



Ай-ТИ Фреш

ТЕХНИЧЕСКИЙ РАЗБОР

Icinga 2: распределённый мониторинг мастер-сателлит-агент

Строим отказоустойчивый мониторинг филиалов: зоны, HA-пары, Icinga DB, Director, TLS-агенты

Июль 2026

itfresh.ru · ИТ-аутсорсинг для юридических лиц

Суть проблемы

У компаний с филиалами мониторинг из центра по VPN живёт плохо: потери пакетов дают ложные CRITICAL, при обрыве канала площадка слепнет, история теряется, а о лежащем сервере 1С админ узнаёт от бухгалтерии. Мы решаем это распределённой Icinga 2: мастер в ЦОД, сателлит на каждой площадке, агенты на хостах — филиал мониторит себя сам, а центр видит всё в одной консоли.

Почему это важно бизнесу

- Простой сервера 1С или почты без алерта — часы потерянной работы офиса; сателлит ловит проблему за минуту даже при упавшем VPN
- 20 ложных алертов в день приучают их игнорировать — реальную аварию пропускают; стейт-машина Icinga режет шум до единиц
- Один экран Icinga Web на все площадки — не нужен отдельный администратор в каждом филиале
- HA-пары master/satellite: отказ самого узла мониторинга не оставляет инфраструктуру без надзора

Ключевые параметры реализации

2.15+

ветка Icinga 2 в наших внедрениях:
IDO устарел и удаляется в 2.18 —
сразу строим на Icinga DB

Icinga 2, Upgrading

5665/tcp

единственный порт связи зон:
взаимная проверка сертификатов
(TLS), соединение в обе стороны

Icinga 2, ApiListener

397 дн

срок сертификата узла; icinga2 сам
перевыпускает его через
родительскую зону за 30 дней до
ист...

Icinga 2, CHANGELOG

60 с

check_interval критичных сервисов;
retry_interval 30 с,
max_check_attempts 3 до
HARD-состояния

наш стандарт

1 день

log_duration replay-лога: столько
сателлит копит результаты при
обрыве и досылает после линка

Icinga 2, Endpoint

2

endpoint'а на зону — максимум по
докам; ставим ровно два в master и
критичных сателлит-зонах

наш стандарт HA

Ядро: мастер-зона с Icinga DB и веб-консолью

Что настраиваем

Центральный узел в ЦОД: *icinga2 + Redis + MariaDB/PostgreSQL + Icinga Web 2*, при HA — второй *endpoint* в зоне *master*

Как мы это делаем

- 1 Подключаем репозиторий `packages.icinga.com`, ставим `icinga2`, `icingadb`, `icingadb-redis`, `icingaweb2` и `icingadb-web` — без устаревшего `ido-mysql`
- 2 `icinga2 node setup --master`: генерирует CA в `/var/lib/icinga2/ca` и `ApiListener` на `5665/tcp`; в `constants.conf` задаём `NodeName`, `ZoneName` и случайный `TicketSalt`
- 3 `icinga2 feature enable icingadb api`; демон `icingadb` перекладывает состояния из Redis в SQL — история и SLA живут в реляционной базе
- 4 Для HA добавляем второй `endpoint` в зону `master` с общей БД — фичи `checker` и `notification` сами балансируются между узлами пары
- 5 Перед каждым перезапуском — `icinga2 daemon -C`: рестарт с битым конфигом не роняет мониторинг

РЕЗУЛЬТАТ

Одна консоль Icinga DB Web на всю инфраструктуру: состояния, история, downtimes, подтверждения; отказ одного узла мастер-пары не останавливает ни проверки, ни уведомления.

КЛЮЧЕВОЙ НЮАНС

С 2.15 строим только на Icinga DB: IDO помечен `deprecated` и удаляется в 2.18 — миграция «потом» означает переделку истории и отчётов заново.



Площадки: сателлиты, агенты и синхронизация зон

Что настраиваем

Каждый филиал — своя зона-сателлит с `parent = master`; агенты на Windows/Linux-хостах для проверок изнутри

Как мы это делаем

- 1 На сателлите `icinga2 node wizard`: CN узла, `parent endpoint` мастера, тикет с мастера (`icinga2 pki ticket --cn <fqdn>`), включаем `accept_config` и `accept_commands`
- 2 На мастере в `zones.conf` описываем `Endpoint` и `Zone` филиала; объекты кладём только в `/etc/icinga2/zones.d/<зона>` — синхронизация идёт сверху вниз автоматически
- 3 Общие `CheckCommand` и шаблоны выносим в глобальные зоны `global-templates` и `director-global`
- 4 Агенты исполняют проверки через `command_endpoint = host.name`: диски, службы Windows, процессы — без NRPE и открытых портов внутри
- 5 Связность зон контролируем сервисом `cluster-zone` на мастере — обрыв канала виден отдельным алертом, а не ложными DOWN всех хостов филиала

РЕЗУЛЬТАТ

При обрыве VPN сателлит продолжает проверки по локальному расписанию, копит результаты в `replay`-лог (`log_duration 1d`) и досылает историю после восстановления — SLA-отчёты остаются честными.

КЛЮЧЕВОЙ НЮАНС

Иерархию глубже `master` → `satellite` → `agent` не строим — по докам она не поддерживается; на сателлитах руками не правим ничего: синхронизация затрёт локальные изменения.



Director: шаблоны, импорт из AD и деплой

Что настраиваем

Модуль Icinga Director 1.11 на мастере: вся конфигурация через UI с валидацией и историей деплоев

Как мы это делаем

- 1 Ставим модуль, БД director (utf8), заводим ApiUser с полными правами на мастере; kickstart-визард подтягивает endpoint'ы, зоны и команды
- 2 Заводим шаблоны хостов tpl-host-linux / tpl-host-windows / tpl-host-network и наборы сервисов: диск, CPU, службы, HTTP/HTTPS, SMTP, SNMP-порты
- 3 Import Source по LDAP из Active Directory + Sync Rule: новые серверы попадают в мониторинг сами, apply-правила вешают сервисы по vars.os и роли
- 4 Уведомления двумя каналами: email всем, Telegram-скрипт как NotificationCommand для vars.priority == critical с эскалацией
- 5 Каждый деплой Director валидируется до применения; откат — в один клик из истории конфигураций

РЕЗУЛЬТАТ

Инженер добавляет сервер в AD — тот попадает в мониторинг с нужным набором проверок без ручной работы; изменения конфигурации проходят валидацию и версионизируются, «сломать всё правкой конфига» больше нельзя.

КЛЮЧЕВОЙ НЮАНС

После внедрения Director вся конфигурация живёт в нём: параллельные ручные файлы в zones.d создают конфликты объектов — миксовать источники нельзя.



Подводные камни

✗ Новый мониторинг на IDO

ido-mysql устарел и удаляется в 2.18: ставим icingadb + icingadb-redis сразу, чтобы через год не мигрировать историю и отчёты

✗ Правка конфига на сателлите

Синхронизация зон идёт сверху вниз и затирает локальные файлы; все объекты — только в zones.d мастера или в Director

✗ Больше двух endpoint'ов в зоне

Зона поддерживает максимум 2 узла HA; третий создаёт петлю маршрутизации сообщений (валидация конфига предупреждает) — масштабируем новыми зонами-сат...

✗ Проверки филиалов с мастера по VPN

Потери на канале дают ложные DOWN; проверки исполняет сателлит на месте, мастер получает только результаты по 5665/tcp

✗ Забытый TicketSalt и висящие CSR

Без тикета запросы сертификатов ждут ручной подписи; либо генерим тикеты `pki ticket --cn`, либо подписываем `icinga2 ca list / ca sign`

✗ Нет контроля связности зон

Без сервиса cluster-zone обрыв линка выглядит как массовый DOWN хостов; отдельная проверка зоны режет шторм алертов до одного

✗ Рассинхрон времени узлов

Дрейф часов ломает валидацию TLS-сертификатов и порядок replay-событий; NTP (chrony) — обязательный пункт на каждом узле кластера

✗ СА «на 15 лет» без календаря

Сертификаты узлов обновляются сами, а корневой СА — нет; фиксируем дату и процедуру ротации СА заранее, а не при отказе соединений

Как правильно

МИНИМУМ

- Один мастер Icinga 2.15+ на Icinga DB, агенты на критичных серверах
- Базовые проверки: ping, диски (-w 20% -с 10%), службы, HTTP/HTTPS
- Email-уведомления Problem/Recovery с рабочим отправителем
- icinga2 daemon -C перед рестартом, бэкап /etc/icinga2 и CA

НОРМАЛЬНО

- Сателлит в каждом филиале, проверки исполняются локально
- Director: шаблоны хостов и сервисов, деплой с валидацией
- Отдельный Telegram-канал для critical с эскалацией
- Сервис cluster-zone на связность каждой зоны

ХОРОШО

- HA-пары (2 endpoint'a) в master и критичных зонах, общая БД
- Импорт хостов из AD по LDAP, автоназначение сервисов apply-правилами
- Git-бэкап конфигурации Director + дампы БД, проверенное восстановление
- SLA-отчёты по Icinga DB и разбор ложных срабатываний раз в месяц



Чек-лист самопроверки

- | | |
|--|---|
| ☐ Мониторинг работает на Icinga DB, а не на устаревшем IDO, который удалят в 2.18? | ☐ Обрыв связи с зоной виден как один алерт cluster-zone, а не шторм DOWN по всем хостам? |
| ☐ В критичных зонах по два endpoint'a — отказ узла мониторинга переживается без слепых пятен? | ☐ Критичные алерты идут отдельным каналом (Telegram) с эскалацией, а не только на email? |
| ☐ Филиал при обрыве канала продолжает проверки и досылает историю через replay-лог? | ☐ Время на всех узлах кластера синхронизировано по NTP? |
| ☐ Сертификаты узлов перевыпускаются автоматически, дата ротации корневого CA зафиксирована? | ☐ Восстановление из бэкапа /etc/icinga2 и БД Director реально проверялось? |
| ☐ Вся конфигурация — в zones.d мастера или Director, на сателлитах руками ничего не правится? | |

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



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).

- Аудит текущего мониторинга и проект зон под ваши площадки: мастер, сателлиты, агенты, НА
- Разворачиваем под ключ: Icinga 2 + Icinga DB + Director, агенты Windows/Linux, импорт из AD
- Настраиваем уведомления и эскалации: email, Telegram, дежурные группы по приоритетам
- Переносим существующий Zabbix/Nagios-мониторинг в Icinga без потери покрытия
- Сопровождаем: обновления версий, ротация сертификатов, бэкапы конфигурации и БД

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



Техническая база

- 01** Icinga 2: Distributed Monitoring (зоны, сателлиты, агенты, сертификаты) (icinga.com — v2.16)
- 02** Icinga 2: Upgrading — депрекация IDO (с 2.14), удаление в 2.18 (icinga.com — v2.16)
- 03** Icinga DB: архитектура Redis + SQL, установка (icinga.com — 1.4)
- 04** Icinga Director: Getting Started, Import/Sync (icinga.com — 1.11)
- 05** Icinga 2: Monitoring Basics — apply-правила, шаблоны, интервалы (icinga.com — v2.16)
- 06** Наш шаблон зон и уведомлений для филиальных сетей (itfresh.ru — 2026)

