Ай-ТИ Фреш

ТЕХНИЧЕСКИЙ РАЗБОР

Checkmk: корпоративный мониторинг под ключ

Разворачиваем Checkmk 2.4: сайт OMD, TLS-агенты, SNMP, пороги правилами, алерты в Telegram с э...

Июль 2026

itfresh.ru · ИТ-аутсорсинг для юридических лиц

О ЧЁМ ЭТОТ РАЗБОР

Суть проблемы

У большинства компаний до 50 рабочих мест мониторинга нет: о переполненном диске узнают, когда встала база 1С, о деградации RAID — после потери массива, о просроченном сертификате — от клиентов. Мы закрываем это внедрением Checkmk: один сервер наблюдает за серверами, СХД, сетью, ИБП и службами, а дежурный получает алерт раньше, чем проблему заметят сотрудники.

Почему это важно бизнесу

- Час простоя сервера 1С — простой бухгалтерии и продаж целиком; алерт по диску и лицензиям приходит за дни до отказа
- Ремонт после сбоя дороже профилактики: замена диска по SMART-алерту — плановая, после развала RAID — восстановление из бэкапа
- Без истории метрик апгрейды делаются вслепую; графики CPU/RAM/дисков за год показывают, что менять реально пора, а что — нет
- Доступность сервисов подтверждается отчётами и BI-агрегатами, а не словами администратора «вроде всё работало»

ЧТО МЫ ЗАКЛАДЫВАЕМ

Ключевые параметры реализации

2.4.0p34

Рабочая ветка Checkmk в наших внедрениях; 2.5.0 обкатываем на staging до стабильных патчей

наш стандарт внедрения

6556/TCP

Порт агент-контроллера cmk-agent-ctl; после команды register принимает только TLS-соединения

по докам Checkmk

60 с

Интервал проверки хостов и сервисов по умолчанию — оставляем, учащаем только точечно правилом

по докам Checkmk

80/90%

Дефолтные пороги WARN/CRIT ruleset'a Filesystems (used space and growth) — база, тюним по ролям

по докам Checkmk

4/8/100

vCPU / ГБ RAM / ГБ SSD — наш типовой сервер сайта до ~250 хостов; RRD-графикам нужны быстрые с…

наш стандарт

6557/TCP

Порт Livestatus — распределённый мониторинг филиалов из одного центрального сайта

по докам Checkmk

РЕАЛИЗАЦИЯ 1 ИЗ 3

Базовый контур: сайт OMD, агенты, авто-discovery

Что настраиваем

Выделенная VM Debian 12 (4 vCPU / 8 ГБ / 100 ГБ SSD) + все Linux- и Windows-серверы клиента

Как мы это делаем

1. Ставим Checkmk Raw из официального .deb, создаём сайт: omd create monitoring && omd start monitoring; фронт закрываем HTTPS
2. Агенты раскатываем массово: Linux — Ansible (пакет check-mk-agent), Windows — MSI через GPO (msiexec /i check_mk_agent.msi /qn)
3. Каждый хост регистрируем на TLS: cmk-agent-ctl register --hostname ... --server ...; порт 6556 остаётся, но канал только шифрованный
4. Хосты заводим пачкой через REST API 1.0 (POST /domain-types/host_config/collections/all), затем Full service scan — 40–60 сервисов на хост
5. Пороги задаём rule-set'ами (Filesystems (used space and growth) 80/90%, CPU load, Memory) на теги хостов и применяем одной кнопкой Activate changes

РЕЗУЛЬТАТ

Через день у клиента полный инвентарь: диски, CPU, RAM, systemd-юниты, службы Windows, Event Log — с порогами и графиками. Новый сервер добавляется за 5 минут по отработанной схеме, вся конфигурация живёт в правилах, а не в ручных настройках по каждому хосту.

КЛЮЧЕВОЙ НЮАНС

Legacy-режим агента (открытый 6556 без регистрации) оставлять нельзя — данные идут открытым текстом. Проверяем cmk-agent-ctl status на каждом хосте: соединение должно быть в состоянии TLS.

РЕАЛИЗАЦИЯ 2 ИЗ 3

Windows-роли, MSSQL и кастомные чеки для 1С

Что настраиваем

Терминальные серверы, контроллеры домена (AD DS), SQL Server и сервер 1С на площадке клиента

Как мы это делаем

1. Windows-агент сам находит службы, диски, Event Log; правилами Logwatch фильтруем журналы до критичных событий AD и дисковой подсистемы
2. Для SQL Server ставим штатный плагин mk-sql: mk-sql.exe в каталог plugins агента + конфиг mk-sql.yml (auth type integrated) — состояние баз, свежесть бэкапов и tran...
3. Сеансы и лицензии 1С снимаем local-чеком: скрипт в /usr/lib/check_mk_agent/local опрашивает rac session list и печатает строку в формате Checkmk
4. Формат вывода: статус, имя сервиса, метрики вида count=57;80;95 — Checkmk сам строит график и алертит по порогам прямо из вывода скрипта
5. Включаем правило Periodic service discovery (каждые 2 ч), чтобы новые базы и диски попадали под мониторинг без ручного скана

РЕЗУЛЬТАТ

Эксплуатация видит не «сервер жив», а состояние прикладного слоя: рост transaction log, свежесть бэкапов SQL, число сеансов и лицензий 1С. Проблемы прикладного уровня перестают приходить «звонком от пользователей».

КЛЮЧЕВОЙ НЮАНС

Устаревший mssql.vbs не используем — с 2.4 стандарт mk-sql. Local-чеки — самый дешёвый способ мониторить нестандартное: любой скрипт со строкой в формате Checkmk за минуту становится сервисом с графиком и поро...

РЕАЛИЗАЦИЯ 3 ИЗ 3

Сеть и ИБП по SNMP, алерты с эскалацией

Что настраиваем

Коммутаторы, шлюзы, ИБП клиента; уведомления дежурному инженеру в Telegram и почту

Как мы это делаем

1. Сетевые устройства заводим по SNMP v2c/v3: Checkmk сам делает bulkwalk и находит интерфейсы, температуру, блоки питания, батареи ИБП
2. Правилом Network interface and switch port discovery мониторим только uplink и порты серверов, а не все 48 портов каждого свитча
3. Telegram подключаем notification-скриптом в ~/local/share/check_mk/notifications/ (переменные NOTIFY_HOSTNAME, NOTIFY_SERVICESTATE)
4. Эскалация правилами Notifications: алерт дежурному, нет реакции 30 минут — руководителю; ночные некритичные глушим timeperiod'ом
5. Доступность бизнес-сервиса собираем в BI-агрегацию: «1С работает» = хост + SQL + служба сервера 1С + веб-публикация базы

РЕЗУЛЬТАТ

Умирающий ИБП или забитый аплинк видны за дни до инцидента. Руководитель смотрит не на сотни сервисов, а на несколько BI-агрегатов «работает/не работает»; дежурный не просыпается ночью от WARN про очередь печати.

КЛЮЧЕВОЙ НЮАНС

Правила уведомлений важнее самих проверок: без timeperiod и эскалации мониторинг за месяц превращается в игнорируемый шум. Алертов должно быть мало, и все — по делу.

ЧТО МЫ ВЫУЧИЛИ НА ПРАКТИКЕ

Подводные камни

✕ Агент без TLS-регистрации

После установки агент отвечает в legacy-режиме открытым текстом; всегда выполняем cmk-agent-ctl register и проверяем состояние TLS в статусе.

✕ Забытая Activate changes

Правки в Setup лежат в staging, пока не нажата Activate changes; крупные пакеты правил активируем только в техокно, а не среди рабочего дня.

✕ RRD-графики на медленном диске

Тысячи RRD-файлов — это случайные IOPS: на HDD сайт «захлёбывается». Кладём /omd/sites на SSD/NVMe; на ~300 хостов закладываем около 50 ГБ.

✕ Мониторинг всех портов свитча

SNMP-discovery тянет все интерфейсы — сотни бессмысленных сервисов. Правилом discovery оставляем uplink и порты серверов.

✕ За самим Checkmk никто не следит

Упавший сайт молчит «зелёно». Ставим внешний watchdog: проба omd status по cron и HTTP-чек фронта с другого узла с алертом в Telegram.

✕ Алерты без timeperiod

Ночные WARN будят дежурного и приучают игнорировать канал. Некритичное копим до утра правилом уведомлений с рабочим timeperiod.

✕ Прыжки через мажорные версии

omd update переводит только на соседнюю ветку (2.3→2.4→2.5); перед апгрейдом читаем werks и прогоняем копию сайта на staging.

✕ SNMP community «public» навсегда

Дефолтная community живёт в проде годами. Задаём свою v2c или SNMPv3 authPriv и закрываем UDP/161 ACL только до сервера мониторинга.

УРОВНИ РЕАЛИЗАЦИИ

Как правильно

МИНИМУМ

- Checkmk Raw на выделенной VM, агенты с TLS-регистрацией на все серверы
- Full service scan и дефолтные пороги; алерты дежурному на почту
- omd backup сайта раз в сутки на внешнее хранилище

НОРМАЛЬНО

- SNMP-мониторинг сети и ИБП, фильтрация интерфейсов правилами discovery
- Telegram-уведомления с эскалацией и рабочими timeperiod
- Local-чеки на прикладное: 1С, бэкапы, сертификаты, очереди
- Periodic service discovery + ревизия порогов раз в квартал

ХОРОШО

- BI-агрегации бизнес-сервисов и дашборды под роли: инженер, руководитель
- Распределённый мониторинг филиалов через Livestatus (TCP 6557)
- Staging-сайт для обкатки обновлений, апгрейд строго по werks
- Внешний watchdog самого сервера мониторинга с независимым каналом алертов

ПРОВЕРЬТЕ У СЕБЯ

Чек-лист самопроверки

- [ ] Все агенты зарегистрированы по TLS (cmk-agent-ctl status), незашифрованный legacy-доступ к 6556 закрыт?
- [ ] Пороги дисков, CPU и RAM заданы правилами на теги хостов, а не точечно, и пересматриваются раз в квартал?
- [ ] Алерт доходит до дежурного за минуту, и есть эскалация, если на него не отреагировали за 30 минут?
- [ ] Ночные некритичные уведомления отфильтрованы timeperiod'ами — среди ночи будит только CRIT?
- [ ] Сам сервер Checkmk мониторится извне, а его сайт бэкапится командой omd backup ежесуточно?
- [ ] Есть чеки прикладного слоя — бэкапы, 1С, MSSQL, сертификаты, — а не только CPU и диски?
- [ ] SNMP-доступ к сети ограничен: своя community или SNMPv3, UDP/161 открыт только серверу мониторинга?
- [ ] Перед обновлением читаются werks, и апгрейд сначала обкатывается на копии сайта?
- [ ] Новые серверы и базы попадают под мониторинг автоматически (periodic discovery + REST API)?

Если хотя бы на два вопроса ответ «нет» или «не знаю» — тема требует внимания.

ЧТО МЫ ДЕЛАЕМ

Как поможет ITFresh

ITFresh — ИТ-аутсорсинг для юридических лиц до 50 рабочих мест в Москве и области. 15+ лет практики, собственная инфраструктура в дата-центре МТС (8 серверов Dell Xeon Platinum).

- Разворачиваем Checkmk под ключ за 3–7 дней: сервер, раскатка агентов (Ansible/GPO), SNMP, пороги, дашборды
- Пишем кастомные чеки под ваш прикладной слой: 1С, MSSQL, бэкапы, ИБП, сертификаты, очереди
- Настраиваем уведомления с эскалацией в Telegram и почту, BI-агрегации бизнес-сервисов для руководства
- Размещаем сервер мониторинга на нашем железе в ЦОД или разворачиваем на вашей площадке
- Сопровождаем: ревизия порогов, обновления по werks, разбор инцидентов по графикам и истории

15+
лет в ИТ-поддержке

50
рабочих мест — наш профиль

МТС
дата-центр, Москва

КОНТАКТЫ

Обсудить вашу задачу

Сайт	**itfresh.ru**
Телефон	**+7 903 729-62-41**
Telegram	**@ITfresh_Boss**

Бесплатно посмотрим вашу инфраструктуру по этому чек-листу и скажем, где тонко — без обязательств.

itfresh.ru

ДОКУМЕНТАЦИЯ И НАШ СТЕК

Техническая база

01 Installation on Linux (omd create / omd start) (docs.checkmk.com — 2.4)

02 Monitoring Linux — Agent Controller, TLS, порт 6556 (docs.checkmk.com — 2.4)

03 Monitoring Windows — агент и плагины (docs.checkmk.com — 2.4)

04 Monitoring Microsoft SQL Server — плагин mk-sql (docs.checkmk.com — 2.4)

05 Understanding and configuring rules (docs.checkmk.com — 2.4)

06 Notifications — правила, эскалация, timeperiods (docs.checkmk.com — 2.4)

07 Monitoring via SNMP (docs.checkmk.com — 2.4)

08 Distributed monitoring (Livestatus, TCP 6557) (docs.checkmk.com — 2.4)

09 REST API — host_config, bulk-операции (docs.checkmk.com — 1.0)

Основано на официальной документации продуктов и нашей практике внедрения.