



Программное обеспечение
«СКАЛА-Р МОНИТОРИНГ».
Руководство по эксплуатации

RU.НРФЛ.00014-01.97.01

Москва
02/06/2023

Содержание

Аннотация.....	3
Перечень эксплуатационных документов	4
Идентификационные данные документа	5
Требования к составу и квалификации обслуживающего персонала	6
Описание ПО.....	7
Назначение ПО.....	7
Архитектура и основные компоненты системы.....	7
Клиент	8
Сервер	8
Агент.....	8
Требования к системе.....	9
Условия применения.....	10
Инструменты администратора	10
Удаленный доступ к информации	10
Общие указания.....	12
Действия по безопасной установке и настройке	13
Аварийные ситуации.....	14
Проверка работы MongoDB	17
Проверка работы MySQL	17
Проверка работы Redis.....	18
Действия в случаях обнаружения несанкционированного вмешательства в данные	19
Действия в других ситуациях.....	19
Описание функционирования	20
Термины и определения.....	21
Перечень сокращений	24

Аннотация

Настоящий документ предназначен для технического администратора ПО «СКАЛА-Р МОНИТОРИНГ» и содержит инструкции по выполнению работ, необходимых для эксплуатации ПО.

Руководство содержит инструкции по выполнению задач, связанных с:

- сопровождением и обслуживанием ПО;
- диагностикой, локализацией и устранением неисправностей.

Перечень эксплуатационных документов

Дополнительно к настоящему документу технические администраторы должны использовать следующие документы:

- RU.НРФЛ.00014-01.96.01«ПО «СКАЛА-Р МОНИТОРИНГ». Руководство по установке».
- RU.НРФЛ.00014-01.95.01«ПО «СКАЛА-Р МОНИТОРИНГ». Руководство администратора».

Идентификационные данные документа

Идентификационные данные ПО	Программа для ЭВМ «СКАЛА-Р МОНИТОРИНГ»
Название документа	«ПО «СКАЛА-Р МОНИТОРИНГ». Руководство по эксплуатации»
Обозначение документа	RU.НРФЛ.00014-01.97.01
Автор документа	ООО «БАЗИС»

Требования к составу и квалификации обслуживающего персонала

Администратор ПО – должностное лицо, служебная деятельность которого связана с эксплуатацией программных продуктов и стороннего ПО, используемого при создании среды функционирования.

Администратор платформы СКАЛА-Р МОНИТОРИНГ должен иметь опыт работы с одной из ОС: MS Windows (XP/Vista/7/8/10), Ubuntu Linux, CentOS Linux, Debian Linux и Oracle 5.0, Raspberry Pi (Raspbian OS) или Mac OS X, навык работы с браузером Chrome версии не ниже 36.0.

Квалификация администратора должна позволять:

- устанавливать и конфигурировать агент системы;
- настраивать и конфигурировать сервер;
- решать прикладные задачи мониторинга.

Описание ПО

Назначение ПО

ПО "СКАЛА-Р МОНИТОРИНГ" — представляет собой платформу, предназначенную для постоянного и статического наблюдения и контроля состояния показателей работы сети, оборудования, приложений и сервисов, а также является основной системой мониторинга для платформ виртуализации развернутых с помощью ПО "СКАЛА-Р МОНИТОРИНГ".

ПО "СКАЛА-Р МОНИТОРИНГ" имеет многочисленный набор средств визуализации объектов и их состояний, методы управления критериями смены состояний, средства оповещения и возможности подключения средств автоматической обработки при смене состояний.

Объектом мониторинга могут быть любые физические или логические объекты как самой платформы виртуализации, так и бизнес-приложений, развернутых на платформе – например, память, процессор, файловая система, процесс или программа, количество пользователей, очередь файлов на обработку, объем обработанного трафика, выручка и иные финансовое показатели, значение температуры или химического состава газа или жидкости.

Отличительной особенностью платформы являются возможности хранения оригинальных, не модифицированных значений показателей за значительные промежутки времени с обеспечением высокой скорости записи и доступа к данным, что позволяет производить быстрый и качественный анализ ситуаций в настоящем и прошлом, строить математически обоснованные прогнозы развития ситуации в будущем в рамках контроля как отдельных объектов мониторинга, так и в разрезе показателей и объектов платформы виртуализации различных конфигураций.

Работа ПО "СКАЛА-Р МОНИТОРИНГ" и его настройка осуществляется через Веб-интерфейс, который визуализирует состояние наблюдаемых объектов и обеспечивает доступ к информации об их состоянии.

Архитектура и основные компоненты системы

Архитектура системы содержит три уровня:

- Клиент – тонкий web-клиент системы и клиенты для мобильных операционных систем Android и iOS;
- Сервер – централизованный сервер, на котором хранится и анализируется информация, полученная от агентов, а затем передается клиенту. Также здесь содержится информация об учётных записях пользователей;
- Агент – множество агентов системы, установленных на узлах инфраструктуры и собирающих информацию по ним.

Схема архитектуры системы представлена на рисунке (Рисунок 1).

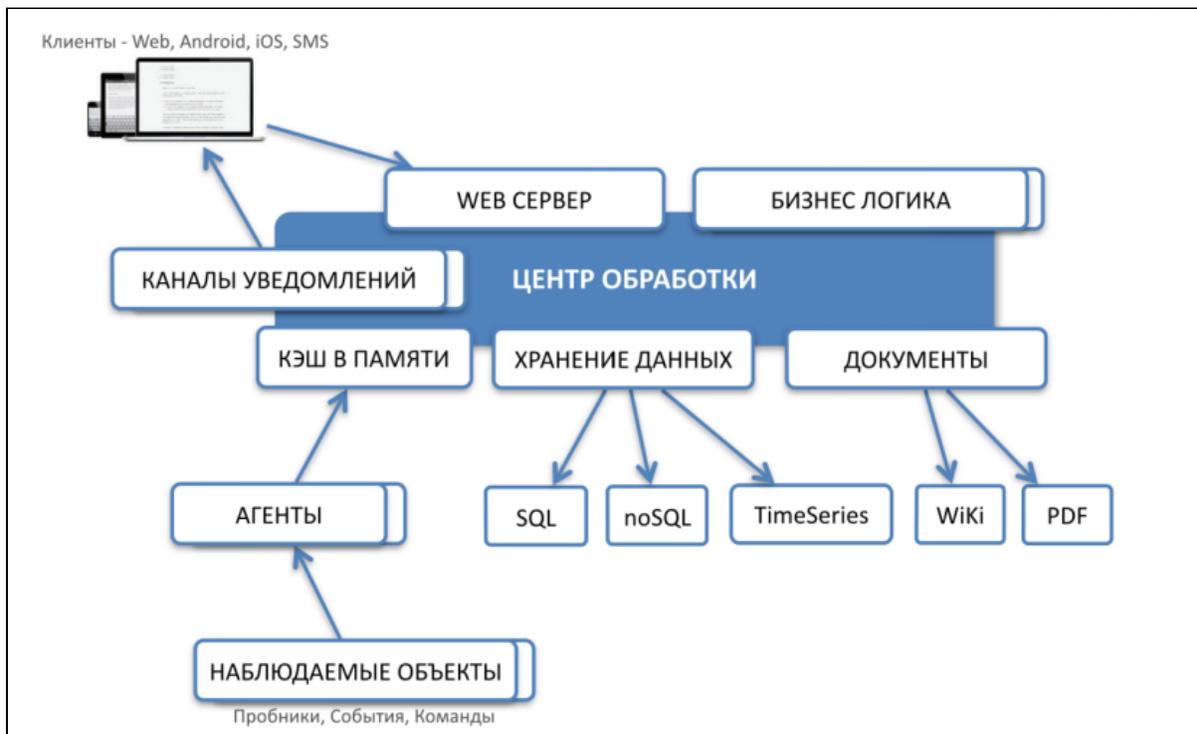


Рисунок 1. Схема архитектуры системы СКАЛА-Р МОНИТОРИНГ

Клиент

Клиент системы – тонкий web-клиент, работающий внутри браузера Google Chrome версии не ниже 58.0. Стабильность работы клиента системы в браузерах Яндекс, Safari, Opera, FireFox не гарантируется.

Для мобильных операционных систем Android и iOS разработаны приложения, поддерживающие работу с платформой для пользователей, находящихся в движении.

Сервер

Сервер включает в себя следующие компоненты:

- Центр обработки (Center) – это логический компонент, объединяющий в себе хранилище данных, кэш, REST-сервер, HTTP-сервер и бизнес-логику.
- Web-сервер – это HTTP-сервер, обслуживающий запросы клиента. Он отдаёт статические элементы web-интерфейса (HTML, JavaScript, CSS), а также проксирует запросы от клиента к REST-серверу. В качестве web-сервера используется NGINX.
- REST-сервер – серверный компонент, обрабатывающий REST-запросы от JavaScript-компонентов на тонком клиенте. Через REST API клиент получает всю информацию об инфраструктуре, а также производит манипуляции с инфраструктурой. REST-сервер реализован в виде отдельного NodeJS-приложения.
- Сервер данных производит анализ поступивших данных. В частности, управляет логикой смены состояний у объектов и связей. Сервер данных реализован в виде отдельного NodeJS-приложения.
- Документы (Хранилище документов) – структура директорий в файловой системе, предназначенная для хранения загруженных документов.
- Кэш в памяти (In-Memory кэш) – NoSQL-хранилище данных типа «ключ-значение». Хранит базу данных в оперативной памяти, благодаря чему возможен быстрый доступ к данным и их быстрая обработка. В качестве In-Memory кэша используется Redis.
- Хранилище данных (База данных). Для постоянного хранения данных используется несколько механизмов:
 - информация по объектам и связям, их свойства и исторические данные о состояниях хранятся в NoSQL-СУБД – MongoDB v.4.2;
 - числовые метрики, по которым строятся графики, сохраняются в базе данных или в специализированном хранилище временных рядов, например, OpenTSDB или InfluxDB.

Администратор серверной части осуществляет работы по установке критических обновлений как системы, так и ПО, входящего в состав серверной части.

Агент

Агент – процесс, собирающий данные о наблюдаемом узле и расположенных на нём объектах, а также связях данного узла. Данные, собранные агентом, периодически отправляются в In-Memory кэш и затем анализируются сервером. Агент реализован в виде Java-приложения.

В некоторых случаях оптимально использовать безагентный (Agentless) мониторинг.

Наблюдаемый объект – элемент инфраструктуры, по которому агент собирает данные. Это может быть, как физический элемент (хост, роутер, сетевая карта, жёсткий диск), так и логический (виртуальная машина, TCP-соединение, файловая система).

Агенты могут собирать данные и об удалённом узле с помощью дистанционных методов:

- SNMP;
- SQL;
- HTTP;
- TCP/UDP.

Более подробно с понятиями "объект" и "связь" в системе можно ознакомиться в документе в разделе "Работа с объектами и связями".

Подробная схема архитектуры системы приведена на рисунке ниже (Рисунок 2).

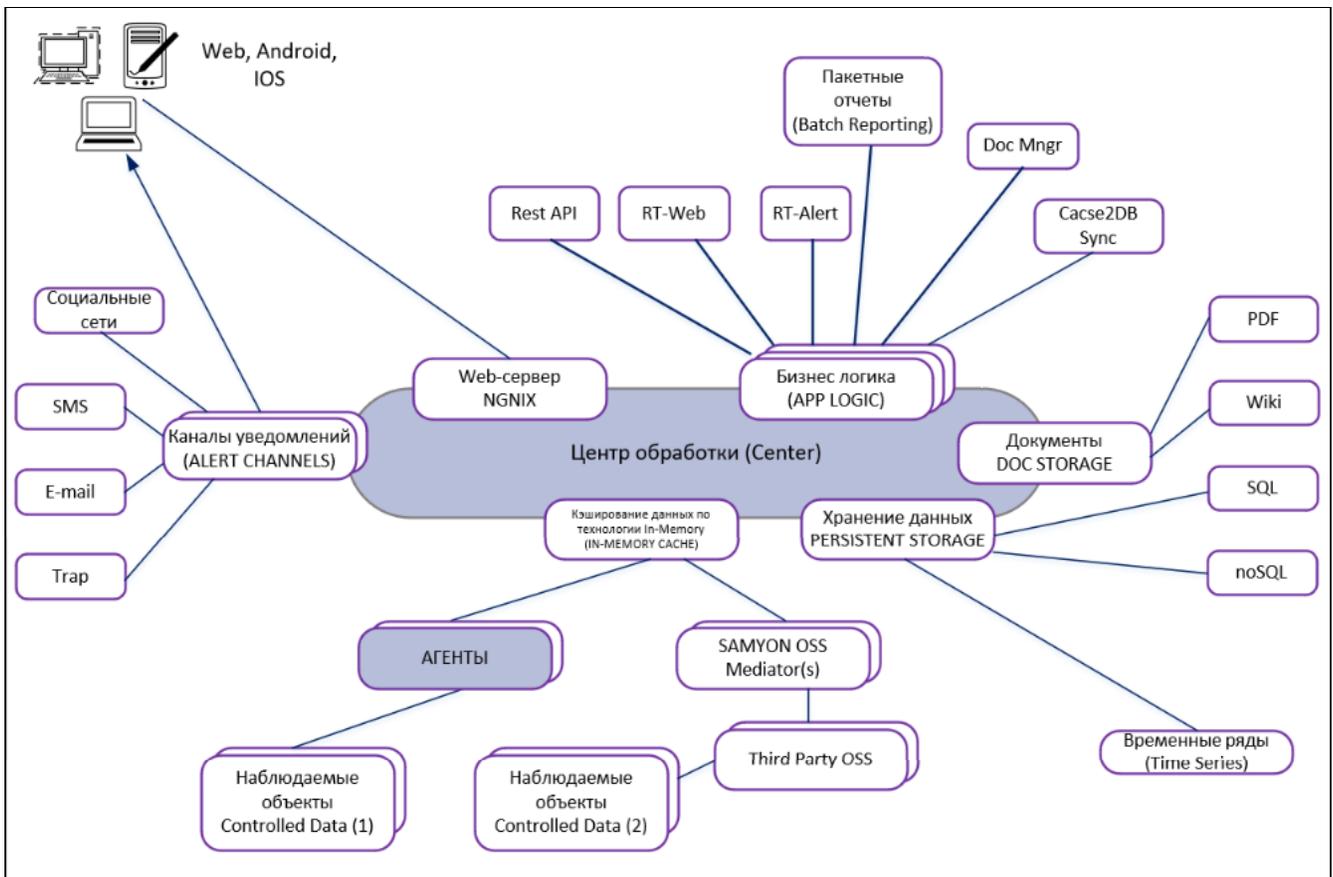


Рисунок 2. Подробная схема архитектуры системы СКАЛА-Р МОНИТОРИНГ

Требования к системе

Требование к клиентскому оборудованию

Для комфортной работы с системой рекомендуются следующие параметры:

- Размер экрана от 20 дюймов;
- Объем свободной оперативной памяти от 512 Мб;
- Скорость интернет – соединения от 2 Мбит/с;
- Устройства воспроизведения при использовании звуковых оповещений.

Технические требования к серверному оборудованию

Система может быть установлена на выделенных аппаратных или виртуальных мощностях.

Для работы сервера системы СКАЛА-Р МОНИТОРИНГ рекомендуется следующая конфигурация:

- 64-bit OS:
 - Ubuntu Linux 10.04 / 12.04 / 14.10 / 16.04;
 - Red Hat Enterprise Linux 5.5+ / 6 / 7;
 - SUSE Linux Enterprise 11 / 12;
- CPU – 2.4 GHz / quad core;
- RAM – от 4 GB;
- HDD – от 24 GB.

Для работы агентов системы рекомендуется следующая конфигурация:

- OS с Java 7/8;
- CPU – 2 GHz single core;
- RAM – 1 GB;
- HDD – OS + 2 GB.

Условия применения

ПО "СКАЛА-Р МОНИТОРИНГ" эксплуатируется в вычислительной среде центра обработки данных (дата-центра), который может быть как централизованным, так и территориально-распределенным.

Основные компоненты составляющие типовой информационно-вычислительный центр (далее – дата-центр или ЦОД):

1. Приложение: компьютерная программа, задающая логику вычислительных операций.
2. Система управления базами данных (СУБД): ПО, обеспечивающее структурированный способ хранения банков (баз) данных.
3. Хост-система (главный компьютер): вычислительная платформа, состоящая из оборудования, программно-аппаратных средств и программного обеспечения, обеспечивающая работу управляющих приложений и СУБД.
4. Сеть: физические каналы обмена данными, обеспечение связи между различными устройствами, подключенными к сети.
5. Хранилище: устройство накопления и постоянного (длительного) хранения данных.

Приложения, наделенные бизнес-логикой, функционируют, как правило, в границах IaaS – изолированной от прямого вмешательства среде.

Качество электропитания, подводимого к хост-системе, сетевому оборудованию и системам хранения данных, равно как и прочим средствам ВТ, включенным в состав облачной инфраструктуры, должно соответствовать действующим нормам.

Инструменты администратора

Деятельность администратора ПО не ограничена использованием одного компьютера (АРМ). В зависимости от характера возникающих задач администратор может использовать различные виды СВТ: от персонального компьютера (ноутбука) с установленной операционной системой Linux до тонкого клиента, с помощью которого пользователь осуществляет подключение к виртуальному рабочему столу.

На СВТ, используемом администратором, должен быть установлен веб-браузер, поддерживаемый операционной системой (Windows, Ubuntu, CentOS и др.). Кроме того, должно быть установлено ПО, позволяющее осуществлять безопасное подключение к управляющим/вычислительным узлам инфраструктуры, а также к вспомогательным виртуальным машинам, если таковые интегрированы в облачную платформу для определенных (сервисно-профилактических) нужд.

Предусмотрен доступ по SSH для развертывания и конфигурирования. Веб-браузер позволяет использовать веб-интерфейс централизованного администрирования ПО.

Удаленный доступ к информации

Удаленный доступ к информации должен обеспечивать безопасные технологии приёма и передачи данных. Например, если настраивается удаленный доступ к серверам, следует использовать SSH или организовывать защищенные (закрытые) сетевые каналы, использующие VPN.

Необходимо соблюдать требования информационной безопасности, установленные в рамках отдела и/или организации. Администратор должен быть бдительным при выполнении авторизации с чужого рабочего места (ТК), так как некоторые веб-браузеры сохраняют вводимые пароли через куки или другими способами.

После того как администратор закончил работу в веб-браузере любого из СВТ, не закрепленного лично за ним, он обязан принять меры по устранению любых сохраненных учетных данных, связанных с доступом к средствам управления или отдельным компонентам облака (имена учетных записей, пароли к ним и т.п.). Записные или электронные книги, равно как и данный документ, не должны находиться без присмотра в помещениях общего пользования.

Внимание

Не допускается случайная или основанная на личном доверии передача третьим лицам учетных данных, смарт-карт, электронных ключей и т.п. средств, позволяющих получить полный или частичный доступ к информации об инфраструктуре.

В конце рабочей смены все персональные компьютеры и СВТ, закрепленные за администратором, должны быть переданы по смене, с соответствующей отметкой в журнале технической эксплуатации, или заблокированы и заперты в специальном помещении, в зависимости от принятых на предприятии организационных мероприятий и политик безопасности.

Если используются АМДЗ, то электронные ключи должны храниться в сейфе или сдаваться под охрану, в соответствии с действующими на предприятии должностными инструкциями по информационной безопасности.

Общие указания

Для реализации функций безопасности среды функционирования ПО «СКАЛА-Р МОНИТОРИНГ» рекомендуется соблюдать следующие требования:

- регулярное обновление всех сред функционирования ПО «СКАЛА-Р МОНИТОРИНГ» до актуальных версий с применением всех необходимых патчей безопасности с официальных сайтов разработчиков сред функционирования;
- компоненты операционной системы и сред функционирования ПО «СКАЛА-Р МОНИТОРИНГ» должны быть максимально ограничены. Компоненты, которые не участвуют в функционировании ПО «СКАЛА-Р МОНИТОРИНГ» должны быть отключены;
- обеспечение предотвращения несанкционированного доступа к идентификаторам и паролям администраторов среды виртуализации, которые необходимы для управления и технической поддержки среды функционирования ПО «СКАЛА-Р МОНИТОРИНГ»;
- использование на серверах, где развернута среда функционирования ПО «СКАЛА-Р МОНИТОРИНГ», в качестве средств защиты информации от несанкционированного доступа, сертифицированных ФСТЭК России версий операционных систем с установленными обновлениями или наложенных средств защиты информации, прошедших сертификацию по требованиям безопасности информации в системе сертификации средств защиты информации по требованиям безопасности информации № РОСС RU.0001.01БИОО;
- обеспечение физической сохранности серверной платформы с установленным ПО «СКАЛА-Р МОНИТОРИНГ» и исключение возможности физического доступа к ней посторонних лиц.

Действия по безопасной установке и настройке

Подробное описание установки ПО приведено в документе «ПО «СКАЛА-Р МОНИТОРИНГ». Руководство по установке» RU.НРФЛ.00014-01.96.01.

Подробные настройки ПО «СКАЛА-Р МОНИТОРИНГ» описаны в документе «ПО «СКАЛА-Р МОНИТОРИНГ». Руководство администратора» RU.НРФЛ.00014-01.95.01.

Аварийные ситуации

Перечень аварийных ситуаций и действия по их устранению приведены в таблице - 2

Таблица 2 - Аварийные ситуации

Класс ошибки	Ошибка	Описание ошибки	Требуемые действия пользователя
Портал платформы СКАЛА-Р МОНИТОРИНГ	Сервер не найден. Невозможно отобразить страницу	Возможны проблемы с сетью или с доступом к порталу платформы СКАЛА-Р МОНИТОРИНГ	Для устранения проблем с сетью обратиться к сотруднику подразделения технической поддержки. В других случаях - к администратору платформы СКАЛА-Р МОНИТОРИНГ
	Требуется ввести действительное имя пользователя	При регистрации на портале платформы СКАЛА-Р МОНИТОРИНГ не введено имя пользователя	Ввести имя пользователя
	Требуется ввести пароль для регистрации	При регистрации на портале платформы СКАЛА-Р МОНИТОРИНГ не введен пароль	Ввести пароль
	Сбой аутентификации Повторите попытку	Неверно введено имя пользователя или пароль, либо такая учетная запись не зарегистрирована	Нужно повторить ввод имени пользователя и пароля
Сбой локальной сети	Нет сетевого взаимодействия между рабочей станцией и сервером приложений платформы СКАЛА-Р МОНИТОРИНГ	Отсутствует возможность начала (продолжения) работы с платформой СКАЛА-Р МОНИТОРИНГ. Нет сетевого подключения к серверу платформы СКАЛА-Р МОНИТОРИНГ.	Перезагрузить рабочую станцию. Проверить доступность сервера платформы СКАЛА-Р МОНИТОРИНГ. После восстановления работы локальной сети повторить попытку подключения (входа) в платформу СКАЛА-Р МОНИТОРИНГ

<p>Виртуальная машина сервером</p>	<p>Нет места на виртуальной машине с сервером СКАЛА-Р МОНИТОРИНГ</p>	<p>Закончилось место на виртуальной машине с сервером СКАЛА-Р МОНИТОРИНГ</p>	<p>1. Понять содержимое и объём занимаемого места:</p> <pre>sudo du -h / sort -h</pre> <p>2. Просмотреть список папок в stdout:</p> <ul style="list-style-type: none"> • если много места занимает папка /var/log/saymon, то можно уменьшить количество хранимых лог-файлов правкой /etc/logrotate.d/saymon для saymon-server.log: rotate X и /opt/saymon-agent/conf/logback-upstart.xml для saymon-agent*.log:<maxHistory>10</maxHistory>; • если много места занимают данные из MongoDB, то зайти в базу данных и оценить размеры коллекций, см Листинг 1. <p>В наиболее объёмных коллекциях используется timestamp, следующей командой можно удалить из коллекции stateHistory массив данных за рамками глубины хранения:</p> <pre>db.stateHistory.remove({'timestamp':{'\$gt':1477994233000}})</pre> <p>После выше описанных действий место в системе не освободится, так как MongoDB аллоцирует дисковое пространство. Требуется сделать бекап и восстановить базу, см. Листинг 2.</p> <ul style="list-style-type: none"> • если много места занимают данные Open TSDB, не вынесенные из Docker-контейнера. Их можно вынести, см. Листинг 3.
---	--	--	---

<p>Взаимодействие агента с сервером</p>	<p>Отсутствие подключения агента к серверу</p>	<p>Агент не подключается к серверу. Запись в логе: 12.10.2020 07:45:59.431 [pool-1-thread-1] WARN n.r.s.agent.connection.Redis Backend - Redis connection failed (will retry in 5 seconds): JedisDataException: ERR max number of clients reached</p>	<p>1. Проверить на сервере проблему локально:</p> <pre># redis-cli -a ' пароль _от_redis_в_кавычках' info clients grep connected_clients sed -e 's/ connected_clients:// g'</pre> <p>Error: Connection reset by peer</p> <p>2. Проверить проблему локально через redis-cli:</p> <pre># redis-cli 127.0.0.1:6379> auth пароль_от_redis (error) ERR max number of clients reached 127.0.0.1:6379> q</pre> <p>3. Рестарт Redis-сервера:</p> <pre># service redis- server restart Stopping redis- server: redis-server Starting redis- server: redis-server</pre>
<p>Проблема OpenTSDB</p>	<p>500 Internal server error</p>	<p>Вместо графиков возникает ошибка 500</p>	<p>Необходимо перезапустить OpenTSDB:</p> <pre>less /var/log/ opentsdb/ opentsdb.log (тут можно увидеть ошибки) sudo service opentsdb stop3. sudo service hbase restart sudo service opentsdb start</pre>

<p>HTTP-проверка</p>	<p>Ошибка работы HTTP-проверки</p>	<p>HTTP-проверка адреса https://xxx.xxx не работает и возникает ошибка</p>	<p>Данная проблема возникает при использовании агента в связке с Java 1.6.</p> <p>Существует 2 варианта решения:</p> <ol style="list-style-type: none"> 1. Обновить установленную Java, в операционной системе, до версии 1.7 или 1.8. 2. Скачать и установить последнюю версию агента со встроенной Java.
-----------------------------	------------------------------------	--	--

Листинг 1

```

mongo saymon
function getReadableFileSizeString(fileSizeInBytes) {
    var i = -1;
    var byteUnits = [' kB', ' MB', ' GB', ' TB', 'PB', 'EB', 'ZB', 'YB'];
    do {
        fileSizeInBytes = fileSizeInBytes / 1024;
        i++;
    } while (fileSizeInBytes > 1024);
    return Math.max(fileSizeInBytes, 0.1).toFixed(1) + byteUnits[i];
};
var collectionNames = db.getCollectionNames(), stats = [];
collectionNames.forEach(function (n) { stats.push(db.getCollection(n).stats()); });
stats = stats.sort(function(a, b) { return b['size'] - a['size']; });
for (var c in stats) {
    print(stats[c]['ns'] + ": " + getReadableFileSizeString(stats[c]['size']) + " (" + getReadableFileSizeString(stats[c]['storageSize']) + ")");
}
    
```

Листинг 2

```

mongodump
sudo rm -rf /var/lib/mongodb/* sudo mongorestore dump/ --dbpath /var/lib/mongodb/ sudo chown -R mongodb:mongodb /var/lib/mongodb sudo service mongod restart
    
```

Листинг 3

```

sudo docker exec -it opentsdb bash cd /data/hbase/hbase-root tar zcvf hbase-root.tar.gz hbase-root scp hbase-root.tar.gz saymon@*host_ip*/opt/ exit cd /opt/ && tar xvf hbase-root.tar.gz sudo docker stop opentsdb sudo docker rm opentsdb sudo docker run -d -p 127.0.0.1:4242:4242 --restart=always -volume /opt/hbase-root/:/data/hbase/hbase-root/ --name=opentsdb rossinno/opentsdb
    
```

Проверка работы MongoDB

Проверка наличия процесса в памяти:

```

ps -ef | grep mongod
mongod 1147 1 0 Nov02 ? 04:23:16 /usr/bin/mongod -config /etc/mongod.conf
    
```

Остановка, запуск и рестарт процесса:

```

sudo service mongod status
sudo service mongod start / stop
sudo service mongod restart
    
```

Проверка работы MySQL

Проверка пароля MySQL (действие на хосте с сервером):

```

cat /etc/saymon/saymon-server.conf
    
```

Просмотр секции db{}

```
"db" : {
  "host" : "localhost",
  "user" : "user",
  "password" : "password",
  "database" : "saymondb"
},
```

Проверка работы Redis

Проверка наличия процесса в памяти:

```
ps -ef | grep redis

redis 1763 1 0 Aug10 ? 00:37:11 /usr/bin/redis-server 0.0.0.0:6379 root 1786 1 0 Aug10 ?
00:00:00
/usr/bin/stunnel4 /etc/stunnel/redis-client.conf root 1787 1 0 Aug10 ? 00:00:00
/usr/bin/stunnel4 /etc/stunnel/redis-client.conf
...
```

Остановка, запуск и рестарт процесса:

```
sudo service redis-server stop/start/restart
```

Номер порта, на котором осуществляется процесс:

```
sudo netstat -lnp | grep redis
tcp 0 0 0.0.0.0:6379 0.0.0.0:* LISTEN 1763/redis-server 0
```

или в конфигурационном файле:

```
cat /etc/saymon/saymon-server.conf | grep cache -A 4
"cache": {
  "auth_pass": "12!@easy",
  "host": "127.0.0.1",
  "port": 6379
},
```

Проверка доступности (открытости) порта:

```
sudo iptables -L INPUT -n -v --line-numbers
Chain INPUT (policy ACCEPT 0 packets, 0 bytes) num pkts bytes target prot opt in out
source destination
1 15M 3082M ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 tcp dpt:6379
```

Добавление порта в список открытых и запись нового правила:

```
sudo iptables -I INPUT 1 -m state --state NEW -p tcp --dport 6379 -j
ACCEPT sudo bash -c "/sbin/iptables-save > /etc/iptables.rules"
```

Проверка доступности порта для агента (действие на хосте с агентом):

```
telnet <адрес_сервера> 6379
Trying <адрес_сервера>...
Connected to <адрес_сервера>.
Escape character is '^['.
```

Проверка пароля Redis (действие на хосте с сервером)

На хосте с сервером (конфигурация Redis):

```
cat /etc/redis/redis.conf | grep requirepass
requirepass Ja!MlK1&
```

```
# If the master is password protected (using the "requirepass" configuration)
# requirepass foobared
```

На хосте с сервером (конфигурация SAYMON):

```
cat /etc/saymon/saymon-server.conf | grep auth_pass
"auth_pass" : "Ja!MlK1&"
```

На хосте с агентом:

```
cat /opt/saymon-agent/conf/agent.properties | grep password
server.password=Ja!MlK1&
```

Пароли должны совпадать, иначе агент не сможет подключиться к серверу для отправки данных.

Действия в случаях обнаружения несанкционированного вмешательства в данные

В случае обнаружения несанкционированного вмешательства в данные необходимо:

1. Изучить журналы и логи системы на предмет доступа и изменения соответствующих данных.
2. Определить корректность настройки прав доступа пользователя, внесшего изменения.
3. При намеренном некорректном изменении данных пользователем, отключить учетную запись пользователя в системе и/или применить административные механизмы воздействия.
4. При необходимости провести восстановление данных из резервной копии.

Действия в других ситуациях

В других аварийных ситуациях необходимо обратиться в сервисную службу:

Электронный адрес: support@basistech.ru

Описание функционирования

Описание совместного функционирования технических средств и ПО, описание организации входных и выходных данных, используемых при обслуживании технических средств и описание взаимодействий устройств с ПО приведено в эксплуатационных документах:

- RU.НРФЛ.00014-01.96.01«ПО «СКАЛА-Р МОНИТОРИНГ». Руководство по установке.
- RU.НРФЛ.00014-01.95.01«ПО «СКАЛА-Р МОНИТОРИНГ». Руководство администратора.

Термины и определения

Термин	Определение
Администратор	лицо, отвечающее за выработку требований к базе данных, её проектирование, реализацию, эффективное использование и сопровождение, включая управление учётными записями пользователей БД и защиту от несанкционированного доступа
Веб – интерфейс	веб-страница или совокупность веб-страниц, предоставляющая пользовательский интерфейс для взаимодействия с сервисом или устройством посредством протокола HTTP и веб-браузера
Виртуальная машина	(VM, от англ. virtual machine) – программная и/или аппаратная система, эмулирующая аппаратное обеспечение компьютера (guest – гостевая платформа) и исполняющая программы для guest-платформы на host-платформе (host – хост-платформа, платформа-хозяин) или виртуализирующая некоторую платформу и создающая на ней среды, изолирующие друг от друга программы и даже операционные системы
Документы (Хранилище документов)	структура директорий в файловой системе, предназначенная для хранения загруженных документов
Жёсткий диск	запоминающее устройство, основанное на принципе магнитной записи
Клиент	тонкий web-клиент системы и клиенты для мобильных операционных систем Android и iOS
Кэш	память с большей скоростью доступа, предназначенная для ускорения обращения к данным, содержащимся постоянно в памяти с меньшей скоростью доступа
Наблюдаемый объект	элемент инфраструктуры, по которому агент собирает данные. Это может быть, как физический элемент (хост, роутер, сетевая карта, жёсткий диск), так и логический (виртуальная машина, TCP-соединение, файловая система)
Объект	физическое устройство (сервер, процессор, маршрутизатор), программный модуль (база данных, Web-сервер) или более высокоуровневые объекты, относящиеся к бизнеспроцессам (услуга, платформа), в пределах которого осуществляется мониторинг с целью сбора информации, её анализа и контроля за состоянием объекта
Роутер	устройство, которое одновременно является и модемом, и маршрутизатором
Сервер	централизованный сервер, на котором хранится и анализируется информация, полученная от агентов, а затем передаётся клиенту. Также здесь содержится информация об учётных записях пользователей
Сервер данных	производит анализ поступивших данных. В частности, управляет логикой смены состояний у объектов и связей
Сетевая карта	компонент электронно-вычислительной техники, с помощью которого она подключается к локальной или глобальной сети

Система управления базами данных (СУБД)	ПО, обеспечивающее структурированный способ хранения банков (баз) данных
Смарт-карта	пластиковые карты со встроенной микросхемой
Учетные данные	имена пользователей и пароли управляемых аккаунтов, позволяющие получать доступ к различным приложениям
Файловая система	порядок, определяющий способ организации, хранения и именования данных на носителях информации в компьютерах
Хост	любое устройство, предоставляющее сервисы формата «клиент-сервер» в режиме сервера по каким-либо интерфейсам и уникально определенное на этих интерфейсах
Хост-система (главный компьютер)	вычислительная платформа, состоящая из оборудования, программно-аппаратных средств и программного обеспечения, обеспечивающая работу управляющих приложений и СУБД
Хранилище	устройство накопления и постоянного (длительного) хранения данных
Центр обработки (Center)	логический компонент, объединяющий в себе хранилище данных, кэш, REST-сервер, HTTP-сервер и бизнес-логику
Agentless	безагентный мониторинг - внешние проверки доступности сервисов по любым TCP и UDP портам, а также ICMP. Отслеживание параметров различных ИТ-объектов. Мониторинг ИТ-процессов на различных объектах инфраструктуры позволяет осуществлять комплексный подход к обслуживанию ИТ-систем предприятия
CPU	(central processing unit, CPU) — центральное обрабатывающее устройство, часто просто процессор
CSS	формальный язык описания внешнего вида документа, написанного с использованием языка разметки. Также может применяться к любым XML-документам, например, к SVG или XUL
HDD	(hard disk drive, жесткий диск) — это запоминающее устройство, основанное на принципе магнитной записи. На сегодняшний день HDD является одним из наиболее распространенных типов памяти, как и flash memory
HTML	стандартизированный язык гипертекстовой разметки документов для просмотра веб-страниц в браузере
HTTP	(HyperText Transfer Protocol) протокол передачи данных, предназначенный для передачи гипертекстовых документов (документов, которые могут содержать ссылки, позволяющие организовать переход к другим документам)
HTTP-проверка	Проверка подлинности в HTTP

HTTP-сервер	сервер, принимающий HTTP-запросы от клиентов, обычно веб-браузеров, и выдающий им HTTP-ответы, вместе с HTML-страницей, изображением, файлом, медиа-поток или другими данными
IaaS	аренда вычислительных мощностей для запуска своих решений и развертывания IT-инфраструктуры компании
InfluxDB	система управления базами данных с открытым исходным кодом для хранения временных рядов, написана на языке Go и не требует внешних зависимостей
In-Memory кэш	NoSQL-хранилище данных типа «ключ-значение». Хранит базу данных в оперативной памяти, благодаря чему возможен быстрый доступ к данным и их быстрая обработка. В качестве In-Memory кэша используется Redis
JavaScript	легковесный, интерпретируемый или JIT-компилируемый, объектно-ориентированный язык с функциями первого класса
MongoDB	документоориентированная система управления базами данных, не требующая описания схемы таблиц. NoSQL-система, использует JSON-подобные документы и схему базы данных. Написана на языке C++
NGINX	программное обеспечение с открытым исходным кодом для создания легкого и мощного веб-сервера. Также его используют в качестве почтового или обратного прокси-сервера
NodeJS	программная платформа, основанная на движке V8, превращающая JavaScript из узкоспециализированного языка в язык общего назначения
NoSQL	категория, объединяющая системы управления базами данных, отличающимися от реляционных БД, для работы с которыми используется язык SQL
OpenTSDB	распределенная, масштабируемая база данных временных рядов (TSDB), написанная поверх HBase
RAM	(Random Access Memory, с англ. — «Запоминающее устройство с произвольным доступом») — оперативная память, один из видов компьютерной памяти
Redis-сервер	(от англ. remote dictionary server) — резидентная система управления базами данных класса NoSQL с открытым исходным кодом, работающая со структурами данных типа «ключ — значение»
REST API	это способ взаимодействия сайтов и веб-приложений с сервером
REST-запросы	серверный компонент, обрабатывающий REST-запросы от JavaScript компонентов на тонком клиенте. Через REST API клиент получает всю информацию об инфраструктуре, а также производит манипуляции с инфраструктурой

REST-сервер	серверный компонент, обрабатывающий REST-запросы от JavaScript компонентов на тонком клиенте. Через REST API клиент получает всю информацию об инфраструктуре, а также производит манипуляции с инфраструктурой
SNMP	(англ. Simple Network Management Protocol — простой протокол сетевого управления) — стандартный интернет-протокол для управления устройствами в IP-сетях на основе архитектур TCP/UDP
SQL	(от англ. Structured Query Language — «язык структурированных запросов») — декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных
SSH	сетевой протокол прикладного уровня, позволяющий производить удалённое управление операционной системой и туннелирование TCP-соединений
TCP	(англ. Transmission Control Protocol — протокол управления передачей) — транспортный протокол передачи данных в сетях TCP/IP, предварительно устанавливающий соединение с сетью
TCP-соединение	TCP-соединение обеспечивает дуплексную передачу данных. Если на двух хостах выполняются соответственно процессы А и В, то данные могут одновременно передаваться как от процесса А к процессу В, так и от процесса В к процессу А
UDP	(англ. User Datagram Protocol — протокол пользовательских датаграмм) — протокол транспортного уровня. С UDP можно посылать сообщения другим хостам по IP-сети без необходимости предварительного сообщения для установки специальных каналов передачи или путей данных
VPN	(англ. virtual private network — «виртуальная частная сеть») — технология, позволяющая обеспечить одно или несколько сетевых соединений поверх чьей-либо другой сети
Web-сервер	HTTP-сервер, обслуживающий запросы клиента. Он отдаёт статические элементы web-интерфейса (HTML, JavaScript, CSS), а также проксирует запросы от клиента к REST-серверу. В качестве web-сервера используется NGINX

Перечень сокращений

В документе использованы следующие сокращения:

Сокращение	Определение
АМДЗ	Аппаратный модуль доверенной загрузки
ВРМ	Виртуальное рабочее место
ОТ	Охрана труда
ПО	Программное обеспечение

Пул	Логический объект СХД, объединяющий пространства нескольких физических накопителей в единое пространство хранения данных
СВТ	Средства вычислительной техники
ТБ	Техника безопасности
ЦОД	Центр обработки данных