

# Корпоративный мессенджер VK Teams

Мониторинг системы

# Оглавление

---

Назначение документа	3
Дополнительная документация	3
Мониторинг базовых (системных) параметров	4
Параметры мониторинга	4
Мониторинг доступности внешнего IP	8
Мониторинг параметров сервиса	9
Скрипт для регулярного мониторинга	9
Интеграция скрипта в систему мониторинга	10
Типы проверок	10
Примеры проверок	11
Сервисы VictoriaMetrics / Grafana	12

# Назначение документа

---

В данной инструкции описаны параметры инсталляции, которые необходимо контролировать.

Документ предназначен для использования администраторами организации.

# Дополнительная документация

---

[Инструкция по подключению к внешней системе мониторинга](#) — в документе описана настройка интеграции с внешними системами мониторинга — VictoriaMetrics и Prometheus.

# Мониторинг базовых (системных) параметров

В разделе приведен список параметров, которые необходимо контролировать, а также рекомендуемые пороги срабатывания.

Действия в зависимости от статуса проблемы:

- **Warning**

Определите, что происходит в системе, и постарайтесь оптимизировать систему так, чтобы вывести ее из этого статуса. Кратковременный переход в состояние Warning не несет опасности для работоспособности системы.

- **Critical**

Ситуация требует немедленного вмешательства. Система работает нестабильно или на пределе.

## Параметры мониторинга

Ниже указаны минимальные параметры для мониторинга, однако рекомендуем контролировать максимально возможное количество параметров. Ориентируйтесь на базовый набор используемых систем мониторинга.

### Пример: мониторинг памяти

В системах с выключенным файлом подкачки (SWAP) основной параметр — количество доступной для приложений памяти: параметр `available` в приложении `free`. Однако возможны ситуации, когда при достаточно большом количестве свободной памяти ее получение будет затруднено. Например, при использовании технологии NUMA возможно неравномерное распределение по NUMA-нодам, что приведет к невозможности распределения памяти в одной из нод.

### CPU

Используйте данные, округленные по достаточно большому периоду времени, не менее 30 секунд. В противном случае будет много ложных срабатываний. Многие параметры зависят от типа инсталляции. Ориентируйтесь не только на пороговые значения, но и на изменение параметров в течение времени. Например, используйте прогноз на дату окончания ресурса.

Параметр	Описание и комментарии	Рекомендации	
		статус Warning	статус Critical
idle	id в top	< 40%	< 20%

Параметр	Описание и комментарии	Рекомендации	
		статус Warning	статус Critical
load average 5m	load average: в top Параметр не является критичным сам по себе, важно его изменение. При возникновении ложных срабатываний порог следует изменить в большую сторону.	> 1x от количества ядер процессора	Critical > 1.5x от количества ядер процессора
wait	wa в top Подбирайте параметр под конкретный тип инсталляции.	> 0,5%	> 1%
steal	steal: в top Количество процессорного времени, в течение которого виртуальная машина не имела доступа к ресурсам CPU. <b>Внимание!</b> Данный параметр стоит подбирать под конкретный тип инсталляции. Проблему можно решить только вне виртуальной машины.	> 2%	> 5%

## Память

Параметр	Описание и комментарии	Рекомендации	
		статус Warning	статус Critical
available	available в free	< 40% от общего	< 20% от общего
shared	shared в free Сравнивайте с количеством разрешенной для распределения памяти в системе.	< 40% от общего	< 20% от общего (kernel.shmall)

Дополнительно: по возможности контролируйте пропорции распределения в NUMA-нодах и уровень фрагментации памяти.

## Диск

Используйте данные, округленные по достаточно большому периоду времени, не менее 30 секунд. В противном случае будет много ложных срабатываний. Многие параметры зависят от типа инсталляции. Например, 20% свободного места для диска 10 Гб — это очень мало, для диска 10 Тб — много. Ориентируйтесь не только на пороговые значения, но и на изменение параметров в течение времени. Например, используйте прогноз на дату окончания ресурса.

Параметр	Описание и комментарии	Рекомендации	
		статус Warning	статус Critical
Свободное место	df -h	занято > 70%	занято > 85%
Нагрузка	Параметры определяйте исходя из системы мониторинга. Здесь указаны предпочтительные параметры на основе вывода iostat -x. <b>Внимание!</b> Значение параметра await стоит подбирать под конкретный тип инсталляции.	• util > 60% • await > 30 мс	• util > 80% • await > 50 мс

## Сеть

Параметр	Описание и комментарии	Рекомендации	
		статус Warning	статус Critical
Загрузка интерфейса	В процентах от пропускной способности.	60%	80%
Ошибки на интерфейсе	(Потери, коллизии и др.)	0,1%	0,5%

Дополнительно: желательно контролировать количество TCP-соединений в различных состояниях: ESTABLISHED, TIME WAIT и т. п.

### Очередь на отправку ОТП-сообщений

Для отправки сообщений используется MTA Postfix. Для просмотра очереди используйте команду `mailq` или `postqueue -p`.

## Синхронизация времени

Для мониторинга статуса ntpd (наличие активных узлов (peer), разница по времени) используйте команды: `ntpq -p`, `ntpstat`

## Мониторинг доступности push-серверов Google и Apple

- <https://support.apple.com/en-is/102266>
- <https://firebase.google.com/docs/cloud-messaging/concept-options?hl=ru#messaging-ports-and-your-firewall>

# Мониторинг доступности внешнего IP

---

Регулярно проверяйте доступность внешнего IP из сети за пределами периметра клиента по следующим портам:

- 443/TCP
- 1443/TCP
- 3478(TCP/UDP)



# Мониторинг параметров сервиса

## Скрипт для регулярного мониторинга

Проверку всех основных параметров сервиса можно выполнять с помощью скрипта:

```
/usr/share/check-mk-agent/local/local_check_exec.py
```

Скрипт проверяет:

- наличие процессов;
- состояние баз данных и сервисов;
- основные TCP-порты.

### Примечание

Проверка происходит внутри системы, по внутренним IP-адресам. Скрипт не проверяет доступы из внешней сети, поэтому такие доступы необходимо проверять отдельно.

Часть проверок выполняется по логам приложений, с расчетом на запуск скрипта раз в минуту. Если запускать скрипт чаще или реже, пороги срабатывания проверок по лог-файлам могут оказаться неадекватными текущим настройкам.

### Формат вывода скрипта

Формат вывода скрипта соответствует требованиям `check_mk local checks`:

- status
- metric\_name
- perfdata
- details

Параметр	Описание	Тип данных
status	Статус проверки: 0 – OK 1 – WARNING 2 – CRITICAL	int

Параметр	Описание	Тип данных
metric_name	Уникальное имя метрики	string
perfdata	Счетчики в формате check_mk	
details	Текстовое описание выполненной проверки в произвольном виде. Может содержать статус в строковом отображении, часть счетчиков и т. п.	

## Интеграция скрипта в систему мониторинга

Чтобы интегрировать скрипт `/usr/share/check-mk-agent/local/local_check_exec.py` в систему мониторинга обслуживающей организации, необходимо вызывать скрипт раз в минуту и добавлять в систему статусы ответов от скрипта. Каждая проверка выводится отдельной строкой, которую следует вносить в ядро мониторинга как отдельную проверку.

### Внимание

Если на момент запуска перестают отдаваться данные по одной или более проверок, то это следует считать критичной проблемой.

Если система мониторинга обслуживающей организации позволяет собирать статистические данные и составлять по ним графики, рекомендуется строить графики по счетчикам скрипта. Такие графики позволят отслеживать нагрузку по отдельным сервисам (память, CPU и т. п.) и оперативно и правильно корректировать настройки сервисов, в зависимости от изменения нагрузки.

Кроме того, данные, которые отклоняются от нормы, передаются во встроенную систему VictoriaMetrics инсталляции VK Teams; оповещения можно увидеть в сервисе Alertmanager, а в mon.sh будут отображаться оповещения из VictoriaMetrics.

## Типы проверок

### Проверки по текущему значению

Запросов в секунды, время отклика и т. п. Изменяет статус ответа, как только происходит нормализация параметров. Проверки такого типа — большая часть всех проверок.

### Проверки по среднему

Изменяют свое значение через некоторое время после изменения параметра. Такими проверками являются, например, `samon-jobs/*` — счетчики неудачных рестартов части сервисов. Для проверки

заданы пороги на количество рестартов каждые 1/5/15/30 минут. По каждому интервалу статус Critical будет выводиться в течение времени, заложенного в этом интервале.

## Проверки с зачисткой

Ошибка будет снята только после ручного удаления статуса ошибки.

Примеры критичных ошибок:

- `error_txt`
- `snmpexec_mon_*_err`

Корректировка таких ошибок выполняется командой:

```
truncate --size 0 /var/tmp/*.{err,txt}
```

По сервисам баз данных (MySQL, Tarantool) дополнительно проверяется параметр `uptime`. Если составляет менее 15 минут, выдается статус Critical. Проверка позволяет вовремя заметить рестарт этих сервисов.

## Примеры проверок

**Пример 1.** Проверка сервиса Beagle на наличие в списке процессов:

```
0 ps_ng_beagle-1 count=1|system_cpu=0;;|user_cpu=0;;|pcpu=0;;|vsz=943;;| rss=202;; (. ) found 1 process
```

Проверка прошла успешно, поэтому `status` — 0 (OK).

Счетчики для построения графиков:

- `count` — количество процессов;
- `system_cpu` — затраты ресурсов в контексте системы в процентах между запусками проверок;
- `user_cpu` — затраты ресурсов в контексте пользователя;
- `vsz` — виртуальная память процесса;
- `rss` — размер страниц памяти.

**Пример 2.** Параметры проверок для экземпляра приложения (инстанса) MySQL №6.

Основные параметры инстанса:

```
0 mysql_6_main uptime=1294487s;600;;300:|threads_connected=18;1500;1900|
threads_created=0.00;50;100|threads_running=2;300;500|threads_cached=47;0;;0:|
aborted_connects=0.00;10;100|aborted_clients=0.00;10;100|qcache_hitrate=0.00%;0;;0:|
qcache_lowmem_prunes=0.00;400;600|slow_queries=0.10;2;5| table_lock_contention=0.00%;10;20|
index_usage=99.67%;0;;0:| open_files=0.00%;70;95|bufferpool_hitrate=100.00%;80;;70:|
bufferpool_wait_free=0.00;1;10|tablecache_hitrate=94.60%;0;;0:| tablecache_fillrate=34.15%;
0;;0:|connection_time=0.01s;2;10| master_position=694895219;0;;0:|
innodb_buffer_pool_pages_data=54.62%;0;;0:| innodb_buffer_pool_pages_free=43.29%;0;;0:|
innodb_buffer_pool_pages_dirty=13.66%;0;;0:| innodb_buffer_pool_pages_flushed=8.79;0;;0:|
```

```
innodb_buffer_pool_pages_lru_flushed=0.00;0;;0|innodb_buffer_pool_size=0.12;0;;0|
innodb_buffer_pool_pages_total=8191.00;0;;0: OK - 0.00925 seconds. Uptime: 1294487 Threads: 2
Slow queries=0.10
```

Счетчики запросов (количество insert, select и т. д. в секунду):

```
0 mysql_6_perf queries=731.58;4000;8000|questions=731.58;2000;5000| commit=2.42;500;1000|
delete=0.52;500;1000|execute_sql=0.00;500;1000| flush=0.00;500;1000|
insert_select=0.00;1000;2000|delete_multi=0.00;100;500| lock_tables=0.00;100;300|
replace=0.00;500;1000|select=528.49;1000;5000| update=28.50;500;1000|
update_multi=0.00;500;1000|insert=142.93;500;1000| set_option=0.12;10000;15000|
call_procedure=0.00;300;900| admin_commands=0.09;1000;5000|change_db=0.00;100;300|
rollback=28.47;100;300 OK - 0.00843 seconds. Questions: 731.58
```

Параметры репликации (в варианте с одной виртуальной машиной, проверка репликации отключена и всегда выдает ОК):

```
0 mysql_6_repl exec_master_log_pos=0;0;;0|read_master_log_pos=0;0;;0: OK - 0.00817 seconds.
```

## Сервисы VictoriaMetrics / Grafana

В инсталляциях VKTeams присутствуют сервисы мониторинга. Доступ к ним можно получить, авторизовавшись в интерфейсе администратора.

Доступ в окружение администратора настраивается при установке системы и описан в документации по установке:

- [На одну виртуальную машину.](#)
- [Кластера.](#)

После этого сервис Grafana станет доступен по адресу:

- для версии VK Teams 24.2 и ниже — <https://admin.<DOMAIN>/myteam-grafana>
- начиная с версии VK Teams 24.3 — <https://admin.<DOMAIN>/grafana>

Чтобы получить логин и пароль учетной записи администратора Grafana, необходимо на машине выполнить команду:

- для версии VK Teams 24.2 и ниже:

```
kubectl get secrets -n prometheus prometheus-stack-grafana -o json | jq '.data |
map_values(@base64d)'
```

- начиная с версии VK Teams 24.3:

```
kubectl get secrets -n monitoring grafana -o json | jq '.data | map_values(@base64d)'
```

Сервис VictoriaMetrics доступен по адресу:

- для версии VK Teams 24.2 и ниже — <https://admin.<DOMAIN>/myteam-prometheus>
- начиная с версии VK Teams 24.3 — <https://admin.<DOMAIN>/mon>

Также этот URL-адрес можно использовать для подключения внешнего сервиса Grafana или сбора метрик другой системой мониторинга с инстанса VictoriaMetrics VK Teams.

Сервис обработки оповещений Alertmanager доступен по адресу:

- для версии VK Teams 24.2 и ниже — <https://admin.<DOMAIN>/myteam-alert>
- начиная с версии VK Teams 24.3 — <https://admin.<DOMAIN>/alertmanager>

Также существует возможность настроить интеграцию с внешней системой мониторинга. Подробнее см. документ [Инструкция по подключению к внешней системе мониторинга](#).