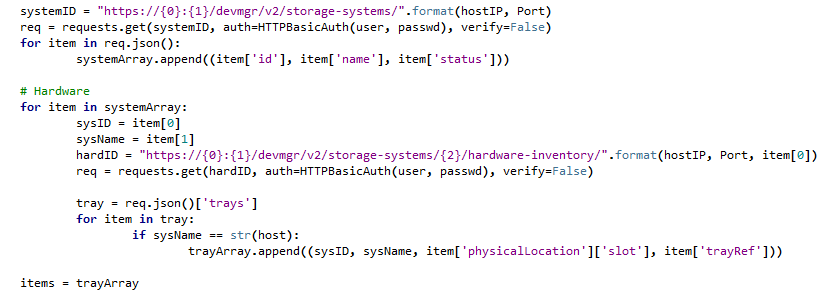
**Port = '8444'**

**user = 'ro'**

**passwd = 'ro'**

### Конфигурирование объекта мониторинга

В шаблоне системы мониторинга задаются правила обнаружения, соответствующие различным комплектующим устройств NetApp. При этом параметры запуска скрипта netapp.py зависят от соответствующего правила обнаружения. Запрашиваемая информация записывается во временные файлы с именем вида <Имя хоста>\_<метрика> в формате JSON. Далее система мониторинга использует информацию из временных локальных файлов, автоматически создавая объекты мониторинга на основе заранее подготовленных прототипов данных, либо для вручную созданных объектов.

Пример получения данных по дисковым полкам:

!

**Важно:** Параметр **Host name** в Zabbix должен **совпадать** со значением переменной sysName из примера выше (заданное имя реального, но не proxy, устройства NetApp). В противном случае данные, получаемые на этапе обнаружения (Discovery), не будут корректно сопоставлены items, создаваемым в Zabbix.

!

**Важно:** В целях снижения нагрузки на NetApp SANtricity Proxy, данные для всех items одного типа запрашиваются одновременно, на этапе обнаружения (Discovery rules в шаблоне Zabbix). Обновление состояний items производится из временных файлов, в которые были записаны ранее полученные данные. Соответственно, реальный временной интервал обновления данных items – временной интервал, заданный для выполнения Discovery rules. Это необходимо учитывать при настройке интервалов мониторинга объекта.

!

**Важно:** Во избежание проблем с правами доступа, владельцем директории "/usr/local/share/zabbix/externalscripts/netapp" и файлов внутри должен быть пользователь zabbix: *chown -r zabbix:zabbix /usr/local/share/zabbix/externalscripts/netapp*



### Шаблоны

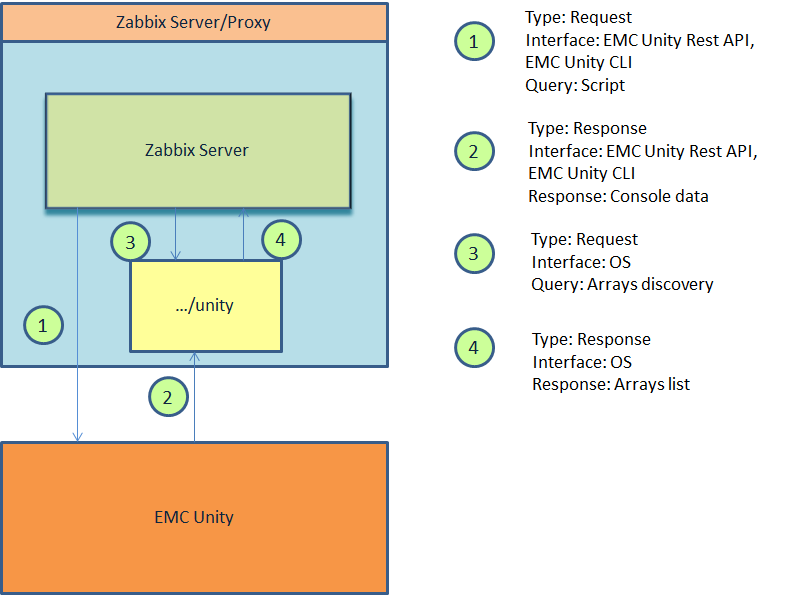
Для описанных процедур в системе мониторинга используется шаблон [Template NetApp E-Series](http://10.45.35.206/templates.php?form=update&templateid=10180&groupid=32).

### Описание необходимых настроек на сервере Zabbix

Для настройки шаблонов и объектов на сервере Zabbix необходимо выполнить процедуры, аналогичные описанным в п.3 настоящего документа.

## EMC Unity

### Метод сбора информации от объекта мониторинга



Для сбора необходимых метрик и статусов система мониторинга использует скрипт emc\_unity.py с различными параметрами, подключаемый через механизм Rest API к устройствам EMC Unity. Скрипт emc\_unity.py для своей корректной работы использует также скрипты Unity.py и UnityClasses.py, которые должны находиться с ним в одной директории. Скрипты Unity.py и UnityClasses.py не подлежат редактированию. Для отдельно взятого небольшого числа метрик, которые невозможно получить с помощью механизма Rest API, используется также обращение к устройствам через механизм CLI. Для вызова команд CLI на сервере мониторинга должен быть предварительно установлен пакет команд **uemcli**, доступный на официальном сайте EMC. Для получения необходимого дистрибутива **uemcli**  рекомендуется обратиться к администратору EMC Unity. Стоит заметить, что поскольку вызов команд CLI для EMC Unity не поддерживается из среды Zabbix, вызов осуществляется с использованием планировщика cron. Информация, полученная от устройств EMC Unity, записывается во временные файлы в директорию /usr/local/share/zabbix/externalscripts/unity, которая должна быть **предварительно создана** на сервере мониторинга. Параметры подключения к EMC Unity задаются в фале скрипта в разделе “Credentials”. Например:

**# Credentials**

**u\_credentials = {**

**"SM22-TEST-0616": {'User': 'zabbix',**

**'Password': '4Monit0r!',**

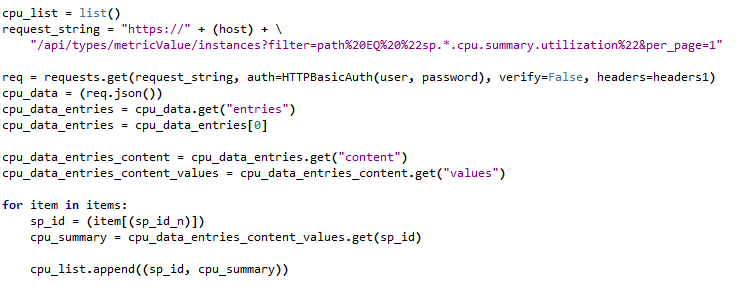
**'Host': '10.45.36.109',**

**}**

**}**

### Конфигурирование объекта мониторинга

В шаблоне системы мониторинга задаются правила обнаружения, соответствующие различным комплектующим устройств EMC Unity. При этом параметры запуска скрипта emc\_unity.py зависят от соответствующего правила обнаружения. Запрашиваемая информация записывается во временные файлы с именем вида <Имя хоста>\_<метрика> в формате JSON. Далее система мониторинга использует информацию из временных локальных файлов, автоматически создавая объекты мониторинга на основе заранее подготовленных прототипов данных, либо для вручную созданных объектов. Пример получения данных по утилизации CPU:



С целью сбора части метрик для LUNs, которые невозможно получить с помощью вызовов Rest API, необходимо использовать функции CLI и вызов их из стандартного планировщика сервера, где установлена система мониторинга, вместо правил обнаружения Zabbix. В частности, для Linux-систем, необходимо в файл **cron** добавить строку вида:

**\*/10 \* \* \* \* /usr/local/share/zabbix/externalscripts/emc\_unity.py --type lun --item LUNmetrics**

с указанием соответствующих интервалов запуска скрипта emc\_unity.py.

!

**Важно:** Параметр **Host name** в режиме конфигурирования хоста в Zabbix должен **совпадать** со значением имени хоста, передаваемого из скрипта. В противном случае данные, получаемые на этапе обнаружения (Discovery), не будут корректно сопоставлены items, создаваемым в Zabbix:



!

**Важно:** В целях снижения нагрузки на EMC Unity, данные для всех items одного типа запрашиваются одновременно, на этапе обнаружения (Discovery rules в шаблоне Zabbix). Обновление состояний items производится из временных файлов, в которые были записаны ранее полученные данные. Соответственно, реальный временной интервал обновления данных items – временной интервал, заданный для выполнения Discovery rules. Это необходимо учитывать при настройке интервалов мониторинга объекта.

!

**Важно:** Во избежание проблем с правами доступа, владельцем директории "/usr/local/share/zabbix/externalscripts/unity" и файлов внутри должен быть пользователь zabbix: *chown -r zabbix:zabbix /usr/local/share/zabbix/externalscripts/unity*

### Шаблоны

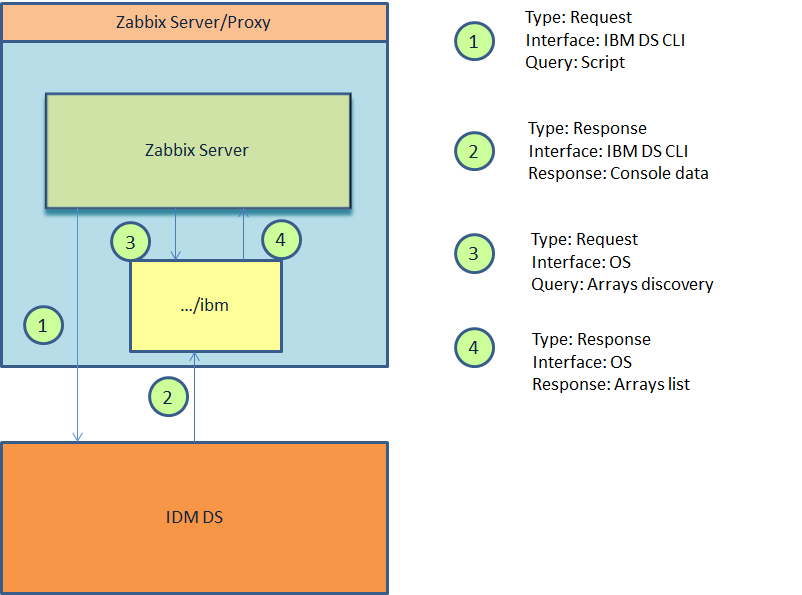
Для описанных процедур в системе мониторинга используется шаблон Template EMC Unity v01.

### Описание необходимых настроек на сервере Zabbix

Для настройки шаблонов и объектов на сервере Zabbix необходимо выполнить процедуры, аналогичные описанным в п.3 настоящего документа.

## IBM DS

### Метод сбора информации от объекта мониторинга



Для сбора необходимых метрик и статусов система мониторинга использует скрипт ibm\_ds8000.py с различными параметрами, подключаемый через механизм CLI к устройствам IBM DS. В своей работе скрипт использует команды **dscli**, соответствующий пакет команд должен быть предварительно установлен на сервере мониторинга. Пакет для различных операционных систем доступен на официальном сайте IBM и требует предварительной регистрации. Для получения необходимого дистрибутива **dscli** рекомендуется обратиться к администратору IBM DS. Информация, полученная от IBM DS, записывается во временные файлы в директорию /usr/local/share/zabbix/externalscripts/ibm, которая должна быть **предварительно создана** на сервере мониторинга. На этапе дискаверинга скрипт формирует в указанной директории файлы вида <Имя\_хоста>\_script\_fc и <Имя\_хоста>\_script\_luns, которые используются в дальнейшем в качестве входных данных для сбора различных параметров по FC-портам и LUNs. Содержимое данных файлов зависит от текущего набора FC-портов и LUNs. Параметры подключения к IBM DS задаются в фале скрипта в разделе “Credentials”. Например:

**# Credentials**

**u\_credentials = {**

**"SM22-DS8870": {'User': 'zabbix',**

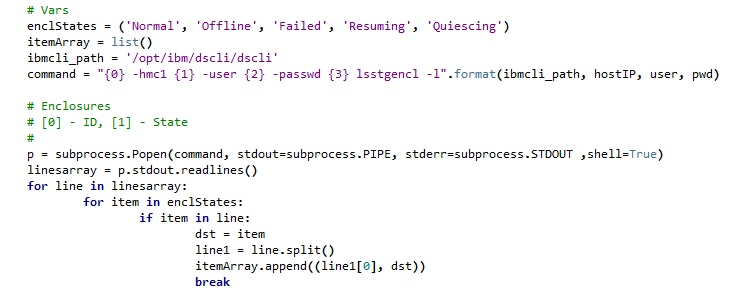
**'Password': '4Monit0r!',**

**'Host': '10.45.98.136',**

**}}**

### Конфигурирование объекта мониторинга

В шаблоне системы мониторинга задаются правила обнаружения, соответствующие различным комплектующим устройств IBM DS. При этом параметры запуска скрипта ibm\_ds8000.py зависят от соответствующего правила обнаружения. Запрашиваемая информация записывается во временные файлы с именем вида <Имя хоста>\_<метрика> в формате JSON. Далее система мониторинга использует информацию из временных локальных файлов, автоматически создавая объекты мониторинга на основе заранее подготовленных прототипов данных, либо для вручную созданных объектов. Пример получения данных по дисковым полкам:



Следует заметить, что в связи с длительным временем выполнения команд **dscli** и, как следствие, скриптов, содержащих эти команды, часть запросов необходимо выполнять из стандартного планировщика сервера, где установлена система мониторинга, вместо правил обнаружения Zabbix. В частности, для Linux-систем, необходимо в файл **cron** добавить строки вида:

**\*/10 \* \* \* \* /usr/local/share/zabbix/externalscripts/ibm\_ds8000.py --type lun --item LUNmetrics**

**\*/10 \* \* \* \* /usr/local/share/zabbix/externalscripts/ibm\_ds8000.py --type fcp --item FCPmetrics**

**\*/10 \* \* \* \* /usr/local/share/zabbix/externalscripts/ibm\_ds8000.py --type system --item SYSmetrics**

с указанием соответствующих временных интервалов запуска скрипта ibm\_ds8000.py. В приведённом примере скрипт запускается каждые 10 минут.

!

**Важно:** Параметр **Host name** в режиме конфигурирования хоста в Zabbix должен **совпадать** со значением имени хоста, передаваемого из скрипта. В противном случае данные, получаемые на этапе обнаружения (Discovery), не будут корректно сопоставлены items, создаваемым в Zabbix.

!

**Важно:** В целях снижения нагрузки на IBM DS, данные для всех items одного типа запрашиваются одновременно, на этапе обнаружения (Discovery rules в шаблоне Zabbix). Обновление состояний items производится из временных файлов, в которые были записаны ранее полученные данные. Соответственно, реальный временной интервал обновления данных items – временной интервал, заданный для выполнения Discovery rules. Это необходимо учитывать при настройке интервалов мониторинга объекта.

!

**Важно:** Во избежание проблем с правами доступа, владельцем директории "/usr/local/share/zabbix/externalscripts/ibm" и файлов внутри должен быть пользователь zabbix: *chown -r zabbix:zabbix /usr/local/share/zabbix/externalscripts/ibm*

### Шаблоны

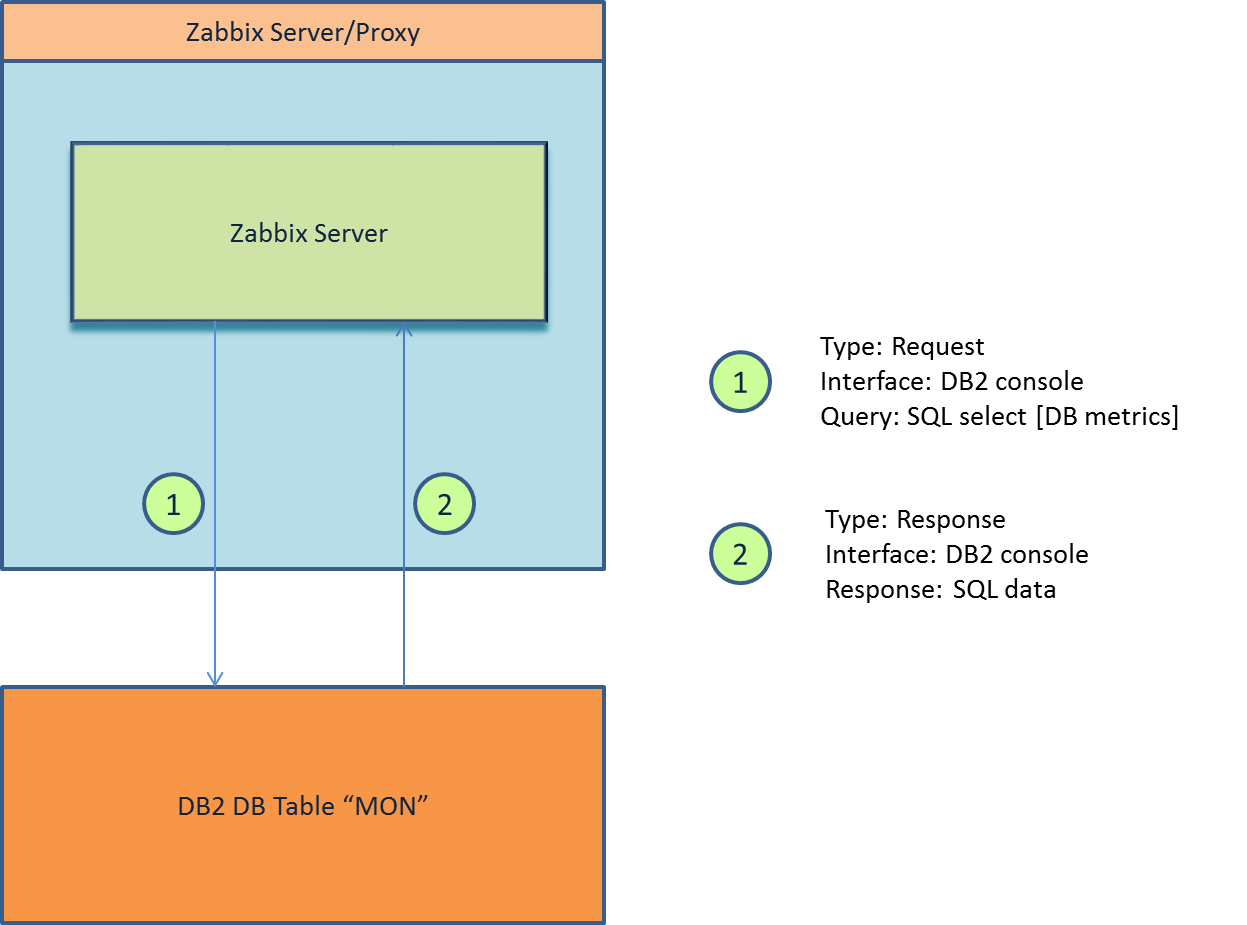
Для описанных процедур в системе мониторинга используется шаблон [Template IBM DS](http://10.45.35.206/templates.php?form=update&templateid=10175).

### Описание необходимых настроек на сервере Zabbix

Для настройки шаблонов и объектов на сервере Zabbix необходимо выполнить процедуры, аналогичные описанным в п.3 настоящего документа.

## IBM DB2

### Метод сбора информации от объекта мониторинга

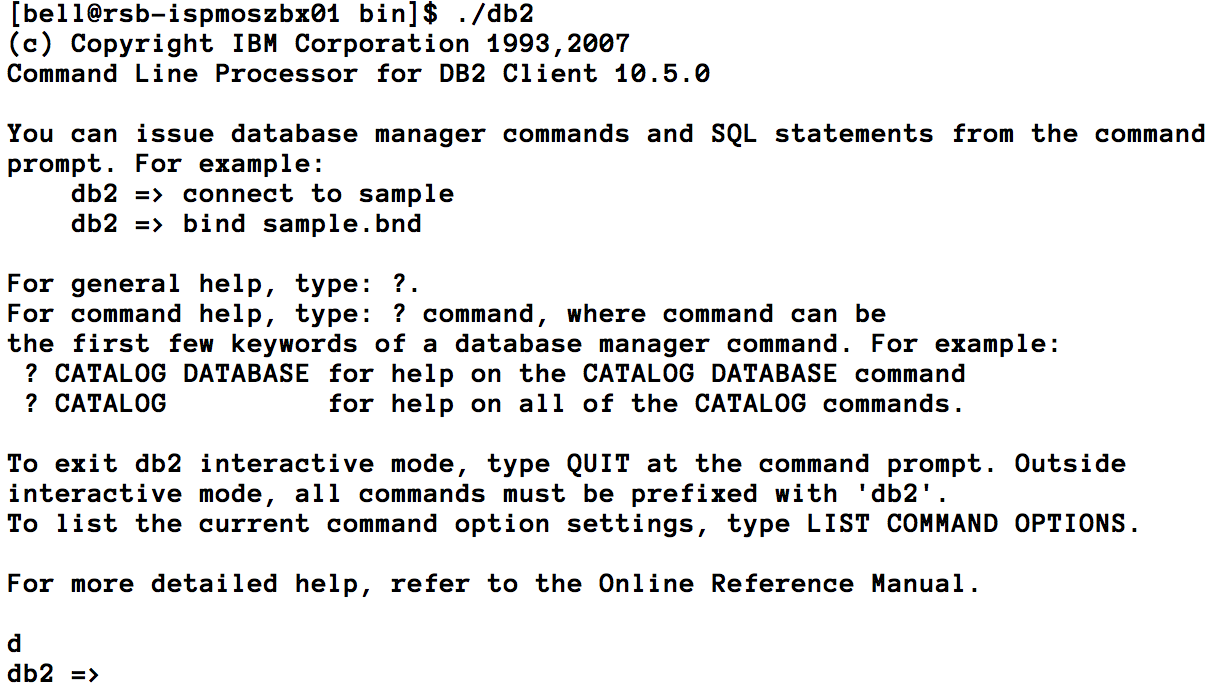


### Конфигурирование объекта мониторинга

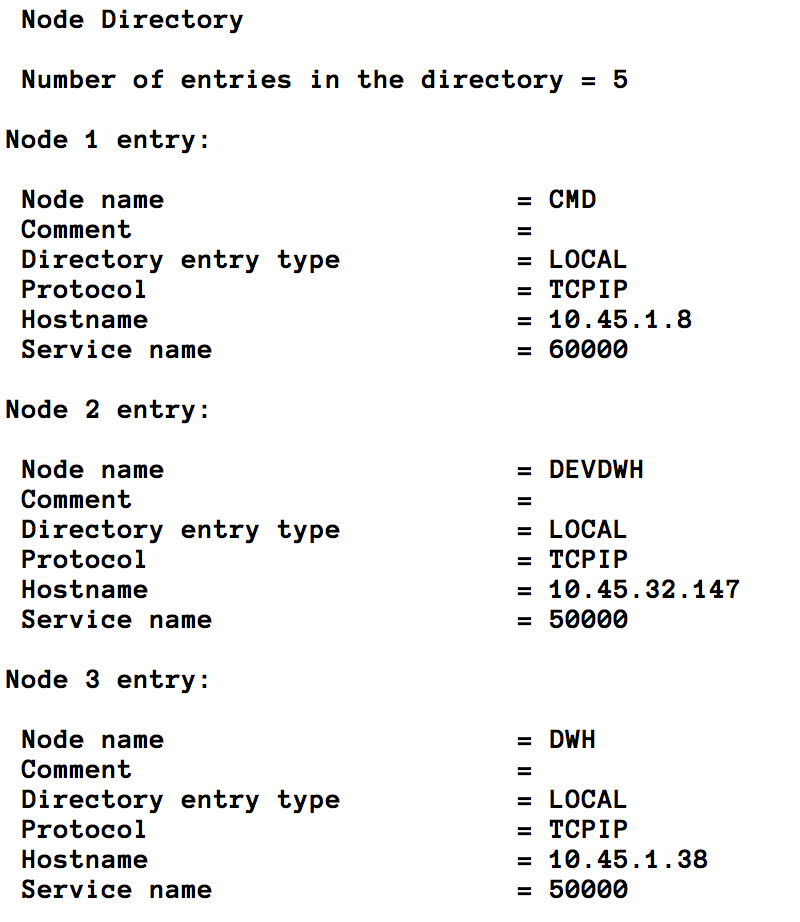
1. Настроить параметры подключения. Подключение будет происходить через консоль db2. Клиент устанавливается вместе с драйверами для подключения.

Клиент находится в директории /opt/ibm/db2/V10.5/bin и называется db2

1. Запускаем консоль:



1. Для того, чтобы установить параметры подключения, необходимо прописать данные в так называемый catalog.
2. Чтобы просмотреть текущий список серверов для подключения, необходимо ввести команду ‘***list node directory***’. Примерный вывод будет выглядеть следующим образом:



1. Чтобы внести новые данные, нужные команды выглядят так:

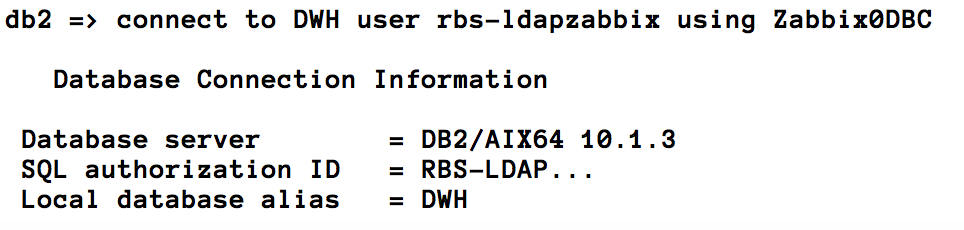
*catalog tcpip node DWH remote 10.10.10.10 server 50000*

*catalog database DWH as DWH at node DWH*

1. После добавления удостовериться, что данные добавлены, снова выведя каталог командой ***list node directory***.
2. Важно проверить, что подключения осуществляется введя команду:

***connect to DWH user rbs-ldapzabbix using uWR.vT2e5B6L1=k4***

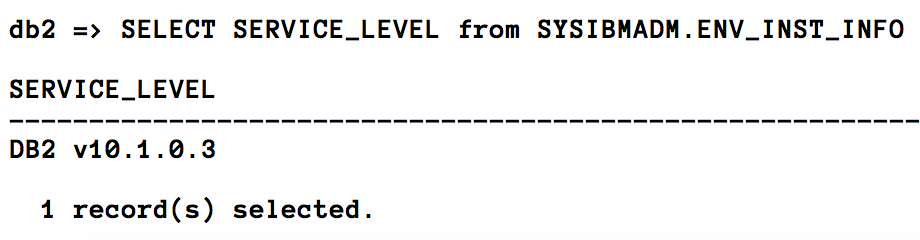
Следующий вывод означает, что подключение осуществлено успешно.



1. Чтобы убедиться, что подключение осуществлено и могут выполнять запросы, ввести команду:

*SELECT SERVICE\_LEVEL from SYSIBMADM.ENV\_INST\_INFO*

В выводе будет значится версия DB2 на сервере:



1. Добавление данных в скрипт:

Данные для подключения хранятся в словаре CRED.

Пример ключа для добавления к словарю CRED

'DWH':

**{'User':'rbs-ldapzabbix',**

**'Password':'uWR.vT2e5B6L1=k4'**

**},**

1. Создать новый хост в Zabbix. С помощью функции клонирования Zabbix, из базового шаблона ‘Template DB2 no ODBC v2’ создать шаблон ‘Template DB2 no ODBC v2 [имя хоста]’. Привязать созданный хост к созданному шаблону.
2. Для корректной работы функций прогнозирования, задать в макросах шаблона пороговые значения в байтах для переменных **{$DB\_FREE\_SIZE\_FORECAST\_LIMIT}**, **{$DB\_GROWTH\_FORECAST\_LIMIT}**, **{$DB\_SIZE\_FORECAST\_LIMIT}** и **{$FS\_FREE\_SIZE\_FORECAST\_LIMIT}**.

!

**Важно:** Параметр **Host name** в режиме конфигурирования хоста в Zabbix должен **совпадать** с соответствующим ключом в словаре CRED скрипта ‘Template DB2 no ODBC v2’. В противном случае, скрипт не будет иметь возможность подключиться к соответствующей хосту базе данных.

!

**Важно:** Для корректной работы функций прогнозирования, для каждой отслеживаемой базы данных в Zabbix должен быть создан отдельный Host и Template.

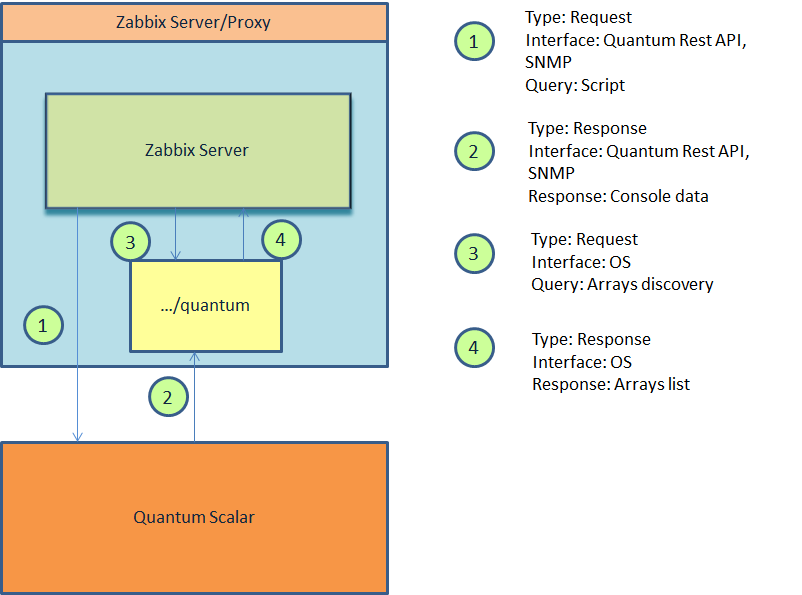
### Шаблоны

Для описанных процедур в системе мониторинга используется шаблон ‘Template DB2 no ODBC v2 [имя хоста]’. На момент написания документации, в системе Zabbix присутствуют шаблоны:

* Template DBMS IBM DB2 v2 DATAHUB
* Template DBMS IBM DB2 v2 DHUB\_F
* Template DBMS IBM DB2 v2 DWH
* Template DBMS IBM DB2 v2 MART
* Template DBMS IBM DB2 v2 (базовый шаблон)

## Quantum Scalar

### Метод сбора информации от объекта мониторинга



Для сбора необходимых метрик и статусов система мониторинга использует скрипт quantum.py с различными параметрами, подключаемый через механизм Rest API к устройствам Quantum Scalar. Информация, полученная от Quantum Scalar, записывается во временные файлы в директорию /usr/local/share/zabbix/externalscripts/quantum, которая должна быть **предварительно создана** на сервере мониторинга. Параметры подключения к Quantum Scalar задаются в фале скрипта в разделе “Credentials”. Например:

**# Credentials**

**u\_credentials = {**

**"RSB-DSPMOS0TL1": {'User': 'monitoring',**

**'Password': 'P@ssw0rd',**

**'Host': '10.45.101.165',**

**},**

**"RSB-DSPMOS1TL1": {'User': 'monitoring',**

**'Password': 'P@ssw0rd',**

**'Host': '10.45.109.165',**

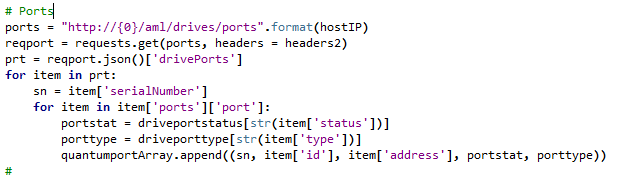
**},**

**}**

Для успешной авторизации при выполнении команд Rest API на устройствах Quantum Scalar вместо обычной аутентификации используются Cookie в заголовках HTTP-запросов. Cookie формируются при выполнении POST-запроса к устройству с помощью команды **http://<HOST>/aml/users/login/.** Скрипт предусматривает правильное формирование HTTP-заголовков с использованием Cookie при последующих GET-запросах для извлечения необходимой информации.

### Конфигурирование объекта мониторинга

В шаблоне системы мониторинга задаются правила обнаружения, соответствующие различным комплектующим устройств Quantum Scalar. При этом параметры запуска скрипта quantum.py зависят от соответствующего правила обнаружения. Запрашиваемая информация записывается во временные файлы с именем вида <Имя хоста>\_<метрика> в формате JSON. Далее система мониторинга использует информацию из временных локальных файлов, автоматически создавая объекты мониторинга на основе заранее подготовленных прототипов данных, либо для вручную созданных объектов. Пример получения информации по портам:



Для сбора части параметров, которые невозможно получить с помощью функций Rest API (например, статус MCB, RCU, CMB), в скрипте предусмотрено получение данной информации через запросы SNMP.

!

**Важно:** Параметр **Host name** в режиме конфигурирования хоста в Zabbix должен **совпадать** со значением имени хоста, передаваемого из скрипта. В противном случае данные, получаемые на этапе обнаружения (Discovery), не будут корректно сопоставлены items, создаваемым в Zabbix.

!

**Важно:** В целях снижения нагрузки на Quantum Scalar, данные для всех items одного типа запрашиваются одновременно, на этапе обнаружения (Discovery rules в шаблоне Zabbix). Обновление состояний items производится из временных файлов, в которые были записаны ранее полученные данные. Соответственно, реальный временной интервал обновления данных items – временной интервал, заданный для выполнения Discovery rules. Это необходимо учитывать при настройке интервалов мониторинга объекта.

!

**Важно:** Во избежание проблем с правами доступа, владельцем директории "/usr/local/share/zabbix/externalscripts/quantum" и файлов внутри должен быть пользователь zabbix: *chown -r zabbix:zabbix /usr/local/share/zabbix/externalscripts/quantum*

### Шаблоны

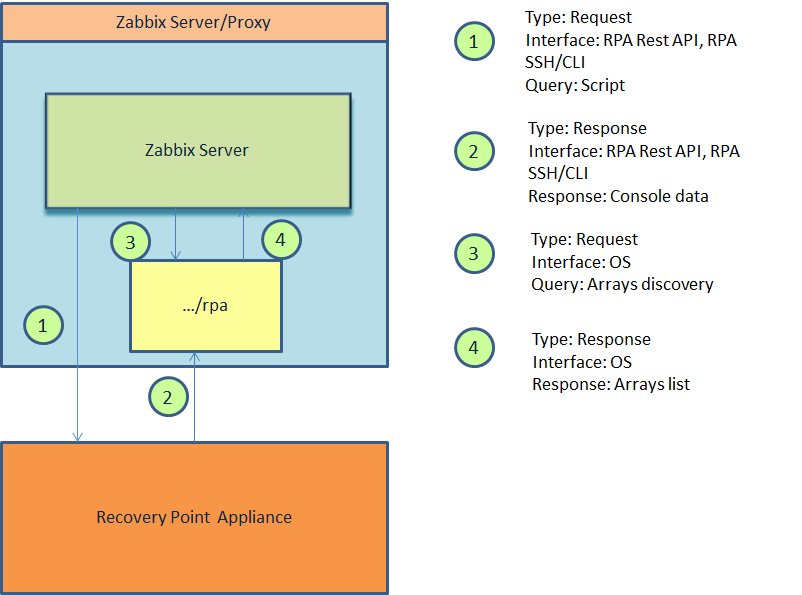
Для описанных процедур в системе мониторинга используется шаблон Template Quantum.

### Описание необходимых настроек на сервере Zabbix

Для настройки шаблонов и объектов на сервере Zabbix необходимо выполнить процедуры, аналогичные описанным в п.3 настоящего документа.

## EMC Recovery Point

### Метод сбора информации от объекта мониторинга



Для сбора необходимых метрик и статусов система мониторинга использует скрипт emc\_rpa.py с различными параметрами, подключаемый через механизмы Rest API и SSH/CLI к устройству Recovery Point. В качестве устройства подключения выбирается сервер с установленной оболочкой Unisphere Recovery Point, который, в свою очередь, обладает всей необходимой информацией обо всех объектах мониторинга. Информация, полученная от сервера с Unisphere Recovery Point, записывается во временные файлы в директорию /usr/local/share/zabbix/externalscripts/rpa, которая должна быть **предварительно создана** на сервере мониторинга. Параметры подключения к серверу Recovery Point задаются в фале скрипта в разделе “Credentials”. Например:

**# Credentials**

**user = 'zabbix'**

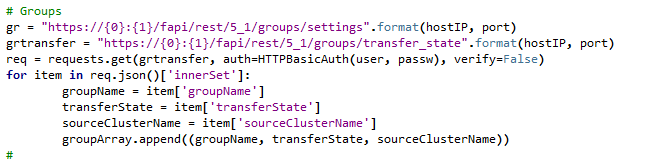
**passw = '4Monit0r!'**

**hostIP = '10.45.101.192'**

**port = '7225'**

### Конфигурирование объекта мониторинга

В шаблоне системы мониторинга задаются правила обнаружения, соответствующие различным комплектующим устройств RPA. При этом параметры запуска скрипта emc\_rpa.py зависят от соответствующего правила обнаружения. Запрашиваемая информация записывается во временные файлы с именем вида <Имя хоста>\_<метрика> в формате JSON. Далее система мониторинга использует информацию из временных локальных файлов, автоматически создавая объекты мониторинга на основе заранее подготовленных прототипов данных, либо для вручную созданных объектов. Пример получения информации по кластерным группам:

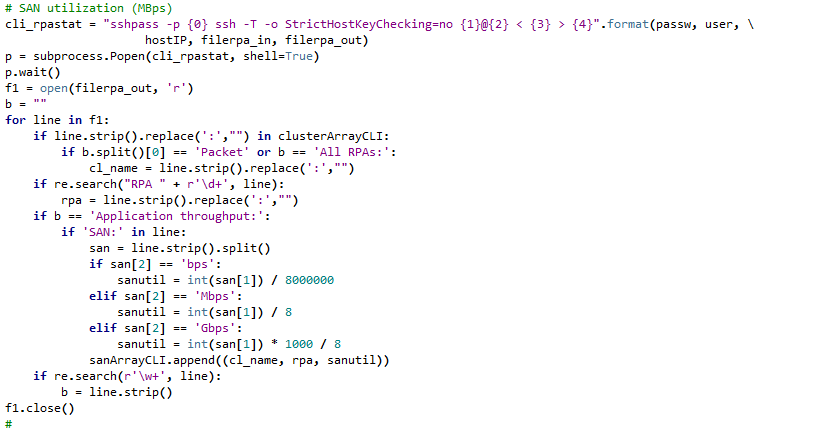


Для сбора части параметров, которые не удаётся получить с помощью функций Rest API, используется подключение к серверу RPA через механизм CLI. Для правильной работы команд **необходимо** наличие в рабочей директории скрипта emc\_rpa.py (или любой другой директории, доступной на чтение пользователю *zabbix*) двух файлов: **rpa\_script.cli** и **group\_script.cli**. Путь к этим файлам должен быть задан в переменных **filerpa\_in** и **filegroup\_in** скрипта emc\_rpa.py:

**filerpa\_in = '/usr/local/share/zabbix/externalscripts/rpa\_script.cli'**

**filegroup\_in = '/usr/local/share/zabbix/externalscripts/group\_script.cli'**

Пример получения информации по утилизации SAN:



**Важно:** Параметр **Host name** для объекта кластера RPA в режиме конфигурирования хоста в Zabbix должен **совпадать** со значением соответствующего имени, передаваемого из скрипта. В противном случае данные, получаемые на этапе обнаружения (Discovery), не будут корректно сопоставлены items, создаваемым в Zabbix.

!

**Важно:** В целях снижения нагрузки на сервер RPA, данные для всех items одного типа запрашиваются одновременно, на этапе обнаружения (Discovery rules в шаблоне Zabbix). Обновление состояний items производится из временных файлов, в которые были записаны ранее полученные данные. Соответственно, реальный временной интервал обновления данных items – временной интервал, заданный для выполнения Discovery rules. Это необходимо учитывать при настройке интервалов мониторинга объекта.

!

**Важно:** Во избежание проблем с правами доступа, владельцем директории "/usr/local/share/zabbix/externalscripts/rpa" и файлов внутри должен быть пользователь zabbix: *chown -r zabbix:zabbix /usr/local/share/zabbix/externalscripts/rpa*

### Шаблоны

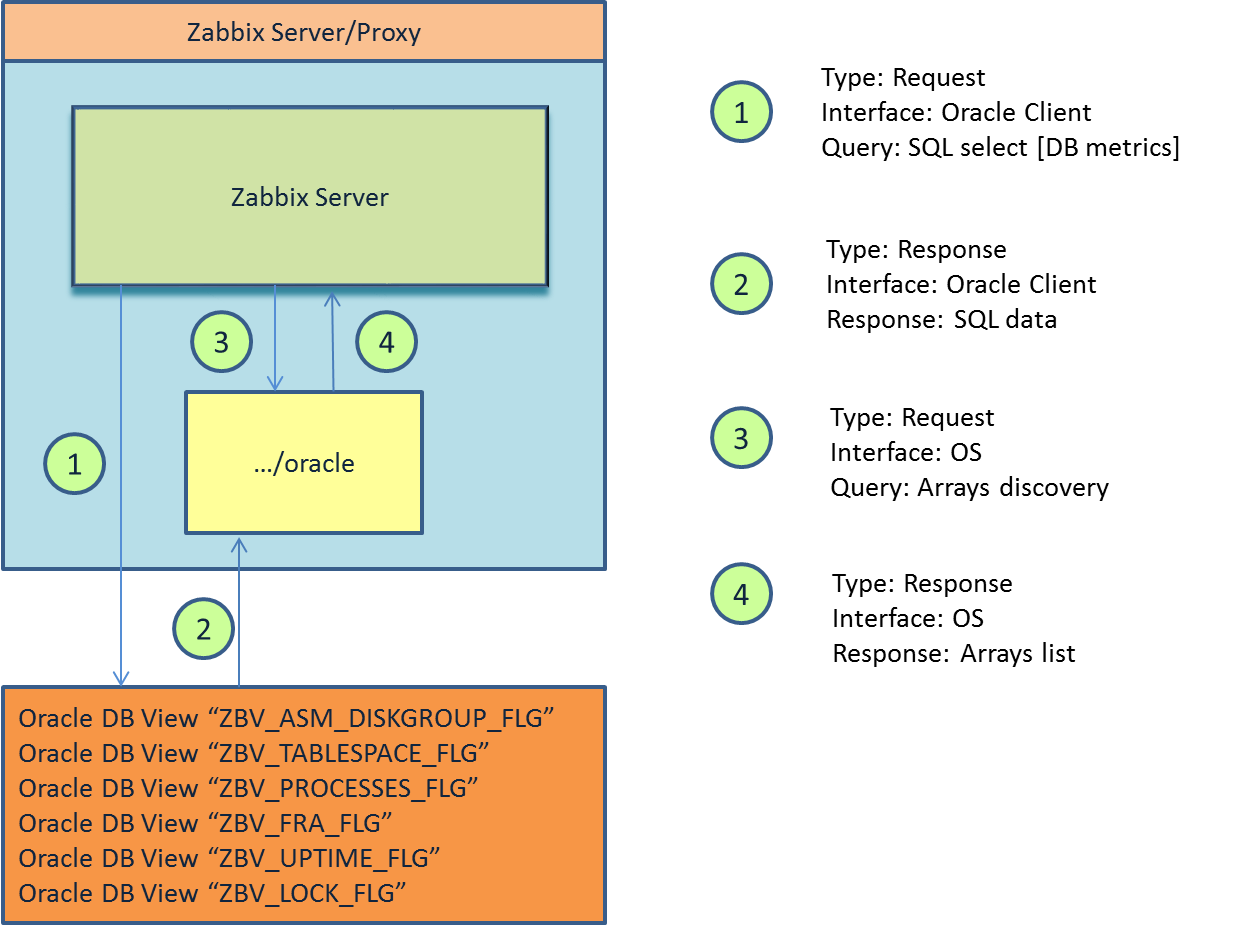
Для описанных процедур в системе мониторинга используется шаблон EMC RecoverPoint.

### Описание необходимых настроек на сервере Zabbix

Для настройки шаблонов и объектов на сервере Zabbix необходимо выполнить процедуры, аналогичные описанным в п.3 настоящего документа.

## Oracle DB

### Метод сбора информации от объекта мониторинга



Для сбора необходимых метрик и статусов, система мониторинга использует скрипт "oracle.py" с различными параметрами. Скрипт обращается к специальному набору представлений, созданных в отслеживаемых базах данных Oracle. При работе с базами данных, скрипт использует возможности модуля Python "cx\_Oracle" и установленного на сервере Zabbix клиента Oracle.

Информация, полученная от представлений БД Oracle, записывается во временные файлы в директории "/usr/local/share/zabbix/externalscripts/oracle", которая должна быть **предварительно создана** на сервере мониторинга. Параметры подключения к отслеживаемой БД Oracle задаются в фале скрипта, в разделе "Credentials" и в файле "tnsnames.ora" клиента Oracle. Пример настройки раздела "Credentials" скрипта:

|  |
| --- |
| # Credentials  u\_credentials **=** **{**  "'SBNRT'"**:** **{**'User'**:** 'ZABBIX'**,**  'Password'**:** 'PEWPEWPEWhandsup1'**,**  'DSN'**:** 'SBNRT'**,**  **},**  "MRT"**:** **{**'User'**:** 'XIBBAZ'**,**  'Password'**:** 'borjomi4'**,**  'DSN'**:** 'MRT'**,**  **},**  **}** |

Пример настройки файла "tnsnames.ora":

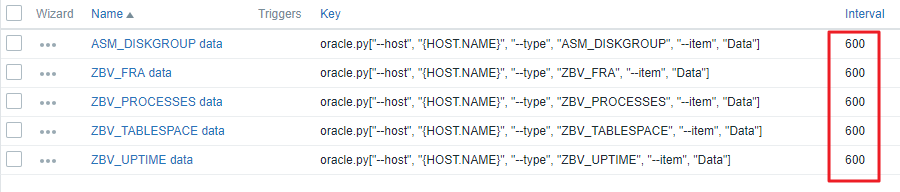
|  |
| --- |
| MRT=  (DESCRIPTION=  (ADDRESS=(PROTOCOL=TCP) (HOST=10.45.22.21) (PORT=1525))  (CONNECT\_DATA=(SERVER=dedicated)(SERVICE\_NAME=MRT)))  SBNRT =  (DESCRIPTION = (ADDRESS\_LIST =  (ADDRESS = (PROTOCOL = TCP)(HOST = rsb-dbpmos0si-scan)(PORT = 1525)) # SMOLNAYA SERVERS  (ADDRESS = (PROTOCOL = TCP)(HOST = rsb-dbpmossidg-scan)(PORT = 1525)) # BOROVAYA SERVER  )  (CONNECT\_DATA =  (SERVICE\_NAME = SBNRT\_PRIMARY)  (SERVER = DEDICATED)  )  ) |

### Конфигурирование объекта мониторинга

В шаблоне системы мониторинга задаются правила обнаружения, соответствующие различным группам метрик (специальным представлениям БД Oracle). Параметры запуска скрипта "oracle.py" зависят от соответствующего правила обнаружения.

Данные для автоматически созданных, в соответствии с правилами обнаружения, Items запрашиваются одномоментно, для всех Items одного типа (относящихся к одной monitoring group Oracle). Запрашиваемая информация записывается во временные файлы с именем вида [Имя хоста]\_[группа метрик], в формате массивов (lists) Python. Интервалы сбора данных задаются как параметр "Update interval" в следующих Items:

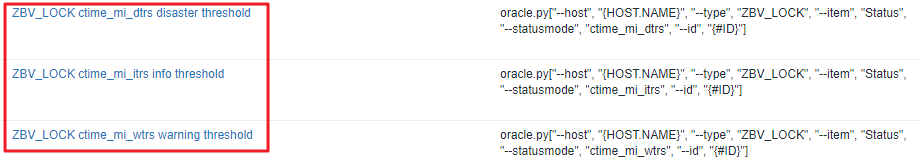
* ASM\_DISKGROUP data
* ZBV\_FRA data
* ZBV\_PROCESSES data
* ZBV\_TABLESPACE data
* ZBV\_UPTIME data



Данные для каждой группы мониторинга запрашиваются скриптом "oracle.py", запускаемым с параметрами, соответствующими группе мониторинга. Пример получения информации по табличным пространствам в скрипте "oracle.py":

|  |
| --- |
| **elif** item\_type **==** 'ZBV\_TABLESPACE'**:**  # Data for Items  **if** args**[**"item"**]** **==** "Data"**:**  dsn\_tns **=** cx\_Oracle**.**makedsn**(**str**(**host**),** str**(**port**),** service\_name**=**str**(**service**))**  db **=** cx\_Oracle**.**connect**(**user**,** pwd**,** dsn\_tns**)**    tablespace\_query\_str **=** "select INSTANCE\_NAME, DB\_UNIQUE\_NAME, HOST\_NAME, METRIC\_GROUP, TABLESPACE\_NAME, CURRENT\_MB, MAX\_MB, USED\_MB, AVAILABLE\_MB, USED\_OF\_MAX\_PCT, " **+** "\n" **+** \  "nvl(AVAILABLE\_MB\_ITRS, 10000000) as AVAILABLE\_MB\_ITRS," **+** "\n" **+** \  "nvl(AVAILABLE\_MB\_WTRS, 10000000) as AVAILABLE\_MB\_WTRS, " **+** "\n" **+** \  "nvl(AVAILABLE\_MB\_DTRS, 10000000) as AVAILABLE\_MB\_DTRS," **+** "\n" **+** \  "nvl(USED\_OF\_MAX\_PCT\_ITRS, 10000000) as USED\_OF\_MAX\_PCT\_ITRS," **+** "\n" **+** \  "nvl(USED\_OF\_MAX\_PCT\_WTRS, 10000000) as USED\_OF\_MAX\_PCT\_WTRS," **+** "\n" **+** \  "nvl(USED\_OF\_MAX\_PCT\_DTRS, 10000000) as USED\_OF\_MAX\_PCT\_DTRS" **+** "\n" **+** \  "from ZBV\_TABLESPACE\_FLG"  cur **=** db**.**cursor**()**  cur**.**execute**(**tablespace\_query\_str**)**  db\_res **=** cur**.**fetchall**()**  db**.**close**()**    # list of data for item status (all\_items) > /tmp/oracle/  filename **=** str**(**args**[**"host"**])** **+** "\_" **+** str**(**args**[**"type"**])** **+** "\_" **+** "all\_items"  write\_to\_file**(**filename**,** db\_res**)**  **print** "OK" |

Далее, система мониторинга использует информацию из временных локальных файлов для обновления состояния автоматически созданных объектов мониторинга (items). Также, в качестве Items, система мониторинга получает информацию о пороговых значениях для автоматически создаваемых триггеров (Items вида "[Имя представления] [отслеживаемая метрика] [severity] threshold").



Таким образом, в шаблоне "Template Oracle DB" нет необходимости в настройке пороговых значений для автоматически создаваемых триггеров. Все необходимые данные поступают из специально созданных представлений БД Oracle.

!

**Важно:** Для корректной работы скрипта "oracle.py", на сервере Zabbix должен быть установлен модуль Python "cx\_Oracle" и клиент БД Oracle. Модуль "cx\_Oracle" может быть установлен через менеджер модулей Python "pip" следующей командой (offline-установка):

*pip install cx\_Oracle-6.4.1-cp27-cp27mu-manylinux1\_x86\_64.whl --no-index*

Файл "cx\_Oracle-6.4.1-cp27-cp27mu-manylinux1\_x86\_64.whl" может быть получен по ссылке <https://pypi.org/project/cx_Oracle/#files>

Для клиента БД Oracle должны быть добавлены настройки отслеживаемых баз в файле "tnsnames.ora".

!

**Важно:** Для корректной работы скрипта "oracle.py", на отслеживаемых БД Oracle должны быть созданы следующие представления:

* ZBV\_ASM\_DISKGROUP\_FLG
* ZBV\_TABLESPACE\_FLG
* ZBV\_PROCESSES\_FLG
* ZBV\_FRA\_FLG
* ZBV\_UPTIME\_FLG
* ZBV\_LOCK\_FLG

Представления создаются администраторами БД Oracle.

!

**Важно:** Параметр **Host name** в режиме конфигурирования хоста в Zabbix должен **совпадать** с соответствующим ключом в словаре CRED (раздел "Credentials") скрипта "oracle.py". В противном случае, скрипт не будет иметь возможность подключиться к соответствующей хосту базе данных.

!

**Важно:** Во избежание проблем с правами доступа, владельцем директории "/usr/local/share/zabbix/externalscripts/oracle" и файлов внутри должен быть пользователь zabbix: *chown -r zabbix:zabbix* /usr/local/share/zabbix/externalscripts/oracle

### Шаблоны

Для описанных процедур в системе мониторинга используется шаблон "Template Oracle DB".

### Настройка возможности мониторинга БД ORACLE с поддержкой ODBC

Для корректной работы мониторинга баз данных Oracle с поддержкой ODBC необходимо, чтобы переменные окружения Oracle были доступны пользователю zabbix. Для этого нужно выполнить следующие шаги:

1. В каталоге /etc/sysconfig/ создать файл zabbix-server
2. В созданный файл zabbix-server добавить строки:

**ORACLE\_HOME=/usr/lib/oracle/11.2/client64**

**LD\_LIBRARY\_PATH=/usr/lib/oracle/11.2/client64/lib:/usr/lib**

**TNS\_ADMIN=/usr/lib/oracle/11.2/client64/network/admin**

**ORACLE\_BASE=/usr/lib/oracle**

**export ORACLE\_HOME**

**export LD\_LIBRARY\_PATH**

**export TNS\_ADMIN**

**export ORACLE\_BASE**

1. В файл /etc/systemd/system/zabbix-server.service в раздел [Service] добавить строку вида:

**EnvironmentFile=-/etc/sysconfig/zabbix-server**

1. Перезапустить сервис сервера Zabbix:

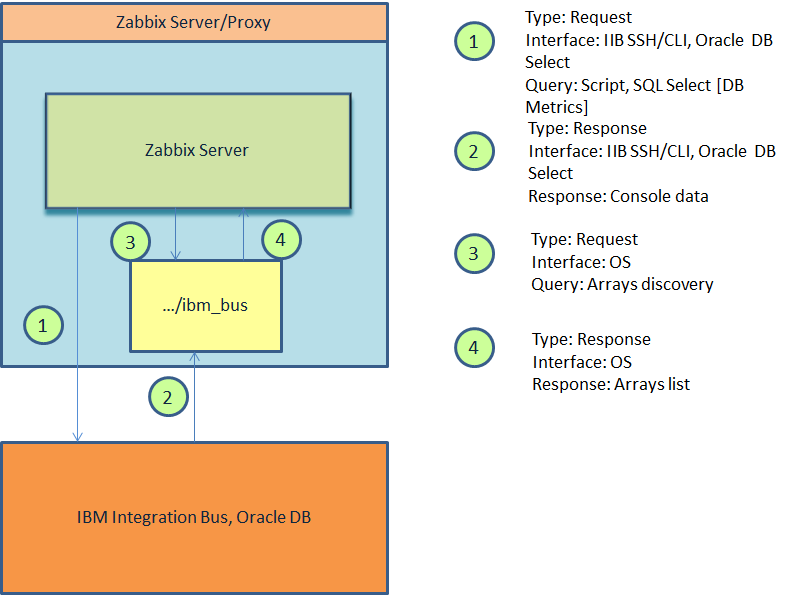
**service zabbix-server restart**

1. Проверить, что переменные окружения Oracle доступны основному процессу сервера Zabbix:

**strings –a /proc/<Main PID Zabbix>/environ**

## IBM Integration Bus

### Метод сбора информации от объекта мониторинга



Для сбора необходимых метрик и статусов система мониторинга использует скрипт ibm\_bus.py с различными параметрами, подключаемый через механизм SSH/CLI к серверам IIB. Кроме того, для извлечения дополнительной информации в скрипте используется прямое подключение к базе Oracle. Информация, полученная от серверов IIB и извлечённая из DB, записывается во временные файлы в директорию /usr/local/share/zabbix/externalscripts/ibm\_bus, которая должна быть **предварительно создана** на сервере мониторинга. Параметры подключения к серверам IIB задаются в фале скрипта в разделе “Credentials”, а также в разделе “Oracle credentials”. Например:

**# Credentials**

**u\_credentials = {**

**"RSBT-ASTIIBNRT2": {'User': ‘iibuser’,**

**'Password': ‘frabR56$ujN’,**

**'Host': '10.45.35.102',**

**}}**

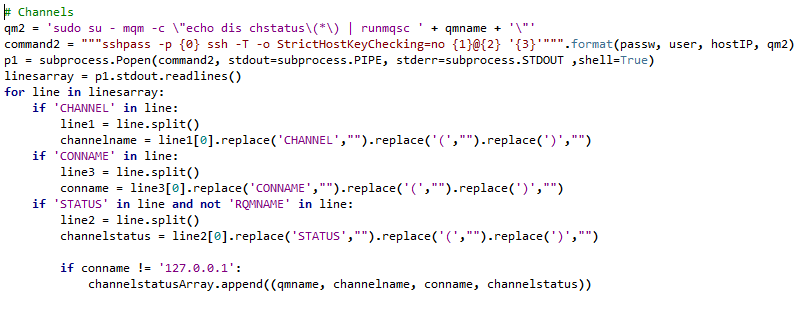
**#Oracle credentials**

**ouser = 'ZABBIX'**

**opwd = 'PEWPEWPEWhandsup1'**

**odsn = 'IIBLOG'**

### Конфигурирование объекта мониторинга

В шаблоне системы мониторинга задаются правила обнаружения, соответствующие различным метриками серверов IIB. При этом параметры запуска скрипта ibm\_bus.py зависят от соответствующего правила обнаружения. Запрашиваемая информация записывается во временные файлы с именем вида <Имя хоста>\_<метрика> в формате JSON. Далее система мониторинга использует информацию из временных локальных файлов, автоматически создавая объекты мониторинга на основе заранее подготовленных прототипов данных, либо для вручную созданных объектов. Часть скрипта, иллюстрирующая получение информации о состоянии каналов:

!

**Важно:** Параметр **Host name** в режиме конфигурирования хоста в Zabbix должен **совпадать** со значением имени хоста, передаваемого из скрипта. В противном случае данные, получаемые на этапе обнаружения (Discovery), не будут корректно сопоставлены items, создаваемым в Zabbix.

!

**Важно:** В целях снижения нагрузки на сервера IIB и на DB Oracle, данные для всех items одного типа запрашиваются одновременно, на этапе обнаружения (Discovery rules в шаблоне Zabbix). Обновление состояний items производится из временных файлов, в которые были записаны ранее полученные данные. Соответственно, реальный временной интервал обновления данных items – временной интервал, заданный для выполнения Discovery rules. Это необходимо учитывать при настройке интервалов мониторинга объекта.

!

**Важно:** Во избежание проблем с правами доступа, владельцем директории "/usr/local/share/zabbix/externalscripts/ibm\_bus" и файлов внутри должен быть пользователь zabbix: *chown -r zabbix:zabbix /usr/local/share/zabbix/externalscripts/ibm\_bus*

### Шаблоны

Для описанных процедур в системе мониторинга используется шаблон “[IBM Integration Bus](http://10.45.35.206/templates.php?form=update&templateid=10215&groupid=10)”.

### Описание необходимых настроек на сервере Zabbix

Для настройки шаблонов и объектов на сервере Zabbix необходимо выполнить процедуры, аналогичные описанным в п.3 настоящего документа.

# Метрики

## NetApp E-Series

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Объект мониторинга** | **Название шаблона** |  |  |  |  |
| **NetApp** | [**Template NetApp E-Series**](http://10.45.35.206/templates.php?form=update&templateid=10180&groupid=32) |  |  |  |  |
|  |  |  |  |  |  |
| **Name** | **Key** | **Descritption** | **Method** | **Type** | **Measurement** |
| [System I/O](http://10.45.35.206/items.php?form=update&hostid=10180&itemid=42293) | netapp.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "IO"] | Суммарное I/O на систему | External script | item | Безразмерное целое число |
| [System response time](http://10.45.35.206/items.php?form=update&hostid=10180&itemid=42287) | netapp.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "resTime"] | Суммарное время отклика на систему | External script | item | ms |
| [System state](http://10.45.35.206/items.php?form=update&hostid=10180&itemid=41587) | netapp.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "Health"] | Состояние системы | External script | item | Строковое значение |
| [System throughput](http://10.45.35.206/items.php?form=update&hostid=10180&itemid=42290) | netapp.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "Throughput"] | Текущая пропускная способность системы | External script | item | MBps |
| [{#NAME} can expire](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41648&parent_discoveryid=41645) | netapp.py["--host", "{HOST.NAME}", "--type", "battery", "--item", "Status", "--statusmode", "CanExpire", "--id", "{#ID}"] | Статус батареи “can expire” | External script | Item prototype | Булевое значение |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41651&parent_discoveryid=41645) | netapp.py["--host", "{HOST.NAME}", "--type", "battery", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние батареи | External script | Item prototype | Строковое значение |
| [{#NAME} CPU average utilization](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41590&parent_discoveryid=41437) | netapp.py["--host", "{HOST.NAME}", "--type", "sp", "--item", "Status", "--statusmode", "CPUaverage", "--id", "{#ID}"] | Текущая утилизация CPU | External script | Item prototype | Проценты |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41440&parent_discoveryid=41437) | netapp.py["--host", "{HOST.NAME}", "--type", "sp", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние CPU | External script | Item prototype | Строковое значение |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41274&parent_discoveryid=41273) | netapp.py["--host", "{HOST.NAME}", "--type", "disk", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние диска | External script | Item prototype | Строковое значение |
| [{#NAME} slot](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41665&parent_discoveryid=41662) | netapp.py["--host", "{HOST.NAME}", "--type", "tray", "--item", "Status", "--statusmode", "Slot", "--id", "{#ID}"] | Расположение дисковой полки | External script | Item prototype | Целое число |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41566&parent_discoveryid=41563) | netapp.py["--host", "{HOST.NAME}", "--type", "fan", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние вентиляторов | External script | Item prototype | Строковое значение |
| [{#NAME} is degrade](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41402&parent_discoveryid=41399) | netapp.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние FC-порта “is degrade” | External script | Item prototype | Булевое значение |
| [{#NAME} utilization %](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41602&parent_discoveryid=41399) | netapp.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "FCPutilization", "--id", "{#ID}"] | Текущая утилизация FC-порта | External script | Item prototype | Проценты |
| [{#NAME} I/O](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41848&parent_discoveryid=41343) | netapp.py["--host", "{HOST.NAME}", "--type", "lun", "--item", "Status", "--statusmode", "IO", "--id", "{#ID}"] | Количество операций I/O для LUN | External script | Item prototype | Безразмерное целое число |
| [{#NAME} response time](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41692&parent_discoveryid=41343) | netapp.py["--host", "{HOST.NAME}", "--type", "lun", "--item", "Status", "--statusmode", "resTime", "--id", "{#ID}"] | Время отклика для LUN | External script | Item prototype | ms |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41346&parent_discoveryid=41343) | netapp.py["--host", "{HOST.NAME}", "--type", "lun", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние LUN | External script | Item prototype | Строковое значение |
| [{#NAME} throughput](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41695&parent_discoveryid=41343) | netapp.py["--host", "{HOST.NAME}", "--type", "lun", "--item", "Status", "--statusmode", "Throughput", "--id", "{#ID}"] | Текущая пропускная способность LUN | External script | Item prototype | MBps |
| {#NAME} free size in % | netapp.py["--host", "{HOST.NAME}", "--type", "pool", "--item", "Status", "--statusmode", "FreePercents", "--id", "{#ID}"] | Свободное место для Pool | External script | Item prototype | Проценты |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41338&parent_discoveryid=41335) | netapp.py["--host", "{HOST.NAME}", "--type", "pool", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние Pool | External script | Item prototype | Строковое значение |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41580&parent_discoveryid=41577) | netapp.py["--host", "{HOST.NAME}", "--type", "ps", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние блока питания | External script | Item prototype | Строковое значение |
| [{#NAME} status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41673&parent_discoveryid=41670) | netapp.py["--host", "{HOST.NAME}", "--type", "sensor", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Статус температурного сенсора | External script | Item prototype | Строковое значение |

## EMC Unity

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Объект мониторинга** | **Название шаблона** |  |  |  |  |
| **EMC Unity** | [**Template EMC Unity v01**](http://10.45.35.206/templates.php?form=update&templateid=10170&groupid=30) |  |  |  |  |
|  |  |  |  |  |  |
| **Name** | **Key** | **Descritption** | **Method** | **Type** | **Measurement** |
| [System I/O](http://10.45.35.206/items.php?form=update&hostid=10180&itemid=42293) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "gs", "--item", "Status", "--statusmode", "IO"] | Суммарное I/O на систему | External script | item | Безразмерное число |
| [System response time](http://10.45.35.206/items.php?form=update&hostid=10180&itemid=42287) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "gs", "--item", "Status", "--statusmode", "RT"] | Суммарное время отклика на систему | External script | item | ms |
| [System General State](http://10.45.35.206/items.php?form=update&hostid=10170&itemid=37101) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "gs", "--item", "Status", "--statusmode", "Health"] | Состояние системы | External script | item | Строковое значение |
| [System Throughput](http://10.45.35.206/items.php?form=update&hostid=10170&itemid=42562) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "gs", "--item", "Status", "--statusmode", "Throughput"] | Текущая пропускная способность на систему | External script | Item | MBps, KBps |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=36491&parent_discoveryid=36490) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "disk", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние диска | External script | Item prototype | Строковое значение |
| {#NAME} state | emc\_unity.py["--host", "{HOST.NAME}", "--type", "dae", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние полки (DAE) | External script | Item prototype | Строковое значение |
| {#NAME} temperature | emc\_unity.py["--host", "{HOST.NAME}", "--type", "dae", "--item", "Status", "--statusmode", "Temperature", "--id", "{#ID}"] | Температура полки (DAE) | External script | Item prototype | Градусы Цельсия |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=37077&parent_discoveryid=37075) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "dpe", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние полки (DPE) | External script | Item prototype | Строковое значение |
| [{#NAME} temperature](http://10.45.35.206/disc_prototypes.php?form=update&itemid=37080&parent_discoveryid=37075) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "dpe", "--item", "Status", "--statusmode", "Temperature", "--id", "{#ID}"] | Температура полки (DPE) | External script | Item prototype | Градусы Цельсия |
| {#NAME} link state | emc\_unity.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "Link", "--id", "{#ID}"] | Состояние линка FC-порта | External script | Item prototype | Строковое значение |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=36520&parent_discoveryid=36518) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние FC-порта | External script | Item prototype | Строковое значение |
| [{#NAME} I/O](http://10.45.35.206/disc_prototypes.php?form=update&itemid=42298&parent_discoveryid=37105) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "lun", "--item", "Status", "--statusmode", "IO", "--id", "{#ID}"] | Количество операций I/O для LUN | External script | Item prototype | IOps |
| [{#NAME} response time](http://10.45.35.206/disc_prototypes.php?form=update&itemid=37107&parent_discoveryid=37105) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "lun", "--item", "Status", "--statusmode", "RT", "--id", "{#ID}"] | Время отклика для LUN | External script | Item prototype | ms |
| [{#NAME} throughput](http://10.45.35.206/disc_prototypes.php?form=update&itemid=42296&parent_discoveryid=37105) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "lun", "--item", "Status", "--statusmode", "Throughput", "--id", "{#ID}"] | Текущая пропускная способность для LUN | External script | Item prototype | MBps, KBps |
| [{#NAME} free size](http://10.45.35.206/disc_prototypes.php?form=update&itemid=37132&parent_discoveryid=36525) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "pool", "--item", "Status", "--statusmode", "FreeGB", "--id", "{#ID}"] | Свободное место для Pool | External script | Item prototype | TB |
| [{#NAME} free size in %](http://10.45.35.206/disc_prototypes.php?form=update&itemid=36527&parent_discoveryid=36525) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "pool", "--item", "Status", "--statusmode", "FreePercents", "--id", "{#ID}"] | Процент свободного места для Pool | External script | Item prototype | Проценты |
| [{#NAME} total size](http://10.45.35.206/disc_prototypes.php?form=update&itemid=37134&parent_discoveryid=36525) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "pool", "--item", "Status", "--statusmode", "TotalGB", "--id", "{#ID}"] | Размер Pool | External script | Item prototype | TB |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=36507&parent_discoveryid=36505) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "ps", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние блока питания | External script | Item prototype | Строковое значение |
| [{#NAME} CPU utilization](http://10.45.35.206/disc_prototypes.php?form=update&itemid=36514&parent_discoveryid=36512) | emc\_unity.py["--host", "{HOST.NAME}", "--type", "sp", "--item", "Status", "--statusmode", "CPU", "--id", "{#ID}"] | Процент утилизации CPU | External script | Item prototype | Проценты |

## IBM DS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Объект мониторинга** | **Название шаблона** |  |  |  |  |
| **IBM DS** | [**Template IBM DS**](http://10.45.35.206/templates.php?form=update&templateid=10175) |  |  |  |  |
|  |  |  |  |  |  |
| **Name** | **Key** | **Descritption** | **Method** | **Type** | **Measurement** |
| [Attention LEDs](http://10.45.35.206/items.php?form=update&hostid=10175&itemid=41331) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "AttentionLED"] | Значение параметра “attentionsMessage” для системы | External script | item | Строковое значение |
| |  |  | | --- | --- | | [Average response time](http://10.45.35.206/items.php?form=update&hostid=10175&itemid=41327) |  | | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "ResponseTimeAll"] | Суммарное время отклика на систему | External script | item | Ms |
| [Serviceable events](http://10.45.35.206/items.php?form=update&hostid=10175&itemid=41333) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "ServiceableEvents"] | Значение параметра “serviceablesMessage” для системы | External script | item | Строковое значение |
| System I/O | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "IOUsage"] | Количество операций I/O для системы | External script | item | IOps |
| System status | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "SystemStatus"] | Значение параметра “systemMessage” для системы | External script | item | Строковое значение |
| [System temperature](http://10.45.35.206/items.php?form=update&hostid=10175&itemid=39927) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "Temperature"] | Температура устройства | External script | item | Градусы Цельсия |
| [System throughput](http://10.45.35.206/items.php?form=update&hostid=10175&itemid=39931) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "DataUsage"] | Текущая суммарная пропускная способность устройства | External script | item | MBps, KBps |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=37893&parent_discoveryid=37891) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "disk", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние диска | External script | Item prototype | Строковое значение |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=38073&parent_discoveryid=38071) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "enclosure", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние дисковой полки | External script | Item prototype | Строковое значение |
| [{#NAME} bandwidth read](http://10.45.35.206/disc_prototypes.php?form=update&itemid=39937&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "BandwitchRead", "--id", "{#ID}"] | Полученные данные для FC-порта в единицу времени | External script | Item prototype | KBps, MBps |
| [{#NAME} bandwidth write](http://10.45.35.206/disc_prototypes.php?form=update&itemid=39939&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "BandwitchWrite", "--id", "{#ID}"] | Переданные данные для FC-порта в единицу времени | External script | Item prototype | KBps, MBps |
| [{#NAME} BitErrRate](http://10.45.35.206/disc_prototypes.php?form=update&itemid=40121&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "BitErrRate", "--id", "{#ID}"] | Количество ошибочных бит за 5 минут для FC-порта | External script | Item prototype | Целое число |
| [{#NAME} CRCErr](http://10.45.35.206/disc_prototypes.php?form=update&itemid=40119&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "CRCErr", "--id", "{#ID}"] | Ошибки проверки целостности фрейма для FC-порта | External script | Item prototype | Целое число |
| [{#NAME} InvTxWordErr](http://10.45.35.206/disc_prototypes.php?form=update&itemid=40117&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "InvTxWordErr", "--id", "{#ID}"] | Значение параметра “InvTxWordErr” для FC-порта | External script | Item prototype | Целое число |
| [{#NAME} LinkFailErr](http://10.45.35.206/disc_prototypes.php?form=update&itemid=40029&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "LinkFailErr", "--id", "{#ID}"] | Количество ошибок FC-линка | External script | Item prototype | Целое число |
| [{#NAME} LossSigErr](http://10.45.35.206/disc_prototypes.php?form=update&itemid=40033&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "LossSigErr", "--id", "{#ID}"] | Количество потерь сигнала на FC-линке с момента последнего нормального сигнала | External script | Item prototype | Целое число |
| [{#NAME} LossSyncErr](http://10.45.35.206/disc_prototypes.php?form=update&itemid=40031&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "LossSyncErr", "--id", "{#ID}"] | Потери из-за ошибок синхронизации для FC-порта | External script | Item prototype | Целое число |
| [{#NAME} PrimSeqErr](http://10.45.35.206/disc_prototypes.php?form=update&itemid=40035&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "PrimSeqErr", "--id", "{#ID}"] | Ошибки протокола согласования для FC-порта | External script | Item prototype | Целое число |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=38821&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние FC-порта | External script | Item prototype | Строковое значение |
| [{#NAME} utilization](http://10.45.35.206/disc_prototypes.php?form=update&itemid=41621&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "UtilizeFCP", "--id", "{#ID}"] | Утилизация FC-порта | External script | Item prototype | Проценты |
| [{#NAME} UtilizeCPU](http://10.45.35.206/disc_prototypes.php?form=update&itemid=40123&parent_discoveryid=38819) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "fcp", "--item", "Status", "--statusmode", "UtilizeCPU", "--id", "{#ID}"] | Утилизация CPU для FC-порта | External script | Item prototype | Проценты |
| [{#NAME} accstate](http://10.45.35.206/disc_prototypes.php?form=update&itemid=38841&parent_discoveryid=38839) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "lun", "--item", "Status", "--statusmode", "accstate", "--id", "{#ID}"] | Состояние “access state” для LUN | External script | Item prototype | Строковое значение |
| [{#NAME} configstate](http://10.45.35.206/disc_prototypes.php?form=update&itemid=38845&parent_discoveryid=38839) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "lun", "--item", "Status", "--statusmode", "configstate", "--id", "{#ID}"] | Состояние “config state” для LUN | External script | Item prototype | Строковое значение |
| [{#NAME} datastate](http://10.45.35.206/disc_prototypes.php?form=update&itemid=38843&parent_discoveryid=38839) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "lun", "--item", "Status", "--statusmode", "datastate", "--id", "{#ID}"] | Состояние “data state” для LUN | External script | Item prototype | Строковое значение |
| [{#NAME} I/O](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43183&parent_discoveryid=38839) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "lun", "--item", "Status", "--statusmode", "IO", "--id", "{#ID}"] | Количество операций I/O для LUN | External script | Item prototype | IOps |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=38099&parent_discoveryid=38097) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "pool", "--item", "Status", "--statusmode", "Health", "--id", "{#ID}"] | Состояние пула | External script | Item prototype | Строковое значение |
| [{#NAME} used size in %](http://10.45.35.206/disc_prototypes.php?form=update&itemid=39933&parent_discoveryid=38097) | ibm\_ds8000.py["--host", "{HOST.NAME}", "--type", "pool", "--item", "Status", "--statusmode", "FreeSpace", "--id", "{#ID}"] | Выделенное место для пула | External script | Item prototype | Проценты |

## IBM DB2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Объект мониторинга** | | **Название шаблона** | |  | | | |
| **IBM DB2** | | **Template DB2 no ODBC v2 [имя хоста]** | |  | | | |
|  |  | |  | |  |  |  |
| **Name** | **Key** | | **Descritption** | | **Method** | **Type** | **Measurement** |
| Размер БД | db2\_no\_odbc\_v2.py["--host", "{HOST.NAME}", "--name", "dbsize"] | | Размер БД | | External script | Item | Байты |
| Размер БД (прогноз) | db\_size\_forecast | | Размер БД – прогноз изменения | | Calculated | Item | Байты |
| Размер БД (прогноз достижения порога) | db\_size\_limit\_forecast | | Размер БД – прогноз достижения порога | | Calculated | Item | Секунды |
| Свободное место в БД | db2\_no\_odbc\_v2.py["--host", "{HOST.NAME}", "--name", "dbfsfree"] | | Свободное место в БД | | External script | Item | Байты |
| Свободное место в БД (прогноз) | db\_free\_size\_forecast | | Свободное место в БД - прогноз изменения | | Calculated | Item | Байты |
| Свободное место в БД (прогноз достижения порога) | db\_free\_size\_limit\_forecast | | Свободное место в БД - прогноз достижения порога | | Calculated | Item | Секунды |
| Свободное место на FS | db2\_no\_odbc\_v2.py["--host", "{HOST.NAME}", "--name", "datafsfree"] | | Свободное место на FS | | External script | Item | Байты |
| Свободное место на FS (прогноз) | fs\_free\_size\_forecast | | Свободное место на FS - прогноз изменения | | Calculated | Item | Байты |
| Свободное место на FS (прогноз достижения порога) | fs\_free\_size\_limit\_forecast | | Свободное место на FS - прогноз достижения порога | | Calculated | Item | Секунды |
| Суточный прирост | db2\_no\_odbc\_v2.py["--host", "{HOST.NAME}", "--name", "dbsizediff"] | | Суточный прирост БД | | External script | Item | Байты |
| Суточный прирост (прогноз) | db\_growth\_size\_forecast | | Суточный прирост БД - прогноз изменения | | Calculated | Item | Байты |
| Суточный прирост (прогноз достижения порога) | db\_growth\_size\_limit\_forecast | | Суточный прирост БД - прогноз достижения порога | | Calculated | Item | Секунды |
| Tablespace {#TBSPNAME} {#PARTNUM} Status | db2\_no\_odbc\_v2.py["--host", "{HOST.NAME}", "--name", "state", "--tbsp", "{#TBSPNAME}", "--partnum", "{#PARTNUM}"] | | Статус табличного пространства | | External script | Item prototype | Строковое значение |

## Quantum Scalar

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Объект мониторинга** | **Название шаблона** |  |  |  |  |
| **IBM DS** | [**Template Quantum**](http://10.45.35.206/templates.php?form=update&templateid=10188&groupid=34) |  |  |  |  |
|  |  |  |  |  |  |
| **Name** | **Key** | **Descritption** | **Method** | **Type** | **Measurement** |
| [Number of free drives](http://10.45.35.206/items.php?form=update&hostid=10188&itemid=43063) | quantum.py["--host", "{HOST.NAME}", "--type", "drive", "--item", "Status", "--statusmode", "Free"] | Количество свободных драйвов | External script | item | Целое число |
| [System library state](http://10.45.35.206/items.php?form=update&hostid=10188&itemid=43060) | quantum.py["--host", "{HOST.NAME}", "--type", "library", "--item", "Status", "--statusmode", "State"] | Общий статус библиотеки | External script | item | Строковое значение |
| {#NAME} status | quantum.py["--host", "{HOST.NAME}", "--type", "component", "--item", "Status", "--statusmode", "Status", "--id", "{#ID}"] | Статус компонентов MCB, RCU, CMB | External script | Item prototype | Строковое значение |
| {#NAME} mode | quantum.py["--host", "{HOST.NAME}", "--type", "drive", "--item", "Status", "--statusmode", "Mode", "--id", "{#ID}"] | Режим работы драйва | External script | Item prototype | Строковое значение |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=42590&parent_discoveryid=42588) | quantum.py["--host", "{HOST.NAME}", "--type", "drive", "--item", "Status", "--statusmode", "State", "--id", "{#ID}"] | Состояние драйва | External script | Item prototype | Строковое значение |
| [{#NAME} status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=42591&parent_discoveryid=42588) | quantum.py["--host", "{HOST.NAME}", "--type", "drive", "--item", "Status", "--statusmode", "Status", "--id", "{#ID}"] | Статус драйва | External script | Item prototype | Строковое значение |
| [{#NAME} position](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43751&parent_discoveryid=42592) | quantum.py["--host", "{HOST.NAME}", "--type", "port", "--item", "Status", "--statusmode", "Position", "--id", "{#ID}"] | Идентификатор порта | External script | Item prototype | Целое число |
| [{#NAME} status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=42593&parent_discoveryid=42592) | quantum.py["--host", "{HOST.NAME}", "--type", "port", "--item", "Status", "--statusmode", "Status", "--id", "{#ID}"] | Статус порта | External script | Item prototype | Строковое значение |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43031&parent_discoveryid=43025) | quantum.py["--host", "{HOST.NAME}", "--type", "robot", "--item", "Status", "--statusmode", "State", "--id", "{#ID}"] | Состояние робота | External script | Item prototype | Строковое значение |
| [{#NAME} status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43028&parent_discoveryid=43025) | quantum.py["--host", "{HOST.NAME}", "--type", "robot", "--item", "Status", "--statusmode", "Status", "--id", "{#ID}"] | Статус робота | External script | Item prototype | Строковое значение |
| [{#NAME} mode](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43051&parent_discoveryid=43042) | quantum.py["--host", "{HOST.NAME}", "--type", "tower", "--item", "Status", "--statusmode", "Mode", "--id", "{#ID}"] | Режим работы Tower | External script | Item prototype | Строковое значение |
| [{#NAME} state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43048&parent_discoveryid=43042) | quantum.py["--host", "{HOST.NAME}", "--type", "tower", "--item", "Status", "--statusmode", "State", "--id", "{#ID}"] | Состояние Tower | External script | Item prototype | Строковое значение |
| [{#NAME} status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43045&parent_discoveryid=43042) | quantum.py["--host", "{HOST.NAME}", "--type", "tower", "--item", "Status", "--statusmode", "Status", "--id", "{#ID}"] | Статус Tower | External script | Item prototype | Строковое значение |

## EMC Recovery Point

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Объект мониторинга** | **Название шаблона** |  |  |  |  |
| **EMC RPA** | **EMC RecoverPoint** |  |  |  |  |
|  |  |  |  |  |  |
| **Name** | **Key** | **Descritption** | **Method** | **Type** | **Measurement** |
| [{#NAME} remote connectivity status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43921&parent_discoveryid=43915) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "eth", "--item", "Status", "--statusmode", "RCon\_stat", "--id", "{#ID}"] | Статус WAN для Ethernet порта | External script | Item prototype | Строковое значение |
| [{#NAME} status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43918&parent_discoveryid=43915) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "eth", "--item", "Status", "--statusmode", "Status", "--id", "{#ID}"] "State"] | Статус Ethernet порта | External script | Item prototype | Строковое значение |
| [{#NAME}](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43987&parent_discoveryid=43984) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "fc", "--item", "Status", "--statusmode", "Data", "--id", "{#ID}"] | Инвентаризация FC-порта | External script | Item prototype | Строковое значение |
| [{#NAME} replication lag](http://10.45.35.206/disc_prototypes.php?form=update&itemid=44033&parent_discoveryid=43864) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rep\_group", "--item", "Status", "--statusmode", "Rep\_lag", "--id", "{#ID}"] | Задержка для групп репликации | External script | Item prototype | Байты |
| [{#NAME} transfer state](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43867&parent_discoveryid=43864) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rep\_group", "--item", "Status", "--statusmode", "Transfer\_State", "--id", "{#ID}"] | Статус групп репликации | External script | Item prototype | Строковое значение |
| [{#NAME} bandwidth reduction](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43943&parent_discoveryid=43876) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rpa", "--item", "Status", "--statusmode", "CRatio", "--id", "{#ID}"] | Значение Bandwidth Reduction для RPA | External script | Item prototype | Ratio |
| [{#NAME} CPU usage](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43940&parent_discoveryid=43876) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rpa", "--item", "Status", "--statusmode", "CPU", "--id", "{#ID}"] | Утилизация CPU для RPA | External script | Item prototype | Проценты |
| [{#NAME} incoming writes](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43946&parent_discoveryid=43876) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rpa", "--item", "Status", "--statusmode", "IWrite", "--id", "{#ID}"] | Значение параметра Incoming writes для RPA в единицу времени | External script | Item prototype | Writes / sec |
| [{#NAME} local connectivity status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43882&parent_discoveryid=43876) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rpa", "--item", "Status", "--statusmode", "LCon\_stat", "--id", "{#ID}"] | Статус локальных соединений для RPA | External script | Item prototype | Строковое значение |
| [{#NAME} [{#NAME} throughput](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43973&parent_discoveryid=43876) SAN](http://10.45.35.206/disc_prototypes.php?form=update&itemid=44022&parent_discoveryid=43876) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rpa", "--item", "Status", "--statusmode", "SAN", "--id", "{#ID}"] | Текущая нагрузка (throughput) SAN | External script | Item prototype | MBps |
| [{#NAME} software status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43885&parent_discoveryid=43876) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rpa", "--item", "Status", "--statusmode", "SStatus", "--id", "{#ID}"] | Состояние ПО для RPA | External script | Item prototype | Строковое значение |
| [{#NAME} status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43879&parent_discoveryid=43876) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rpa", "--item", "Status", "--statusmode", "Status", "--id", "{#ID}"] | Статус RPA | External script | Item prototype | Строковое значение |
| [{#NAME} throughput](http://10.45.35.206/disc_prototypes.php?form=update&itemid=43973&parent_discoveryid=43876) WAN | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rpa", "--item", "Status", "--statusmode", "Throughput", "--id", "{#ID}"] | Текущая нагрузка (throughput) WAN | External script | Item prototype | MBps |
| [{#NAME} Protection Window](http://10.45.35.206/disc_prototypes.php?form=update&itemid=44886&parent_discoveryid=43864) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rep\_group", "--item", "Status", "--statusmode", "PW", "--id", "{#ID}"] | Значение параметра Current Protection Window | External script | Item prototype | Время (часы, дни, недели, месяцы) |
| [{#NAME} Protection Window status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=44889&parent_discoveryid=43864) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rep\_group", "--item", "Status", "--statusmode", "PWStatus", "--id", "{#ID}"] | Статус для Protection Windows | External script | Item prototype | Строковое значение |
| [Getting ethernet ports data](http://10.45.35.206/items.php?form=update&hostid=10206&itemid=44934) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "eth", "--item", "Data"] | Технический объект | External script | Item | Строковое значение |
| [Getting FC ports data](http://10.45.35.206/items.php?form=update&hostid=10206&itemid=44937) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "fc", "--item", "Data"] | Технический объект | External script | Item | Строковое значение |
| [Getting replication groups data](http://10.45.35.206/items.php?form=update&hostid=10206&itemid=44940) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rep\_group", "--item", "Data"] | Технический объект | External script | Item | Строковое значение |
| [Getting RPAs data](http://10.45.35.206/items.php?form=update&hostid=10206&itemid=44943) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "rpa", "--item", "Data"] | Технический объект | External script | Item | Строковое значение |
| [Getting system data](http://10.45.35.206/items.php?form=update&hostid=10206&itemid=44956) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Data"] | Технический объект | External script | Item | Строковое значение |
| [RPAs status](http://10.45.35.206/items.php?form=update&hostid=10206&itemid=44895) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "RPAstatus"] | Общий статус для всех RPA | External script | Item | Строковое значение |
| [Splitters status](http://10.45.35.206/items.php?form=update&hostid=10206&itemid=44901) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "SplitterStatus"] | Общий статус для Splitters | External script | Item | Строковое значение |
| [System status](http://10.45.35.206/items.php?form=update&hostid=10206&itemid=44892) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "SystemStatus"] | Общий статус для системы | External script | Item | Строковое значение |
| [Volumes status](http://10.45.35.206/items.php?form=update&hostid=10206&itemid=44898) | emc\_rpa.py["--host", "{HOST.NAME}", "--type", "system", "--item", "Status", "--statusmode", "VolumeStatus"] | Общий статус для Volumes | External script | Item | Строковое значение |

## ORACLE DB

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Объект мониторинга** | **Название шаблона** |  |  |  |  |
| **Oracle DB** | **Template Oracle DB** |  |  |  |  |
|  |  |  |  |  |  |
| **Name** | **Key** | **Descritption** | **Method** | **Type** | **Measurement** |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Group {#METRIC\_GROUP} Tablespace {#GROUP\_NAME} free space | oracle.py["--host", "{HOST.NAME}", "--type", "ASM\_DISKGROUP", "--item", "Status", "--statusmode", "free\_mb", "--id", "{#ID}"] | Метрика " ASM\_DISKGROUP free space " | External script | Item prototype | Байты |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Group {#METRIC\_GROUP} Tablespace {#GROUP\_NAME} total space | oracle.py["--host", "{HOST.NAME}", "--type", "ASM\_DISKGROUP", "--item", "Status", "--statusmode", "total\_mb", "--id", "{#ID}"] | Метрика " ASM\_DISKGROUP total space " | External script | Item prototype | Байты |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Group {#METRIC\_GROUP} Tablespace {#GROUP\_NAME} used space | oracle.py["--host", "{HOST.NAME}", "--type", "ASM\_DISKGROUP", "--item", "Status", "--statusmode", "used\_mb", "--id", "{#ID}"] | Метрика " ASM\_DISKGROUP used space " | External script | Item prototype | Байты |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Group {#METRIC\_GROUP} Tablespace {#GROUP\_NAME} used space in percent | oracle.py["--host", "{HOST.NAME}", "--type", "ASM\_DISKGROUP", "--item", "Status", "--statusmode", "used\_pct", "--id", "{#ID}"] | Метрика " ASM\_DISKGROUP used space in percent " | External script | Item prototype | Проценты |
| ASM\_DISKGROUP {#ID} free\_mb disaster threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ASM\_DISKGROUP", "--item", "Status", "--statusmode", "free\_mb\_dtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ASM\_DISKGROUP {#ID} free\_mb info threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ASM\_DISKGROUP", "--item", "Status", "--statusmode", "free\_mb\_itrs", "--id", "{#ID}"] | Пороговое значение триггера | External script |  | Целое число |
| ASM\_DISKGROUP {#ID} free\_mb warning threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ASM\_DISKGROUP", "--item", "Status", "--statusmode", "free\_mb\_wtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script |  |  |
| ASM\_DISKGROUP {#ID} used\_pct disaster threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ASM\_DISKGROUP", "--item", "Status", "--statusmode", "used\_pct\_dtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ASM\_DISKGROUP {#ID} used\_pct info threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ASM\_DISKGROUP", "--item", "Status", "--statusmode", "used\_pct\_itrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ASM\_DISKGROUP {#ID} used\_pct warning threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ASM\_DISKGROUP", "--item", "Status", "--statusmode", "used\_pct\_wtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} FRA group {#FRA\_GROUP} current | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_FRA", "--item", "Status", "--statusmode", "current", "--id", "{#ID}"] | Метрика " ZBV\_FRA current " | External script | Item prototype | Байты |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} FRA group {#FRA\_GROUP} free | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_FRA", "--item", "Status", "--statusmode", "free", "--id", "{#ID}"] | Метрика " ZBV\_FRA free " | External script | Item prototype | Байты |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} FRA group {#FRA\_GROUP} max | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_FRA", "--item", "Status", "--statusmode", "max", "--id", "{#ID}"] | Метрика " ZBV\_FRA max " | External script | Item prototype | Байты |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} FRA group {#FRA\_GROUP} used | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_FRA", "--item", "Status", "--statusmode", "used", "--id", "{#ID}"] | Метрика " ZBV\_FRA used " | External script | Item prototype | Байты |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} FRA group {#FRA\_GROUP} used percent | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_FRA", "--item", "Status", "--statusmode", "used\_pct", "--id", "{#ID}"] | Метрика " ZBV\_FRA used percent " | External script | Item prototype | Проценты |
| ZBV\_FRA {#ID} free\_mb\_dtrs disaster threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_FRA", "--item", "Status", "--statusmode", "free\_mb\_dtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_FRA {#ID} free\_mb\_itrs info threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_FRA", "--item", "Status", "--statusmode", "free\_mb\_itrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_FRA {#ID} free\_mb\_wtrs warning threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_FRA", "--item", "Status", "--statusmode", "free\_mb\_wtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_FRA {#ID} used\_pct\_dtrs disaster threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_FRA", "--item", "Status", "--statusmode", "used\_pct\_dtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_FRA {#ID} used\_pct\_itrs info threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_FRA", "--item", "Status", "--statusmode", "used\_pct\_itrs", "--id", "{#ID}" | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_FRA {#ID} used\_pct\_wtrs warning threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_FRA", "--item", "Status", "--statusmode", "used\_pct\_wtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} block | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "block", "--id", "{#ID}"] | Метрика " ZBV\_LOCK block" | External script | Item prototype | Целое число |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} ctime | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "ctime\_mi", "--id", "{#ID}"] | Метрика " ZBV\_LOCK ctime " | External script | Item prototype | секунды |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} lock held | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "lock\_held", "--id", "{#ID}"] | Метрика " ZBV\_LOCK lock held " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} lock requested | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "lock\_requested", "--id", "{#ID}"] | Метрика " ZBV\_LOCK lock requested " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} lock type | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "lock\_type", "--id", "{#ID}"] | Метрика " ZBV\_LOCK lock type " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} module | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "module", "--id", "{#ID}"] | Метрика " ZBV\_LOCK module " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} object | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "object", "--id", "{#ID}"] | Метрика " ZBV\_LOCK object " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} object type | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "object\_type", "--id", "{#ID}"] | Метрика " ZBV\_LOCK object type " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} osuser | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "osuser", "--id", "{#ID}"] | Метрика " ZBV\_LOCK osuser " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} program | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "program", "--id", "{#ID}"] | Метрика " ZBV\_LOCK program " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} sql id | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "sql\_id", "--id", "{#ID}"] | Метрика " ZBV\_LOCK sql id " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} status | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "status", "--id", "{#ID}"] | Метрика " ZBV\_LOCK status " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} terminal | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "terminal", "--id", "{#ID}"] | Метрика " ZBV\_LOCK terminal " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} SID {#SID} username | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "username", "--id", "{#ID}"] | Метрика " ZBV\_LOCK username " | External script | Item prototype | Строковое значение |
| ZBV\_LOCK ctime\_mi\_dtrs disaster threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "ctime\_mi\_dtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_LOCK ctime\_mi\_itrs info threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "ctime\_mi\_itrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_LOCK ctime\_mi\_wtrs warning threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_LOCK", "--item", "Status", "--statusmode", "ctime\_mi\_wtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Group {#METRIC\_GROUP} Resource name {#RESOURCE\_NAME} free | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_PROCESSES", "--item", "Status", "--statusmode", "free", "--id", "{#ID}"] | Метрика " ZBV\_PROCESSES free " | External script | Item prototype | Байты |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Group {#METRIC\_GROUP} Resource name {#RESOURCE\_NAME} limit | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_PROCESSES", "--item", "Status", "--statusmode", "limit", "--id", "{#ID}"] | Метрика " ZBV\_PROCESSES limit " | External script | Item prototype | Байты |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Group {#METRIC\_GROUP} Resource name {#RESOURCE\_NAME} used | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_PROCESSES", "--item", "Status", "--statusmode", "used", "--id", "{#ID}"] | Метрика " ZBV\_PROCESSES used " | External script | Item prototype | Байты |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Group {#METRIC\_GROUP} Resource name {#RESOURCE\_NAME} used percent | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_PROCESSES", "--item", "Status", "--statusmode", "used\_pct", "--id", "{#ID}"] | Метрика " ZBV\_PROCESSES used percent " | External script | Item prototype | Проценты |
| ZBV\_PROCESSES {#ID} free disaster threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_PROCESSES", "--item", "Status", "--statusmode", "free\_dtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_PROCESSES {#ID} free info threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_PROCESSES", "--item", "Status", "--statusmode", "free\_itrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_PROCESSES {#ID} free warning threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_PROCESSES", "--item", "Status", "--statusmode", "free\_wtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_PROCESSES {#ID} used\_pct disaster threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_PROCESSES", "--item", "Status", "--statusmode", "used\_pct\_dtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_PROCESSES {#ID} used\_pct info threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_PROCESSES", "--item", "Status", "--statusmode", "used\_pct\_itrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_PROCESSES {#ID} used\_pct warning threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_PROCESSES", "--item", "Status", "--statusmode", "used\_pct\_wtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} startup time | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_UPTIME", "--item", "Status", "--statusmode", "startup\_time", "--id", "{#ID}"] | Метрика " ZBV\_UPTIME startup time " | External script | Item prototype | Строковое значение |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} uptime | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_UPTIME", "--item", "Status", "--statusmode", "uptime\_mi", "--id", "{#ID}"] | Метрика " ZBV\_UPTIME uptime " | External script | Item prototype | Секунды |
| Instance {#INSTANCE\_NAME} DB {#DB\_UNIQUE\_NAME} Host {#HOST\_NAME} Metric group {#METRIC\_GROUP} uptime interval | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_UPTIME", "--item", "Status", "--statusmode", "uptime\_interval", "--id", "{#ID}"] | Метрика " ZBV\_UPTIME uptime interval " | External script | Item prototype | Строковое значение |
| ZBV\_UPTIME uptime\_mi\_dtrs disaster threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_UPTIME", "--item", "Status", "--statusmode", "uptime\_mi\_dtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_UPTIME uptime\_mi\_itrs info threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_UPTIME", "--item", "Status", "--statusmode", "uptime\_mi\_itrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |
| ZBV\_UPTIME uptime\_mi\_wtrs warning threshold | oracle.py["--host", "{HOST.NAME}", "--type", "ZBV\_UPTIME", "--item", "Status", "--statusmode", "uptime\_mi\_wtrs", "--id", "{#ID}"] | Пороговое значение триггера | External script | Item prototype | Целое число |

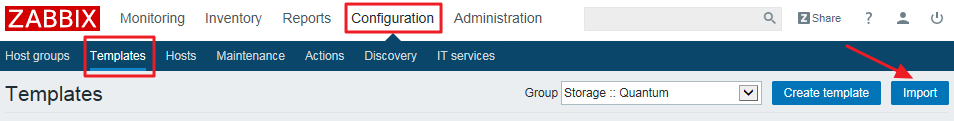
## IBM Integration Bus

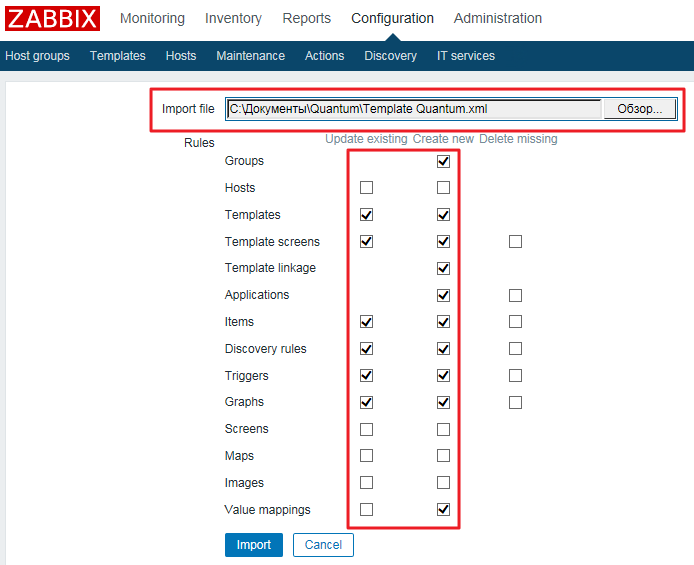
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Объект мониторинга** | **Название шаблона** |  |  |  |  |
| **IBM Integration Bus, DB Oracle** | [**IBM Integration Bus**](http://10.45.35.206/templates.php?form=update&templateid=10215&groupid=10) |  |  |  |  |
|  |  |  |  |  |  |
| **Name** | **Key** | **Descritption** | **Method** | **Type** | **Measurement** |
| [Getting number of incoming requests](http://10.45.35.206/items.php?form=update&hostid=10215&itemid=44833) | ibm\_bus.py["--type", "irn", "--item", "Status"] | Количество входящих запросов по логируемым операциям | External script | Item | Строковое значение |
| [Getting channel status data](http://10.45.35.206/items.php?form=update&hostid=10215&itemid=44662) | ibm\_bus.py["--host", "{HOST.NAME}", "--type", "cs", "--item", "Data"] | Технический объект | External script | Item | Строковое значение |
| [Getting expire queues data](http://10.45.35.206/items.php?form=update&hostid=10215&itemid=44652) | ibm\_bus.py["--host", "{HOST.NAME}", "--type", "eq", "--item", "Data"] | Технический объект | External script | Item | Строковое значение |
| [Getting listener status data](http://10.45.35.206/items.php?form=update&hostid=10215&itemid=44670) | ibm\_bus.py["--host", "{HOST.NAME}", "--type", "ls", "--item", "Data"] | Технический объект | External script | Item | Строковое значение |
| [Channel {#NAME} number of repeat](http://10.45.35.206/disc_prototypes.php?form=update&itemid=44668&parent_discoveryid=44664) | ibm\_bus.py["--host", "{HOST.NAME}", "--type", "cs", "--item", "Status", "--statusmode", "RN", "--id", "{#ID}"] | Количество повторяющихся каналов | External script | Item prototype | Целое число |
| [Channel {#NAME} status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=44666&parent_discoveryid=44664) | ibm\_bus.py["--host", "{HOST.NAME}", "--type", "cs", "--item", "Status", "--statusmode", "Status", "--id", "{#ID}"] | Состояние канала | External script | Item prototype | Строковое значение |
| [Queue {#NAME} TTL](http://10.45.35.206/disc_prototypes.php?form=update&itemid=44656&parent_discoveryid=44654) | ibm\_bus.py["--host", "{HOST.NAME}", "--type", "eq", "--item", "Status", "--statusmode", "ttl", "--id", "{#ID}"] | Время жизни сообщений (очереди с состоянием expired) | External script | Item prototype | Время |
| [Listener {#NAME} status](http://10.45.35.206/disc_prototypes.php?form=update&itemid=44674&parent_discoveryid=44672) | ibm\_bus.py["--host", "{HOST.NAME}", "--type", "ls", "--item", "Status", "--statusmode", "Status", "--id", "{#ID}"] | Состояние Listener | External script | Item prototype | Строковое значение |

# Описание необходимых настроек на сервере Zabbix

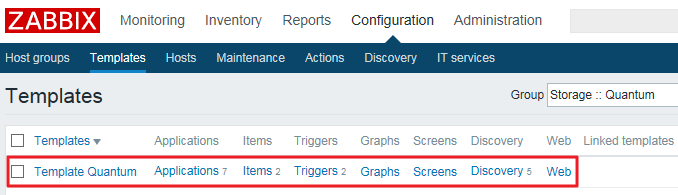
Для настройки шаблонов и объектов на сервере Zabbix необходимо проделать шаги, аналогичные описанным ниже. Настройки представлены на примере ленточной библиотеки Quantum Scalar.

1. Поместить в папку “/usr/local/share/zabbix/externalscripts” скрипт “quantum.py”.
2. Внести в скрипт “quantum.py” изменения, описанные в пункте 1.5.1 настоящего документа.
3. Создать папку “/usr/local/share/zabbix/externalscripts/quantum”. Убедится, что пользователь “zabbix” имеет права для создания и изменения файлов в этом каталоге (см. п 1.5.2 настоящего документа).
4. Из файла “Template Quantum.xml” импортировать шаблон в Zabbix (Configuration > Templates > Import).

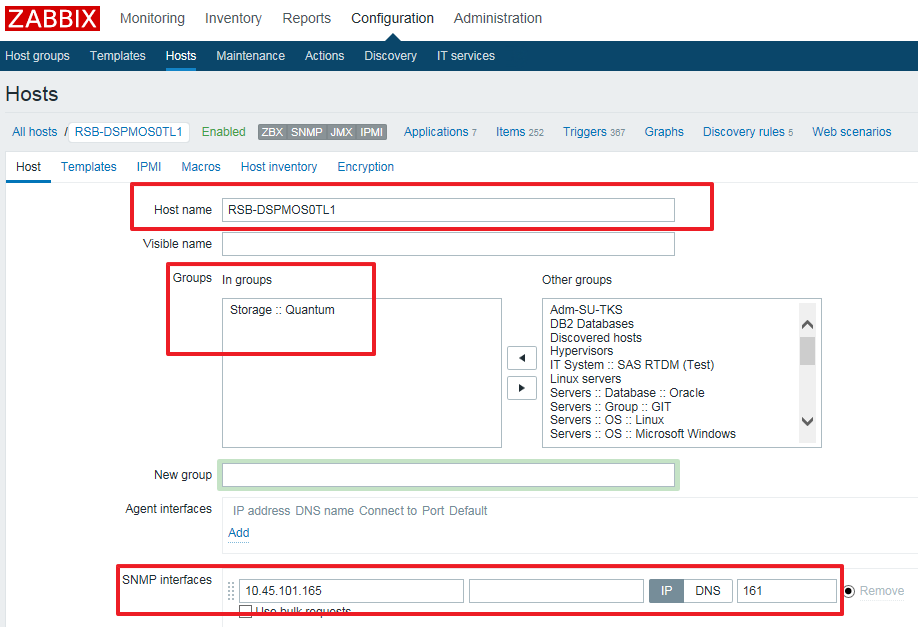




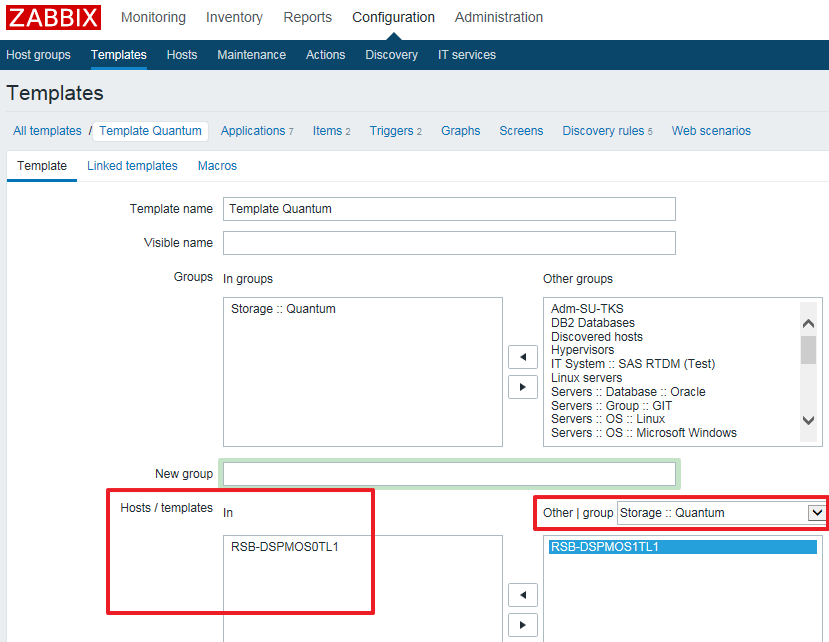
Убедиться, что в списке шаблонов (Configuration > Templates) появился шаблон “Template Quantum”.



1. Создать в Zabbix хосты, соответствующие отслеживаемым ленточным библиотекам (Configuration > Hosts > Create host). Имена создаваемых хостов должны соответствовать именам, внесенным в раздел “Credentials” скрипта “quantum.py” (см пп 1.5.1 и 1.5.2 настоящего документа). При создании хоста необходимо выбрать подходящую группу, указать в поле “SNMP interfaces” FQDN-имя или IP-адрес ленточной библиотеки.



1. Привязать созданные хосты к ранее импортированному шаблону “Template Quantum” (Configuration > Templates > “Template Quantum” > Template > Hosts / templates In).



1. В шаблоне “Template Quantum” необходимо выставить требуемые интервалы получения данных от ленточных библиотек:

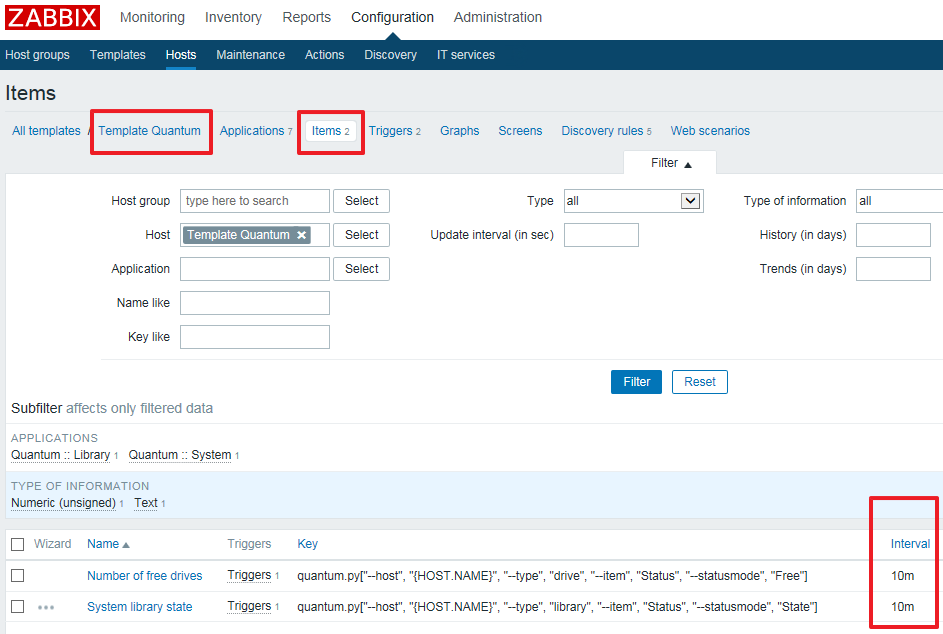
Configuration > Templates > “Template Quantum” > Items > Interval

Configuration > Templates > “Template Quantum” > Discovery rules > Interval

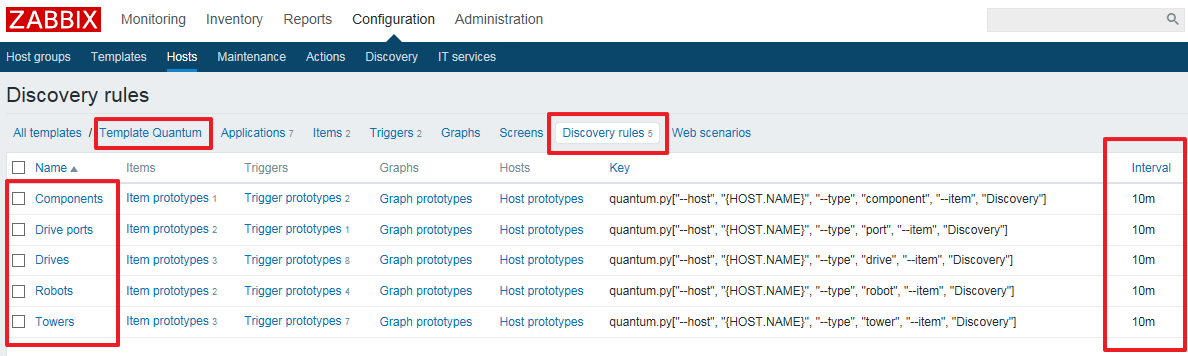
Configuration > Templates > “Template Quantum” > Discovery rules > Item prototypes > Interval

При выставлении интервалов необходимо учитывать информацию, представленную в п. **1.5.2** настоящего документа.

Интервалы для отдельных Items:



Интервалы дискаверенга и получения данных для автоматически создаваемых Items:



Интервалы обновления данных для автоматически создаваемых Items (на примере drives):

