

Эксплуатационная документация (v 0.6)

Astra Monitoring

Table of Contents

1	1. Обзор Платформы AM	10
1.1	Глоссарий	10
1.2	Введение	11
1.3	Архитектура Astra Monitoring	12
1.4	Клиентская часть Платформы Мониторинга.....	13
1.4.1	Объект мониторинга	13
1.4.2	Интерфейс: Агент	13
1.4.3	Другие интерфейсы	14
1.4.4	Коллектор для диагностической информации	14
1.5	Центральная часть Платформы Мониторинга.....	15
1.5.1	Получение данных	15
1.5.2	Хранение данных.....	15
1.5.2.1	Victoria Metrics: хранение метрик	15
1.5.2.2	ClickHouse: хранение логов и сигналов.....	16
1.5.2.3	PostgreSQL: хранение внутренних сущностей.....	16
1.5.3	Настройка агентов и коллекторов	16
1.5.4	Состояния наблюдаемых объектов	17
1.5.5	События о наблюдаемых объектах.....	17
1.5.6	Пользовательский интерфейс.....	17
2	2. Подготовка к установке	18
2.1	Системные требования.....	18
2.2	Программные требования	20
2.3	Сетевые настройки	21
2.4	Требования к учётной записи.....	24
3	3. Установка и обновление	26
3.1	Установка и обновление Astra Monitoring с использованием Docker Compose	26

3.1.1	Закрытый контур. Настройка проксирования	27
3.1.1.1	Squid конфигурация прокси сервер	27
3.1.1.2	Nexus менеджер репозитория	28
3.1.1.3	Artifactory менеджер репозитория	29
3.1.2	Закрытый контур. Без возможности проксирования	30
3.2	Установка и обновление Astra Monitoring в кластер Kubernetes (Helm Chart)	32
3.3	Установка базы данных ClickHouse	40
3.4	Установка и обновление клиентской части Astra Monitoring	40
3.4.1	Агент / Коллектор	41
3.4.2	Установка	42
3.4.3	Быстрый старт	43
3.4.3.1	Пример стандартного базового конфигурационного файла Коллектора (Внимание! Необходимо заменить значение "remote_address" на своё):	43
3.4.3.2	Пример стандартного конфигурационного файла Агента:	44
3.4.4	Конфигурирование	46
3.4.4.1	Варианты запуска	46
3.4.4.2	Алгоритм запуска	46
3.4.4.3	Список всех доступных параметров для конфигурации агента/коллектора	47
3.4.5	Экспортёры доступные для установки через агент	55
3.5	Установка внешних экспортёров для продуктов астры	63
3.5.1	Установка клиентской части для ALD Pro / FreeIPA	63
3.5.2	Установка клиентской части для ПК СВ Брест	65
3.5.3	Установка клиентской части для RuPost	68
3.5.4	Установка клиентской части для RuBackup	72
3.5.5	Установка клиентской части для Termidesk	74
3.5.6	Установка клиентской части для Billmanager	76
3.6	Установка внешних компонентов удаленного мониторинга	78

3.6.1	Установка snmp-exporter	78
3.6.2	Установка ipmi-exporter	79
3.6.3	Установка ssl-exporter.....	81
3.6.4	Установка SNMP Агента	83
3.6.5	Установка Vector	84
3.6.5.1	Обзор конфигурационного файла Vector	86
3.6.5.2	Особенности конфигурации Vector в am-agent	86
3.6.6	Установка node-exporter	89
3.6.7	Установка windows-exporter.....	89
3.6.8	Установка systemd-exporter	89
3.6.9	Установка postgres-exporter.....	90
3.6.10	Установка sql-exporter	91
3.6.11	Установка script_exporter	92
4	4. Настройка Платформы AM	93
4.1	Вход в Платформу	93
4.2	Настройка интеграции с LDAP	96
4.3	Добавление объектов мониторинга	96
4.4	Настройка триггеров для создания событий.....	104
4.5	Настройка оповещений.....	106
4.6	Настройка (добавление) дополнительных дашбордов Grafana	109
4.7	Использование API.....	110
5	5. Резервное копирование	114
5.1	Резервное копирование базы данных ClickHouse	114
5.2	Резервное копирование базы данных PostgreSQL	115
6	6. Мониторинг продуктов	116
6.1	6.1 Мониторинг ОС Astra Linux (node-exporter)	116
6.2	6.2 Мониторинг Windows Server	117
6.3	6.3 Мониторинг PostgreSQL	134
6.4	6.4 Мониторинг FreeIPA	134

6.5	6.5 Мониторинг ALD Pro	135
6.5.1	Используемые метрики	135
6.5.2	Конфигурирование Vector для сбора логов по продукту	136
6.6	6.6 Мониторинг ПК СВ Брест	138
6.6.1	Используемые метрики	138
6.6.2	Конфигурирование Vector для сбора логов по продукту	138
6.7	6.7 Мониторинг RuPost	145
6.7.1	Используемые метрики	145
6.7.2	Конфигурирование Vector для сбора логов по продукту	145
6.8	6.8 Мониторинг RuBackup	146
6.8.1	Используемые метрики	146
6.8.2	Конфигурирование Vector для сбора логов по продукту	146
6.9	6.9 Мониторинг Termidesk	147
6.9.1	Используемые метрики	147
6.9.2	Конфигурирование Vector для сбора логов по продукту	147
6.10	6.10 Мониторинг Billmanager	149
6.10.1	Используемые метрики	149
6.11	6.10 Триггеры для создания событий по логам	149
6.11.1	6.10.1 Сводный список собираемых логов для AIC	150
6.12	6.11 Мониторинг Astra Kube	152
6.13	6.12 Мониторинг SNMP устройств (агент мониторинга)	154
6.13.1	SNMP поллинг	154
6.13.2	Запуск snmp-exporter с помощью агента мониторинга	154
6.14	6.13 Мониторинг устройств с помощью IPMI (агент мониторинга)	155
6.14.1	IPMI мониторинг	155
6.14.2	Запуск ipmi-exporter с помощью агента мониторинга	156
6.15	6.14 Базовые правила для создания событий	158
6.15.1	6.14.1 Сводные правила оповещений для AIC	160
6.16	6.15 Мониторинг SSL сертификатов (агент мониторинга)	185

6.16.1	Ключи запуска ssl-exporter.....	185
6.16.2	Метрики.....	186
6.16.3	Запуск ssl-exporter с помощью агента мониторинга.....	188
6.16.4	Настройка конфигурационного файла	189
6.17	6.16 Запуск и мониторинг локальных скриптов (агент мониторинга) ...	190
6.17.1	Ключи запуска script-exporter	190
6.17.2	Метрики.....	191
6.17.3	Запуск script-exporter с помощью агента мониторинга	192
6.17.4	Пример запуска кастомного скрипта.....	193
6.17.5	Особенность запуска на Windows	194
6.18	6.17 Мониторинг SNMP трапов	194
6.18.1	Введение	194
6.18.2	Header - подготовка переменных.....	195
6.18.3	Lookup - таблицы приведения	196
6.18.4	Preparation - подготовка данных	196
6.18.5	Body - определение трапов и извлечение необходимых данных для создания события	197
6.18.6	Footer - подготовка данных для передачи в хранимую процедуру	201
6.18.7	Алерты	201
6.18.8	Доступные параметры в файле правил	202
6.18.9	Примечания.....	203
6.19	6.18 Приложения. Метрики, используемые для мониторинга.....	204
6.19.1	1. Метрики, предоставляемые windows-exporter.....	204
6.19.2	2. Метрики, используемые для мониторинга FreeIPA.....	208
6.19.3	3. Метрики, используемые для мониторинга ALD Pro	219
6.19.4	4. Метрики, используемые для мониторинга ПК СВ "Брест"	242
6.19.5	5. Метрики, используемые для мониторинга RuPost	278
6.19.6	6. Метрики, используемые для мониторинга RuBackup	294
6.19.7	7. Метрики, используемые для мониторинга Termidesk	299

6.19.8	8. Метрики, используемые для мониторинга Billmanager	331
7	7. Пользовательский интерфейс	334
7.1	Пользовательское окно "Проблемы"	334
7.1.1	Общая информация	334
7.1.2	Основная таблица с проблемами.....	335
7.1.3	Правая боковая панель.....	336
7.1.4	Контекстный переход на страницу "События"	338
7.1.5	Дополнительные элементы	338
7.2	Пользовательское окно "События"	341
7.2.1	Общая информация	341
7.2.2	Основная таблица с событиями	342
7.2.3	Правая боковая панель.....	343
7.2.4	История изменения событий.....	345
7.2.5	Дополнительные элементы	346
7.3	Пользовательское окно "Объекты"	347
7.3.1	Общая информация	347
7.3.2	Основная таблица с объектами.....	348
7.3.3	Дополнительные элементы	348
7.3.4	Форма для создания/редактирования/клонирования объекта.....	349
7.4	Пользовательское окно "Интерфейсы объектов"	357
7.4.1	Общая информация	357
7.4.2	Основная таблица с интерфейсами	358
7.4.3	Дополнительные элементы	358
7.5	Пользовательское окно "Коллекторы"	359
7.5.1	Общая информация	359
7.5.2	Основной список карточек коллекторов.....	359
7.5.3	Дополнительные элементы	361
7.6	Пользовательское окно "Теги"	361
7.6.1	Общая информация	361

7.6.2	Основная таблица с тегами.....	362
7.6.3	Дополнительные элементы.....	363

- 1. Обзор Платформы AM (see page 10)
- 2. Подготовка к установке (see page 18)
- 3. Установка и обновление (see page 26)
- 4. Настройка Платформы AM (see page 93)
- 5. Резервное копирование (see page 114)
- 6. Мониторинг продуктов (see page 116)
 - 6.1 Мониторинг ОС Astra Linux (node-exporter) (see page 116)
 - 6.2 Мониторинг Windows Server (see page 117)
 - 6.3 Мониторинг PostgreSQL (see page 134)
 - 6.4 Мониторинг FreeIPA (see page 134)
 - 6.5 Мониторинг ALD Pro (see page 135)
 - 6.6 Мониторинг ПК СВ Брест (see page 138)
 - 6.7 Мониторинг RuPost (see page 145)
 - 6.8 Мониторинг RuBackup (see page 146)
 - 6.9 Мониторинг Termidesk (see page 147)
 - 6.10 Мониторинг Billmanager (see page 149)
 - 6.10 Триггеры для создания событий по логам (see page 149)
 - 6.10.1 Сводный список собираемых логов для AIC (see page 150)
 - 6.11 Мониторинг Astra Kube (see page 152)
 - 6.12 Мониторинг SNMP устройств (агент мониторинга) (see page 154)
 - 6.13 Мониторинг устройств с помощью IPMI (агент мониторинга) (see page 155)
 - 6.14 Базовые правила для создания событий (see page 158)
 - 6.14.1 Сводные правила оповещений для AIC (see page 160)
 - 6.15 Мониторинг SSL сертификатов (агент мониторинга) (see page 185)
 - 6.16 Запуск и мониторинг локальных скриптов (агент мониторинга) (see page 190)
 - 6.17 Мониторинг SNMP трапов (see page 194)
 - 6.18 Приложения. Метрики, используемые для мониторинга (see page 204)
 - 1. Метрики, предоставляемые windows-exporter (see page 204)
 - 2. Метрики, используемые для мониторинга FreeIPA (see page 208)
 - 3. Метрики, используемые для мониторинга ALD Pro (see page 219)
 - 4. Метрики, используемые для мониторинга ПК СВ "Брест" (see page 242)
 - 5. Метрики, используемые для мониторинга RuPost (see page 278)
 - 6. Метрики, используемые для мониторинга RuBackup (see page 294)
 - 7. Метрики, используемые для мониторинга Termidesk (see page 299)
 - 8. Метрики, используемые для мониторинга Billmanager (see page 331)
- 7. Пользовательский интерфейс (see page 334)

1 1. Обзор Платформы АМ

1.1 Глоссарий

	Термин/сокращение	Описание
1	Агент/Agent	Компонента Платформы мониторинга, отвечающее за сбор и предоставление диагностической информации в рамках отдельного сервера/хоста
2	АМ, Платформа мониторинга, Платформа АМ/Astra Monitoring	Программная платформа "Астра Мониторинг"
3	Диагностическая информация/Monitoring data	Данные, собираемые с узлов IT инфраструктуры, которые затем используются в платформе мониторинга для контроля состояния и обнаружения проблем; другими словами – метрики, логи и сигналы
4	Интерфейс/Interface	Описание, как и какие данные собирать с объекта мониторинга, например правила парсинга SNMP или ip:port адрес сервера, предоставляющего метрики
5	Коллектор/Collector	Особый режим запуска агента, позволяющий управлять сбором диагностической информации на уровне подсети/ локальной сети
6	Компонента/Component	«Класс» приложений, взаимодействующих с АМ Backend (например, агент или коллектор)
7	Лейбл/Label	Произвольная пара Ключ:Значение с дополнительной информацией о сообщении, например hostname: 127.0.0.1
8	Лог/Logs	Текстовые данные, создаваемые запущенными приложениями для информирования о своей работе, например http server started at port 6344

	Термин/сокращение	Описание
9	Метрики/Metrics	Prometheus-метрики, данные вида Ключ:Значение, создаваемые преимущественно экспортерами, например http_total_requests 10
10	Нода/Node	Узел/сервер/хост, на котором расположен объект мониторинга
11	Объект мониторинга/ Monitoring object	Источник диагностической информации. Объектом мониторинга может быть как физический объект, например, сервер или маршрутизатор, так и логический элемент системы, например, web-сервис. Для сбора данных с объекта используются интерфейсы.
12	Подсеть/Subnet	Совокупность объектов мониторинга из одной локальной сети
13	Сигналы/Signals	Данные в произвольном формате, для которых есть правила парсинга в сообщения, например SNMP-trap; другими словами, это любая диагностическая информация, которая не классифицируется, как метрика или лог
14	Событие/Event	Сущность Платформы мониторинга, которая создается на основе собранной диагностической информации и нужна для оповещения о возникновении определенных состояний в наблюдаемой подсети/локальной сети, например о достижении нодой критического состояния по потреблению ресурсов
15	Состояние/State	То, что происходит с узлами подсети/локальной сети в определенный момент времени; с точки зрения Платформы мониторинга – комбинация диагностической информации, конкретные значения метрик, наличие определенных логов и сигналов
16	Уведомление/Notification	Оповещение в виде текстового сообщения или письма на email, которое сигнализирует о возникновении события

1.2 Введение

Astra Monitoring – платформа сбора и анализа информации о состоянии узлов IT инфраструктуры с целью их диагностики, своевременного обнаружения неполадок и повышения стабильности работы.

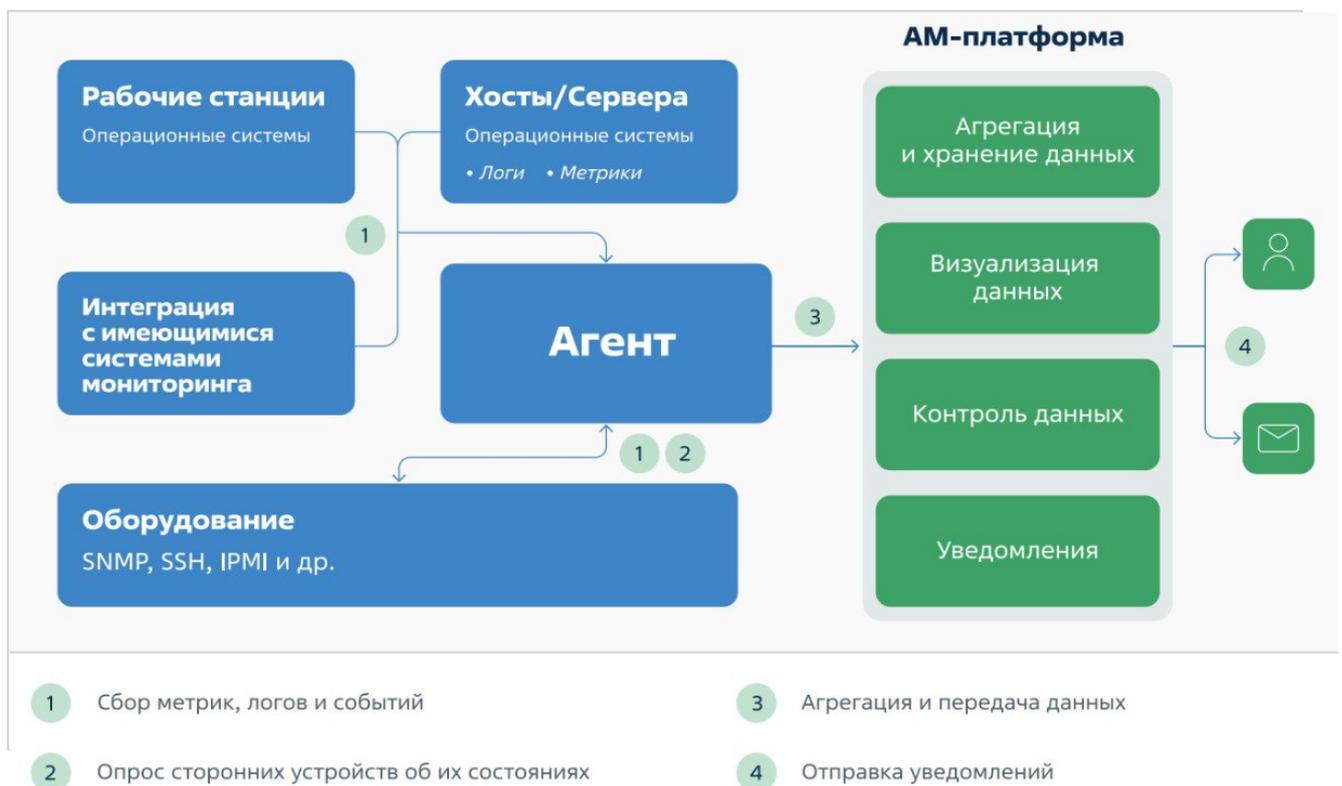
Платформа масштабируема и может охватывать как несколько hardware устройств, например, с SNMP интерфейсом, так и географически распределенные сети с тысячами высоконагруженных нод.

Техническая реализация строится на взаимодействии двух частей – **клиентской**, выполняющий непосредственный сбор диагностической информации, и **центральной**, отвечающей за обработку полученных данных. На клиентской части с объектов мониторинга собираются метрики, логи и сигналы, которые сохраняются в центре обработки и превращаются в события, несущие полезную информацию о возникающих состояниях объектов наблюдения.

Возможности Astra Monitoring:

- Сбор метрик, логов и сигналов с узлов IT инфраструктуры;
- Хранение диагностической информации в центре обработки;
- Отображение собранной информации в панели администратора;
- Создание событий при изменении состояний объектов наблюдения;
- Оповещение пользователей по различным каналам связи.

1.3 Архитектура Astra Monitoring



1.4 Клиентская часть Платформы Мониторинга

1.4.1 Объект мониторинга

Объект мониторинга — источник диагностической информации. Информация о наблюдаемой подсети/локальной сети предоставляется только объектами мониторинга, которые зарегистрированы в АМ. Объект является отображением ноды/узла/сервера/устройства с уникальным FQDN/IP-адресом в рамках подсети/локальной сети. **Для сбора данных с объекта используются его интерфейсы:** сущности, описывающие правила получения и парсинга данных с источника, располагаемого на объекте. Интерфейсов можно определять сколько угодно — главное задать правила парсинга входящего потока байт.

Примеры интерфейсов:

- **agent** — хост/сервер, на котором устанавливается агент — приложение, выполняющее самостоятельную регистрацию в АМ, поддерживающее удаленную настройку, сбор метрик/логов/сигналов с последующей отправкой на коллектор;
- **exporter** — HTTP сервер, предоставляющий метрики в prometheus формате;
- **snmp-trap** — устройство, отправляющее SNMP traps, в интерфейсе описаны правила парсинга приходящего трапа;
- **snmp-poll** — устройство, предоставляющее порт для сбора информации по протоколу SNMP, в интерфейсе описаны правила парсинга;
- **ipmi** — IPMI устройство;
- **http** — произвольные HTTP запросы (обычно с JSON в теле), для которых определены правила парсинга в сообщении.

Каждому объекту может быть назначен только один Коллектор, через который собранная информация передается в АМ.

1.4.2 Интерфейс: Агент

Агент — основное приложение для сбора и предоставления диагностической информации в рамках отдельного сервера/хоста. Технически, агент является оркестратором процессов — он запускает экспортеры и `vmagent` для сбора метрик, `vector` для сбора логов, `Signals Adapter` для сбора сигналов. Агент автоматически регистрируется в АМ, предоставляя информацию о наблюдаемом хосте. Агент состоит из внутренних модулей, каждый из которых используется для своей задачи:

- **Exporter Manager.** Отвечает за запуск экспортеров в отдельных процессах ОС, их контроль и конфигурирование. Экспортер — специальное программное обеспечение для сбора диагностической информации и представления её в виде prometheus метрик. Обычно запускается несколько экспортеров, так как каждый отвечает за сбор информации только о конкретном продукте или части системы. Например, `systemd-exporter` предоставляет метрики только о работе `systemd`, `node-exporter` — общую информацию об ОС, и так далее.

- **Metrics Manager.** Настраивает и запускает [vmagent](https://docs.victoriametrics.com/vmagent/)¹ для сбора метрик с экспортеров. vmagent потребляет мало ресурсов, при этом способен собирать метрики с огромного количества источников, модифицировать лейблы и локально кэшировать собранные данные при недоступности AM Backend. Помимо экспортеров, запущенных Exporter Manager, vmagent на агенте может собирать метрики с любого другого экспортера при соответствующей настройке.
- **Logs Manager.** Настраивает и запускает [vector](https://vector.dev/)² для сбора логов. Он поддерживает большое количество источников и способен трансформировать логи перед отправкой, а также сохранять в локальном буфере при ошибках отправки. Кастомизация и сбор логов с произвольных источников выполняется путем предоставления сторонних файлов конфигурации для vector.
- **Signals Manager.** Настраивает и запускает **Signals Adapter** для получения сигналов. Адаптер использует два сервера – UDP для приема SNMP-traps и HTTP, представляющий API для загрузки сигналов со сторонних клиентов. Сигналы парсятся на основе настраиваемых правил (актуально для SNMP) и отправляются в коллектор.
- **Watcher & Configurator.** Контролирует взаимодействие с Config API, расположенном на AM Backend. Выполняет регистрацию агента при запуске, позволяет удаленно настраивать агент, предоставляет информацию о работоспособности всех его модулей.

1.4.3 Другие интерфейсы

У каждого объекта может быть несколько интерфейсов. Объект считается активным в тот момент, когда на нем регистрируется хотя бы один интерфейс. Агент регистрируется автоматически при запуске, в то время как все другие интерфейсы должны быть добавлены вручную через UI Платформы мониторинга.

1.4.4 Коллектор для диагностической информации

Коллектор – приложение, управляющее сбором диагностической информации на уровне подсети/ локальной сети. Технически, коллектор является агентом, запущенном в специальном режиме. Коллектор принимает данные с агентов и других интерфейсов. Собранная диагностическая информация отправляется в AM Backend для дальнейшей обработки. Для простоты настройки и управления рекомендуется устанавливать только один коллектор в рамках отдельной подсети/ локальной сети.

Как и агент, коллектор состоит из внутренних модулей:

- **Exporter Manager.** Настраивает и запускает [snmp-exporter](https://github.com/prometheus/snmp_exporter)³ для общения с hardware устройствами по SNMP и [ipmi-exporter](https://github.com/prometheus-community/ipmi_exporter)⁴ для сбора метрик с удаленных устройств IPMI.
- **Metrics Manager.** Настраивает и запускает [vmagent](https://docs.victoriametrics.com/vmagent/)⁵ для агрегации метрик с агентов. Способен кэшировать метрики при недоступности AM Backend, используется для добавления лейблов. Обеспечивает возможность наблюдения сторонних объектов мониторинга типа `exporter` и сбора метрик с локальных SNMP и IPMI exporters.

1 <https://docs.victoriametrics.com/vmagent/>

2 <https://vector.dev/>

3 https://github.com/prometheus/snmp_exporter

4 https://github.com/prometheus-community/ipmi_exporter

5 <https://docs.victoriametrics.com/vmagent/>

- **Logs Manager.** Настраивает и запускает [vector](#)⁶ для агрегации логов с агентов. Способен трансформировать проходящие данные и обогащать логи дополнительной информацией, Также ведет сбор логов с подсистем коллектора.
- **Signals Manager.** Настраивает и запускает **Signals Adapter** для получения сигналов. Работает так же, как и на агенте, но дополнительно получает запросы от других адаптеров для агрегации и последующей отправки на AM Backend. Используется для добавления в сигналы недостающих лейблов и приема сигналов на стороне коллектора.
- **Proxy Manager.** Настраивает и запускает [vmauth](#)⁷ для агрегации всех входящих на коллектор запросов от агентов. Каждый агент отправляет запросы к Config API, которые проксируются vmauth на AM Backend. Также агенты присылают диагностическую информацию, которая сначала собирается в vmagent, vector и Signals Adapter на коллекторе, а затем отправляется на AM Backend.
- **Watcher & Configurator.** Контролирует взаимодействие с Config API, расположенном на AM Backend. Выполняет регистрацию при запуске, позволяет удаленно настраивать коллектор, предоставляет информацию о работоспособности всех его модулей.

1.5 Центральная часть Платформы Мониторинга

1.5.1 Получение данных

Для взаимодействия с коллектором AM Backend предоставляет HTTP(S) API. Технически, это API Gateway, который распределяет запросы по нужным микросервисам в зависимости от пути HTTP запроса. AM Backend и коллектор могут располагаться как в одной локальной сети, так и в разных. Коллекторов, работающих с одним AM Backend, может быть несколько — их количество ограничено только доступными ресурсами. В качестве API используется [vmauth](#)⁸ — решение для балансировки нагрузки из экосистемы Victoria Metrics, которое обладает простой установкой и конфигурацией.

1.5.2 Хранение данных

1.5.2.1 Victoria Metrics: хранение метрик

[Victoria Metrics](#)⁹ — NoSQL база данных, специализированная для эффективного хранения метрик и временных рядов. Она используется для полной замены Prometheus из-за явных недостатков последнего при росте объема хранимых данных. VM предоставляет язык запросов [MetricsQL](#)¹⁰, который является расширением языка [PromQL](#)¹¹ и используется для получения и агрегирования метрик. Также эта БД способна горизонтально масштабироваться, что позволяет использовать её для хранения и обработки огромного объема данных.

Метрики в Victoria Metrics попадают напрямую из vmauth.

⁶ <https://vector.dev/>

⁷ <https://docs.victoriametrics.com/vmauth/>

⁸ <https://docs.victoriametrics.com/vmauth/>

⁹ <https://docs.victoriametrics.com/single-server-victoriametrics/>

¹⁰ <https://docs.victoriametrics.com/metricsql/>

¹¹ <https://prometheus.io/docs/prometheus/latest/querying/basics/>

1.5.2.2 ClickHouse: хранение логов и сигналов

[ClickHouse](https://clickhouse.com/)¹² – NoSQL база данных колоночного типа, специализированная для эффективного хранения огромных объемов данных различной природы. Основная сфера применения – OLAP (Online analytical processing) запросы, целью которых является анализ данных, создание отчетов, графиков и дашбордов. База данных имеет свой SQL-подобный язык запросов с широким функционалом, способна горизонтально масштабироваться.

Логи и сигналы сначала попадают в микросервис **ClickHouse Adapter**. Этот сервис предоставляет API для записи данных в ClickHouse – преобразует пришедший JSON в поля SQL запроса и формирует батч.

1.5.2.3 PostgreSQL: хранение внутренних сущностей

[PostgreSQL](https://www.postgresql.org/)¹³ – реляционная база данных, обладает широким функционалом, а главное – поддержкой JSON, что сильно облегчает задачу хранения внутренних сущностей.

Внутренние сущности – данные, которые не относятся непосредственно к диагностической информации. В основном, это конфигурация различных компонент – объектов мониторинга, коллекторов, Event Processor. Также там хранятся события и лейблы.

1.5.3 Настройка агентов и коллекторов

Для удаленной настройки агентов и коллекторов используется сервис Config API. Он хранит данные в PostgreSQL и выполняет следующие функции:

- **Установка.** Для запуска агента ему необходимо предоставить бинарные файлы экспортеров, vmagent, vector, signals adapter. Чтобы не включать эти файлы в пакет поставки и тем самым не увеличивать его объем, агенту дана функция самому скачивать необходимые бинарные файлы с Config API.
- **Регистрация.** Агент отправляет регистрационный запрос в Config API при запуске, передавая информацию о хосте, на котором запущен. Это позволяет автоматически добавлять объекты мониторинга – нужно просто запустить агент на нужной ноде и дальше с ним можно взаимодействовать удаленно.
- **Конфигурация.** Config API предоставляет конфигурацию агентам, позволяя удаленно влиять на процесс сбора диагностической информации. Например, в UI можно указать, какие экспортеры запускать, с каких SNMP-устройств получать трапы и так далее. При обновлении конфигурации создается новая запись в БД; старая версия не удаляется, а помечается исторической – это позволяет отслеживать, как и когда менялась конфигурация.
- **Получение информации.** Config API принимает периодические запросы от агентов о статусе их работы. Агент сообщает информацию о состоянии каждого запущенного экспортера, предоставляет данные об ошибках.

¹² <https://clickhouse.com/>

¹³ <https://www.postgresql.org/>

1.5.4 Состояния наблюдаемых объектов

Astra Monitoring предоставляет два варианта получения информации о наблюдаемом объекте – состояние и события. Состояние представляет из себя диагностическую информацию в «сыром» виде, в то время как события – сгенерированные на основе этой информации сущности, необходимые для оповещения о наступлении определенных состояний.

1.5.5 События о наблюдаемых объектах

События создаются на основе приходящей в АМ диагностической информации. Смысл событий – выделить из большого потока данных состояния, имеющие ценность в контексте наблюдения за жизнеспособностью объектов. Они могут быть как с положительным окрасом – сообщать о запуске сервиса, успешно проведенной миграции или завершеном автотестировании нового релиза, так и с отрицательным – сигнализировать о высоком потреблении ресурсов, аномально большом количестве запросов или о падении значений бизнесовых метрик.

1.5.6 Пользовательский интерфейс

Компоненты:

- Модуль визуализации метрик и логов построен на базе программного продукта Grafana, представляет из себя набор представлений данных и интерфейс анализа логов.
- Интерфейс управления - Admin UI. Предназначен для добавления объектов мониторинга, а также для просмотра информации о событиях по объектам мониторинга.
- Keycloak. Обеспечивает аутентификацию пользователей, поддерживает интеграцию с внешними системами аутентификации и каталогами пользователей (LDAP).

Пользовательский интерфейс позволяет визуализировать собранные данные, отображать метрики в виде индикаторов и графиков, выделять и представлять пользователю информацию об обнаруженных на объектах мониторинга проблемах, добавлять объекты мониторинга в Платформу АМ или удалять их и т.д.

2 2. Подготовка к установке

Платформа Astra Monitoring может быть развернута в нескольких вариантах:

Размер установки	Способ распространения	Ограничения
Small	1 сервер, Docker compose/ Однонодовый кластер Kubernetes	Нет отказоустойчивости на уровне приложения Низкая производительность из-за совмещения компонент
Medium	2 сервера, Docker compose / Однонодовый кластер Kubernetes + Сервер БД	Нет отказоустойчивости на уровне приложения Производительность средняя из-за вынесения роли БД на отдельный сервер
Large	3+ нодовый кластер Kubernetes	Отказоустойчивость на уровне кластера Производительность выше среднего



Примечание к релизу

Конфигурация Medium не покрывается этой инструкцией и будет дополнена в новых релизах.

2.1 Системные требования

Платформа Astra Monitoring может быть установлена в кластер Kubernetes или используя Docker Compose сборку.

Потребность в ресурсах:

- Requests (Memory): 3460Mi (\approx 3.67 GiB),
- Requests (CPU): 2400m,
- Limits (Memory): 14220Mi (\approx 14.22GiB),
- Limits (CPU): 11000m,

- Minimum 200GB SSD

Сервис	Подсервис	Requests (Memory)	Requests (CPU)	Limits (Memory)	Limits (CPU)
clickhouse	-	1500Mi	200m	5Gi (5120Mi)	500m
clickhouse	updatejob	100Mi	100m	200Mi	200m
clickhouse	backupjob	300Mi	300m	2000Mi	1000m
clickhousekeeper	-	200Mi	100m	400Mi	200m
clickhouseproxy	-	100Mi	50m	200Mi	100m
postgresql	-	200Mi	200m	400Mi	4000m
postgresql	updatejob	100Mi	100m	200Mi	200m
postgresql	backupjob	100Mi	100m	200Mi	200m
admin_ui	front	10Mi	100m	100Mi	200m
admin_ui	back	10Mi	100m	200Mi	200m
alert_manager	-	100Mi	50m	300Mi	300m
config_api	-	10Mi	100m	100Mi	200m
mongodb	-	100Mi	100m	500Mi	2000m
event_adapter	-	10Mi	100m	100Mi	200m
prometheus	-	120Mi	100m	1500Mi	300m
victoria	-	500Mi	300m	2000Mi	500m
victoria	vmalert	100Mi	100m	200Mi	200m

Сервис	Подсервис	Requests (Memory)	Requests (CPU)	Limits (Memory)	Limits (CPU)
victoria	vmauth	100Mi	100m	200Mi	200m
vector	-	100Mi	100m	300Mi	300m

Минимальные требования для запуска с использованием Docker Compose:

- 4 vCPU с поддержкой SSE3 (режим host-passthrough или аналогичный),
- 8GB RAM,
- 200GB SSD.

2.2 Программные требования

- При разворачивании в кластере Kubernetes должны быть выполнены следующие условия:

1. Kubernetes cluster - версия не ниже v1.23.5 (протестирована работа на версиях v1.23.5 - локальная установка и v1.23.6 - Yandex Managed Kubernetes),
2. Установленный Ingress Nginx для публикации ресурсов,
3. Настроенный Persistent Storage (для Clickhouse инстансов - SSD Storage, для остального можно HDD). В случае Yandex Managed Kubernetes можно использовать uc-network-hdd\uc-network-ssd\uc-network-nvme для дисков БД (Clickhouse, PostgreSQL) и csi-s3 для бэкапов Clickhouse (т.е. требуется ReadWriteMany для CronJob для бэкапа),
4. Установленный Helm v3,
5. Установленный Reloader¹⁴ для перезапуска приложения при изменении конфигурации компонентов,
6. Выделенный Namespace для запуска приложения.

- При разворачивании как Docker Compose должны быть выполнены следующие условия:

1. Версия докера, начиная с 24.0.6,
2. Убедиться, что необходимые порты не заняты другими приложениями на сервере:

Компонент	Порт
vector	9102/TCP

¹⁴ <https://github.com/stakater/Reloader>

Компонент	Порт
alertmanager	9093/TCP
grafana	3000/TCP
prometheus	9090/TCP
victoria metrics	8428/TCP
vmalert	8880/TCP
gatekeeper	80/TCP
admin-ui	-
admin-back	8000/TCP
keycloak	8081/TCP

2.3 Сетевые настройки

Должны быть настроены:

- Сетевая связность до хранилища Docker образов и Helm charts (<https://registry.astralinux.ru/> , <https://dl.astralinux.ru/>).



Примечание к релизу

Установка в закрытом контуре будет прорабатываться в следующих релизах

- Сетевая связность сервера мониторинга и объектов мониторинга.
- Открытые на сетевых экранах и доступные для сервера Платформы мониторинга следующие сетевые потоки:

Для Агента необходимо предусмотреть доступность следующих ресурсов

Источник (source)	Назначение (destination)	Порт*	Протокол	Экспортер	Комментарий
Платформа мониторинга (сервер сбора данных)	registry.astralinux.ru ¹⁵	443	HTTPS		Доступ до опубликованных Docker образов
	dl.astralinux.ru ¹⁶	443	HTTPS		Доступ до опубликованных Helm чартов и экспортеров
	im.astralinux.ru ¹⁷	443	HTTPS		Доступ к корпоративному Mattermost (опционально)
	Все объекты мониторинга	9100	TCP	node-exporter	Сбор системных метрик
	Контроллеры FreeIPA/ALD-Pro	9888	TCP	freeipa-exporter	Сбор метрик FreeIPA/ALD-Pro
	Все объекты мониторинга	9558	TCP	systemd-exporter	Сбор метрик сервисов systemd
	Ноды Бреста/ OpenNebula	9177	TCP	libvirt-exporter	Сбор метрик системы виртуализации
	Хост для мониторинга Бреста/ OpenNebula	9621	TCP	one-exporter	Сбор метрик через API Бреста/ OpenNebula. Может быть одна из нод Бреста/ OpenNebula либо выделенный сервер
	Mail хосты RuPost	9777	TCP	rupost-exporter	Сбор метрик RuPost

15 <http://registry.astralinux.ru>16 <http://dl.astralinux.ru>17 <http://im.astralinux.ru>

Источник (source)	Назначение (destination)	Порт*	Протокол	Экспортер	Комментарий
	Mail хосты RuPost	8000	TCP	haproxy-exporter	Сбор метрик Haproxy, компонента RuPost
	Mail хосты RuPost	9900	TCP	dovecot-exporter	Сбор метрик Dovecot, компонента RuPost
	Хосты с БД Postgres	9187	TCP	postgres-exporter	Сбор метрик с Postgres DB (стандартные метрики)
	Хосты с БД Postgres	9399	TCP	sql-exporter	Сбор метрик с Postgres DB (кастомные запросы)
	Хосты с RuBackup Server	9444	TCP	rubackup-exporter	Сбор метрик с RuBackup Server
	Хосты с Windows Server	9182	TCP	windows-exporter	Сбор метрик с Windows Server
	Хосты с Termidesk (VDI)	9555	TCP	termidesk-exporter	Сбор метрик с API Termidesk
Все объекты мониторинга	Платформа мониторинга (сервер сбора данных)	30607 (для Kubernetes) 9102 (для Docker)	TCP		Точка доступа для сбора логов с объектов мониторинга

*В зависимости от особенностей среды мониторинга или требований экспортеров, порты могут отличаться от указанных.

- Для мониторинга ПК СВ Брест необходимо обеспечить доступность API Брест (как общего кластерного URL, так и URL менеджмент-серверов) для установленного one-exporter.
- Для мониторинга RuBackup необходимо заранее установить компонент RuBackup REST API на сервер с rubackup-server компонентом.

2.4 Требования к учётной записи

Для запуска Платформы в Kubernetes необходим пользователь с полным доступом до нужного namespace. Для использования утилит kubectl и helm необходимо иметь рабочую конфигурацию .kube/config для подключения к требуемому кластеру Kubernetes.

Для запуска Платформы в Docker Compose необходим пользователь, состоящий в группе `docker`.

На объектах мониторинга Агент для запуска необходимы root права

На объектах мониторинга нужны следующие права, в зависимости от типа экспортера:

1. freeipa-exporter:
 - для установки и запуска сервиса необходимы root права,
 - для получения данных необходима учетная запись с пользовательскими правами на контроллере домена,
 - для получения данных по репликации данная учетная запись должна иметь роль с привилегией на чтение (Read Replication Agreements).
2. one-exporter:
 - для установки и запуска сервиса необходимы root права,
 - для получения данных необходима локальная учетная запись в системе Брест (Драйвер авторизации: соге, нельзя для этого использовать доменную учётную запись - ограничение Брест),
 - для проверки доступности веб-консоли Бреста необходимо создать обычного пользователя на контроллере домена.
3. systemd-exporter:
 - для установки и запуска сервиса необходимы root права.
4. libvirt-exporter:
 - для установки и запуска сервиса необходимы root права.
5. node-exporter:
 - для установки и запуска сервиса необходимы root права.
6. rumpost-exporter:
 - для установки и запуска сервиса необходимы root права.
7. rubackup-exporter:
 - для установки и запуска сервиса необходимы root права.
8. postgres-exporter:
 - для установки и запуска сервиса необходимы root права,
 - для получения данных необходима учетная запись с правами чтения статистики из БД.
9. sql-exporter:
 - для установки и запуска сервиса необходимы root права,
 - для получения данных необходима учетная запись с правами чтения статистики из БД.

10. termidesk-exporter:

- для установки и запуска сервиса необходимы root права,
- для получения данных нужен пользователь с правами в Termidesk.

11. billm-exporter:

- для установки и запуска необходимы root права
- для получения данных необходимо создать пользователя в приложении с правами (Интеграция/Обработчики услуг и полностью Состояние системы)



Примечание к релизу

В будущих релизах Платформы Astra Monitoring требуемые уровни доступов будут пересмотрены

3. Установка и обновление Astra Monitoring

Установка AM осуществляется в кластер Kubernetes или с использованием Docker Compose. Детали установки серверной части предоставляются потенциальному заказчику.

3.1 Установка базы данных ClickHouse

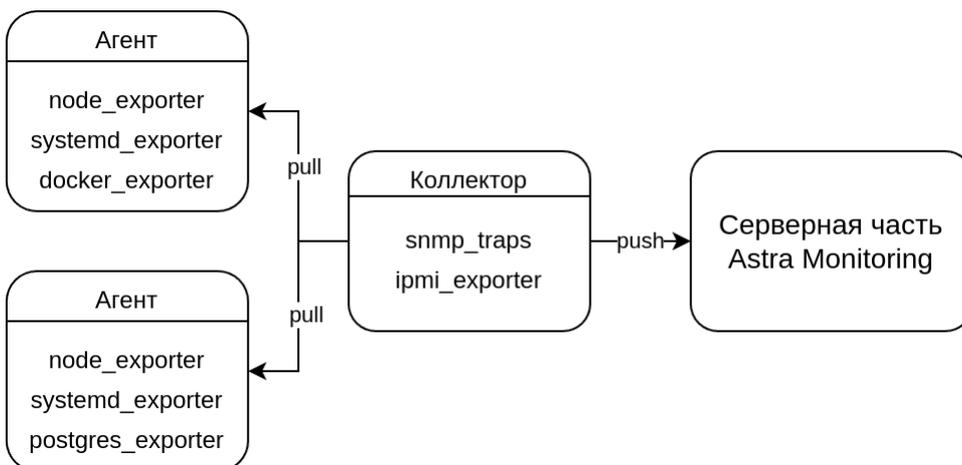
Установка AM в кластер Kubernetes или с использованием Docker Compose включает в себя разворачивание отдельного экземпляра базы данных ClickHouse.
Возможность установки и использования внешней базы данных Clickhouse будет доработана в следующей версии Платформы AM.

3.2 Установка и обновление клиентской части Astra Monitoring

i Примечание к релизу

Здесь и далее установка пакетов описана для Astra Linux 1.7. В будущих релизах документация будет расширена.

Общая схема сбора метрик и логов через агенты/коллекторы Astra Monitoring:



Запуск некоторых элементов зависит друг от друга! Для корректного запуска агента и регистрации его в системе требуется коллектор, для коллектора требуется configari и postgresql.

Таким образом правильная последовательность установки и запуска выглядит так: Серверная часть → Коллектор → Агенты.

3.2.1 Агент / Коллектор

Агент/Коллектор - компонент Платформы мониторинга, написанное на языке программирования Go. Физически представляет собой один и тот же файл, работающий в режиме агент или коллектор в зависимости от конфигурации, по этому в контекста установки далее они будут употребляться как равносильные значения.

Агент – конечный компонент Платформы на узлах мониторинга, основные задачи:

- **Сбор метрик и отправка их в Платформу Astra Monitoring.** Метрики представляют собой различные параметры ОС и процессов. Так как на разных объектах мониторинга могут быть запущены разные сервисы, то и собираемые метрики могут отличаться. Для достижения подобной гибкости используются экспортеры, которые создаются для предоставления метрик с конкретного сервиса или бизнес-приложения. Для самого сбора метрик и отправки их в Платформу мониторинга используется [vmagent](#)²⁸. Он также решает проблему кэширования метрик при невозможности их отправки.
- **Сбор логов и отправка их в Платформу Astra Monitoring.** На разных машинах источники логов могут различаться. Для сбора логов используется [vector](#)²⁹.
- **Управление экспортёрами** с которых будет осуществляться **сбор метрик и отправка их в Платформу Astra Monitoring.** Возможны два сценария работы с экспортёрами: 1) Экспортёры устанавливаются, конфигурируются запускаются сторонними средствами, тогда сбор с них метрик будет осуществляться как со "стороннего экспортёра" с указанием точки сбора метрик. 2) Экспортёры устанавливаются и конфигурируются через Агент, в данной ситуации будет доступно конфигурирование экспортёров через конфигурационный файл агента или из `ui` пользователя, будут доступны шаблоны настроек и возможность их добавления/удаления. Второй способ является предпочтительным.

Коллектор –промежуточный компонент Платформы между агентом и платформой, основной задачи:

- **Кеширование метрик.** В случае кратковременной потери соединения между компонентами клиентской и серверной части (30 минут по умолчанию), метрики будут кешироваться на коллекторе и не будут потеряны после восстановления соединения.
- **Кеширование логов.** В случае кратковременной потери соединения между компонентами клиентской и серверной части, а также в случае временного переполнения количества запросов на отправку логов, данные будут кешироваться на коллекторе и не будут потеряны после восстановления.
- **Регистрация агентов в платформе.** Для реализации возможностей управления агентом, ему необходима регистрация в сервисе управления конфигурациями. В данном случае, за регистрацию агентов отвечает компонент "коллектор".
- **Сбор не систематизированных метрик и отправка их в Платформу Astra Monitoring.** Коллектор может собирать метрики с компонентов не управляющимися агентами.

²⁸ <https://docs.victoriametrics.com/vmagent/>

²⁹ <https://vector.dev/>

3.2.2 Установка

Коллектор рекомендуется устанавливать ближе к агентам.

```
# Создаём папку для коллектора
sudo mkdir -p /opt/am-collector && sudo chown -R $USER:$USER /opt/am-collector && cd /opt/am-collector

# Скачиваем архив с коллектором
> curl -sLo agent.tgz https://dl.astralinux.ru/am/generic/agent/latest/linux/amd64/agent.tgz

# Распаковываем архив с коллектором
> tar zxvf agent.tgz

# Создаём systemd service для автостарта коллектора
> cat << EOF | sudo tee -a /etc/systemd/system/collector.service
[Unit]
Description=am-collector
Wants=basic.target
After=basic.target network.target
[Service]
WorkingDirectory=/opt/am-collector
ExecStartPre=-/bin/chmod +x /opt/am-collector/agent
ExecStart=/opt/am-collector/agent -c config.yml
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=always
RestartSec=10
[Install]
WantedBy=multi-user.target
EOF

# Добавляем службу коллектора в автостарт
> sudo systemctl enable --now collector.service
```

Агент устанавливается на все хосты.

```
# Создаём папку для агента
sudo mkdir -p /opt/am-agent && sudo chown -R $USER:$USER /opt/am-agent && cd /opt/am-agent

# Скачиваем архив с агентом
> curl -sLo agent.tgz https://dl.astralinux.ru/am/generic/agent/latest/linux/amd64/agent.tgz

# Распаковываем архив с агентом
> tar zxvf agent.tgz
```

```

# Создаём systemd service для автостарта агента
> cat << EOF | sudo tee -a /etc/systemd/system/agent.service
[Unit]
Description=am-agent
Wants=basic.target
After=basic.target network.target
[Service]
WorkingDirectory=/opt/am-agent
ExecStartPre=/bin/chmod +x /opt/am-agent/agent
ExecStart=/opt/am-agent/agent -c config.yml
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=always
RestartSec=10
[Install]
WantedBy=multi-user.target
EOF

# Добавляем службу агента в автозапуск
> sudo systemctl enable --now agent.service

```

3.2.3 Быстрый старт

Конфигурационный файл должен находиться рядом с запускаемым файлом.

По умолчанию это: `/opt/am-collector/config.yml`

3.2.3.1 Пример стандартного базового конфигурационного файла Коллектора (Внимание! Необходимо заменить значение "remote_address" на своё):

```

> cat << EOF | tee -a /opt/am-collector/config.yml
general:
  component: collector # тип агента agent/collector
  mode: manual # тип запуска
  remote_address: https://vmauth.astra-monitoring/ # адрес vmauth серверной части
  astra_monitoring
  insecure: True # проверка https
  load_binaries: True # разрешить автоматическую установку необходимых компонентов и
  экспортёров

watcher:
  force_local: False # локальная конфигурация важнее чем из ui

logs:
  enabled: false # будет доступно в новых релизах

logger:

```

```
level: info # уровень детализации логов
EOF
```



Корректный запуск с этим конфигурационным файлом требует `load_binaries: True`
 Для закрытого контура требуется в ручную поместить бинарные файлы `/opt/am-collector/data/metrics/vmagent` и `/opt/am-collector/data/proxy/vmauth`

Перезапускаем службу коллектора и проверяем корректность запуска.

```
> sudo systemctl restart collector.service
> sudo systemctl status collector.service
```

После выполнения команды проверки статуса должны отсутствовать ошибки и `Active: active (running)`

Конфигурационный файл должен находиться рядом с запускаемым файлом.

По умолчанию это: `/opt/am-agent/config.yml`

3.2.3.2 Пример стандартного конфигурационного файла Агента:

```
> cat << EOF | tee -a /opt/am-agent/config.yml
general:
  component: agent # тип агента agent/collector
  mode: manual # тип запуска
  remote_address: http://collector:9700 # адрес коллектора для подключения. По
у умолчанию порт 9700
  insecure: True # проверка https
  load_binaries: True # разрешить автоматическую установку необходимых компонентов и
экспортёров

watcher:
  config_update_interval: 1m # как часто проверять изменения в локальном конфиге
  instance_send_interval: 1m # как часто проверять изменения в удалённом конфиге
  force_local: True # локальная конфигурация важнее чем из ui

metrics:
  enabled: True
  health_path: ""
  custom_targets:
# При запуске необходимо заменить настройки ниже для модуля metrics на реальные (если
есть кастомные экспортеры)
# - name: custom-exporter
#   target: 127.0.0.1:1235
#   labels:
#     hostname: {name_vm}
```

```

#     group: all

exporters:
  enabled: True
  disable_health: True
  exporters:
# При запуске необходимо заменить настройки ниже для модуля exporters на реальные
# - name: node_exporter
#   start_type: args
#   args: --log.format=json --web.listen-address=127.0.0.1:9100
#   address: 127.0.0.1:9100
#   labels:
#     hostname: {name_vm}
#     group: all

# logs:
#   enabled: true
#   sources:
#     - agent
#     - collector
#     - exporters
#     - vmagent

logger:
  level: info # уровень детализации логов
  file: /var/log/am-agent.log
EOF

```



Корректный запуск с этим конфигурационным файлом требует `load_binaries: True`. Для закрытого контура требуется в ручную поместить бинарные файлы `/opt/am-collector/data/metrics/vmagent` и `/opt/am-collector/data/proxy/vmauth` и `/opt/am-collector/data/exporters/node_exporter`

Перезапускаем службу агента и проверяем корректность запуска.

```

> sudo systemctl restart agent.service
> sudo systemctl status agent.service

```

После выполнения команды проверки статуса должны отсутствовать ошибки и `Active: active (running)`.

После запуска коллектора и агентов, они должны отобразиться в `ui` и стать доступными для конфигурирования из графического интерфейса `astra monitoring`.

Если нужно удалить службы с коллектором и агентом, то:

```

# Останавливаем службы коллектора и агента

```

```

> sudo systemctl stop collector.service
> sudo systemctl stop agent.service

# Удаляем директории с коллектором и агентом
> sudo rm -rf /opt/am-agent /opt/am-collector

# Удаляем службы с коллектором и агентом
> sudo rm /etc/systemd/system/collector.service
> sudo rm /etc/systemd/system/agent.service

# Перезапускаем daemon
> sudo systemctl daemon-reload
> sudo systemctl reset-failed

```

3.2.4 Конфигурирование

3.2.4.1 Варианты запуска

Автоматический: есть доступ к Config API

- запуск с параметров запуска `--remoteAddr` и `--collector` при необходимости
- все бинарники качаются м AM
- при первичной регистрации отдается дефолтный конфиг
- при вторичной выкачивается конфиг из AM

Ручной: конфиг в файле и есть доступ к Config API

- `general.mode: manual`
- `general.load_binaries` задает возможность скачать бинарники с AM
- запуск с настроек из yaml
- при первичной регистрации переданный конфиг сохраняется в AM
- при вторичной выкачивается конфиг из AM

Тестовый: конфиг в файле и нет доступа к Config API

- `general.mode: test`
- запуск с настроек из yaml
- настройки не меняются в процессе работы

3.2.4.2 Алгоритм запуска

В процессе старта происходит обработка предоставленного конфига и проверка наличия бинарного файла в директории. Если файла нет, то либо агент падает (если загрузка не предусмотрена), либо загружает файл.

Автоматический

1. Запускаемся с параметров запуска
2. Инициализируем все менеджеры дефолтными настройками
3. Проверяем наличие файлов для всех менеджеров и запоминаем, не падаем при отсутствии
4. Делаем регистрацию в конфиг апи и получаем дефолтный конфиг
5. Выкачиваем все недостающие бинарники
6. Запускаем менеджеры, вытягиваем новые конфиги и отправляем собранные данные

Ручной

1. Берем настройки из уатл конфига
2. Инициализируем все менеджеры
3. Проверяем наличие файлов для всех менеджеров
 - если нет и скачивание включено - запоминаем и качаем после регистрации
 - если нет и скачивание отключено - падаем
4. Делаем регистрацию в конфиг апи
5. Скачиваем недостающие файлы, если включено
6. Запускаем менеджеры, вытягиваем новые конфиги и отправляем собранные данные

Тестовый

1. Берем настройки из уатл конфига
2. Инициализируем только те менеджеры, которые включены в настройках
3. Проверяем наличие файлов для включенных менеджеров, если чего то нет - падаем
4. После инициализации запускаем менеджеры и отправляем собранные данные на тестовый локальный сервер

3.2.4.3 Список всех доступных параметров для конфигурации агента/коллектора

```
### Основные настройки запуска бинарника
general:
  # Default: обязательный параметр
  # Validate: "agent" | "collector"
  # Description: режим работы бинарника, агент или коллектор
  component: agent

  # Default: обязательный параметр
  # Validate: ^https?:\\\/(www\.)?([-a-zA-Z0-9@:%._\+~#={1,256}(\.|\:|:)[a-zA-Z0-9()
{1,5}|:[0-9]{2,5})/([-a-zA-Z0-9()@:%_\+~#?&/=]*)*$
```

```

# Description: адрес, куда отправлять данные, коллектор для агента, AM для
коллектора
remote_address: http://127.0.0.1:9700

# Default: берется из файла './data/ID_(agent|collector)', который генерируется
автоматически, если его нет
# Validate: ^[0-9][0-9]x0[0-9a-f]{12}$ | (example: "51x0412a8f882c5f")
# Description: идентификатор бинарника, используется в ConfigAPI для конфигурации и
определения привязанного объекта
id: ""

# Default: "./data"
# Validate: ^(\.\.\/)*(\.\/|\/)?[\w-\/]*(\.\w+)*$
# Description: директория с данными бинарника, где хранятся экспортеры, другие
бинарники, их логи и тд
base_directory: ""

# Default: "0.0.0.0:9700" if component == "collector", "" if component ==
"agent" (proxy is disabled for "agent")
# Validate: ^[\w\-\\/:@\.]*:[0-9]{1,5}$
# Description: адрес vmauth на входе в коллектор для получения запросов от агентов
(для агента не используется)
listen_address: ""

# Default: ":9701" if component == "collector", ":9711" if component == "agent"
# Validate: ^[\w\-\\/:@\.]*:[0-9]{1,5}$
# Description: адрес сервера бинарника, который предоставляет информацию о своей
работе по эндпоинту /api/v1/info
server_address: ""

# Default: ""
# Validate: length < 256
# Description: подставляется в хэдер Authorization во всех HTTP запросах
auth_token: ""

# Default: хостнейм текущей машины, полученный от ОС
# Validate: length < 256
# Description: имя хоста бинарника
hostname: ""

# Default: "v1"
# Validate: "v1"
# Description: API версия бэкенда, используемая в формировании пути запроса
api_version: ""

# Default: false
# Description: запустить бинарник в тестовом режиме, он не общается с внешними
сервисами
# и отправляет данные на локальный тестовый сервер
test_mode: false

# Default: true if start from command line, false in all other cases

```

```

# Description: указывает, нужно ли загружать бинарные файлы при старте бинарника из
Config API
load_binaries: false

# Default: false
# Description: не проверять сертификат при HTTPS запросах
insecure: false

### Менеджер метрик, запускает vmagent для сбора метрик с HTTP серверов,
предоставляющих метрики в prometheus формате
metrics:
# Default: true if component == "collector", false in all other cases
# Description: запустить ли vmagent, в коллекторе запускается всегда
enabled: true

# Default: general.base_directory + "/metrics/vmagent"
# Validate: ^(\.\./*)(\./|/)?[\w-/*](\.\w+)*$
# Description: путь до vmagent
path: ""

# Default: general.remote_address + "/api/" + general.api_version + "/write/
metrics"
# Validate: ^https?:\./\.(www\.)?([-a-zA-Z0-9@:%_\~#={1,256}(\.|\:|)[a-zA-Z0-9()
{1,5}|:[0-9]{2,5})/([-a-zA-Z0-9()@:%_\~#?&//=]*)*$
# Description: адрес, куда отправлять метрики
remote_write: ""

# Default: ":9702" if component == "collector", ":9712" if component == "agent"
# Validate: ^[\w\-\:/:@\.]*:[0-9]{1,5}$
# Description: адрес vmagent для получения метрик в push режиме и информации о
работе
server_address: ""

# Default: general.base_directory + "/metrics/vmagent.yaml"
# Validate: ^(\.\./*)(\./|/)?[\w-/*](\.\w+)*$
# Description: путь к файлу, в который записывается конфиг vmagent перед запуском
save_config_path: ""

# Default: general.base_directory + "/metrics/vmagent.log"
# Validate: ^(\.\./*)(\./|/)?[\w-/*](\.\w+)*$
# Description: путь к файлу, в который записывается лог vmagent
# ! Если указать log_file: "", то логирования не будет, так его можно отключить
log_file: ./data/metrics/vmagent.log

# Default: general.base_directory + "/metrics/data/"
# Validate: ^(\.\./*)(\./|/)?[\w-/*](\.\w+)*$
# Description: папка, в которой сохраняются данные vmagent при недоступности remote
write
# ! Если указать cache_directory: "", то кэширования не будет, так его можно
отключить
cache_directory: ./data/metrics/data/

```

```

# Default: "/health"
# Validate: ^(/[w\-\//]*) (\?[\w()@:%_\+.~#?&//=\*]+)?$
# Description: путь, по которому обращаться в metrics.server_address для получения
информации о работе vmagent
# ! Если указать health_path: "", то проверок не будет, так их можно отключить
health_path: /health

# Default: 20s
# Validate: health_interval >= 100ms
# Description: интервал healthcheck проверок vmagent (не рекомендуется ставить
меньше 1 секунды)
health_interval: ""

# Default: latest
# Description: версия для скачивания с Config API при включенном load_binaries
version: ""

# Default: 30s
# Validate: scrape_timeout >= 1s
# Description: таймаут запроса vmagent для получения метрик с экспортера
scrape_timeout: ""

# Default: []
# Description: описания целей наблюдения для vmagent кроме экспортеров из вкладки
конфига exporters
custom_targets: []

### Менеджер логов, запускает vector для сбора логов с внутренних компонент агента/
коллектора и с других источников
logs:
# Default: true if component == "collector", false in all other cases
# Description: запустить ли vector, в коллекторе запускается всегда
enabled: true

# Default: general.base_directory + "/logs/vector"
# Validate: ^(\.\.\/)*(\.\/|\/)?[w-\/]*(\.\w+)*$
# Description: путь до vector
path: ""

# Default: general.remote_address + "/api/" + general.api_version + "/write/logs"
# Validate: ^https?:\//\/(www\.)?([-a-zA-Z0-9@:%_\+.~#={1,256}(\.|\:)[a-zA-Z0-9()
{1,5}]|:[0-9]{2,5})/([-a-zA-Z0-9()@:%_\+.~#?&//=\*])*$
# Description: адрес, куда отправлять логи
remote_write: ""

# Default: ":9703" if component == "collector", ":9713" if component == "agent"
# Validate: ^[\w\-\//:@\.]*: [0-9]{1,5}$
# Description: адрес vector для получения логов в push режиме и информации о работе
server_address: ""

```

```

# Default: general.base_directory + "/logs/vector.yaml"
# Validate: ^(\.\.\/)*(\.\/|\/)?[\w-\/]*(\.\w+)*$
# Description: путь к файлу, в который записывается конфиг vector перед запуском
save_config_path: ""

# Default: general.base_directory + "/logs/vector.log"
# Validate: ^(\.\.\/)*(\.\/|\/)?[\w-\/]*(\.\w+)*$
# Description: путь к файлу, в который записывается лог vector
# ! Если указать log_file: "", то логирования не будет, так его можно отключить
log_file: ./data/logs/vector.log

# Default: general.base_directory + "/logs/data/"
# Validate: ^(\.\.\/)*(\.\/|\/)?[\w-\/]*(\.\w+)*$
# Description: папка, в которой сохраняются данные vector при недоступности remote
write
# ! Если указать cache_directory: "", то кэширования не будет, так его можно
отключить
cache_directory: ./data/logs/data/

# Default: "/health"
# Validate: ^(/[\\w\\-\\/]*)((?[\w()@:%_\\+.~#?&//=\\*]+)?)?$
# Description: путь, по которому обращаться в logs.server_address для получения
информации о работе vector
# ! Если указать health_path: "", то проверок не будет, так их можно отключить
health_path: /health

# Default: 20s
# Validate: health_interval >= 100ms
# Description: интервал healthcheck проверок vector (не рекомендуется ставить
меньше 1 секунды)
health_interval: ""

# Default: latest
# Description: версия для скачивания с Config API при включенном load_binaries
version: ""

# Default: general.base_directory + "/logs/configs/"
# Validate: ^(\.\.\/)*(\.\/|\/)?[\w-\/]*(\.\w+)*$
# Description: папка, куда складывать кастомные конфиги для vector для source ==
"custom"
custom_configs_dir: ""

# Default: "", не используется
# Validate: ^(\.\.\/)*(\.\/|\/)?[\w-\/]*(\.\w+)*$
# Description: файл, куда записывать все логи, собираемые vector
all_logs_file: ""

# Default: false
# Description: выключить требование подтверждения remote write о получении пакета
ЛОГОВ
disable_acknowledgements: true

```

```

# Default: ["agent", "exporters", "vector", "vmagent", "custom"]
# Validate: agent, exporters, vector, vmagent, custom
# Description: список источников, откуда брать логи
# ! Если указать sources: [], то логирования не будет, так его можно отключать без
отключения vector
sources: ["agent", "exporters", "vector", "vmagent", "custom"]

### Менеджер экспортеров, запускает экспортеры для предоставления метрик в prometheus
формате
exporters:
# Default: true if component == "collector", false in all other cases
# Description: запустить ли менеджер экспортеров
enabled: true

# Default: general.base_directory + "/exporters"
# Validate: ^(\.\.\/)*(\.\/|\/)?[\w-\/]*(\.\w+)*$
# Description: путь до папки с экспортерами
directory: ""

# Default: 20s
# Validate: health_interval >= 100ms
# Description: интервал healthcheck проверок экспортеров (не рекомендуется ставить
меньше 1 секунды)
health_interval: ""

# Default: false
# Description: отключить healthcheck проверки экспортеров
disable_health: true

# Default: []
# Description: описание экспортеров, которые нужно запустить
exporters: []

### Менеджер прокси, всегда запускает vmauth на коллекторе для принятия запросов с
множества агентов
proxy:
# Default: general.base_directory + "/proxy/vmauth"
# Validate: ^(\.\.\/)*(\.\/|\/)?[\w-\/]*(\.\w+)*$
# Description: путь до vmauth
path: ""

# Default: general.base_directory + "/proxy/vmauth.yaml"
# Validate: ^(\.\.\/)*(\.\/|\/)?[\w-\/]*(\.\w+)*$
# Description: путь к файлу, в который записывается конфиг vmauth перед запуском
save_config_path: ""

# Default: general.base_directory + "/proxy/vmauth.log"
# Validate: ^(\.\.\/)*(\.\/|\/)?[\w-\/]*(\.\w+)*$
# Description: путь к файлу, в который записывается лог vmauth
# ! Если указать log_file: "", то логирования не будет, так его можно отключить

```

```

log_file: ./data/proxy/vmauth.log

# Default: "/health"
# Validate: ^(/[\\w\\-\\/]*)(\\?[\\w()@:%_\\.~#?&//=\\*]+)?$
# Description: путь, по которому обращаться в general.listen_address для получения
информации о работе vmauth
# ! Если указать health_path: "", то проверок не будет, так их можно отключить
health_path: /health

# Default: 20s
# Validate: health_interval >= 100ms
# Description: интервал healthcheck проверок vector (не рекомендуется ставить
меньше 1 секунды)
health_interval: ""

# Default: latest
# Description: версия для скачивания с Config API при включенном load_binaries
version: ""

# Default: ""
# Validate: ^(\\.\\.\\.)*\\.\\.\\.?[/\\w-]*\\.\\.\\.+$
# Description: путь до файла с TLS сертификатом для включения HTTPS
tls_cert_file: test/tls/cert.pem

# Default: ""
# Validate: ^(\\.\\.\\.)*\\.\\.\\.?[/\\w-]*\\.\\.\\.+$
# Description: путь до файла с TLS ключом для включения HTTPS
tls_key_file: test/tls/key.pem

# Default: ""
# Validate: ^(\\.\\.\\.)*\\.\\.\\.?[/\\w-]*\\.\\.\\.+$
# Description: путь до файла с TLS Root CA для верификации TLS сертификатов бэкенов
tls_ca_file: test/tls/ca.pem

### Взаимодействие с Config API
watcher:
# Default: ${GOOS}
# Validate: "darwin", "linux", "windows"
# Description: ОС для скачивания бинарников с Config API
os: ""

# Default: ${GOARCH}
# Validate: "386", "amd64", "arm", "armv7", "arm64"
# Description: архитектура для скачивания бинарников с Config API
arch: ""

# Default: 4
# Validate: load_concurrency >= 1
# Description: сколько воркеров конкурентно скачивают бинарники
load_concurrency: 4

```

```

# Default: infinite
# Description: сколько делать попыток попытаться зарегистрироваться в Config API
перед падением
register_retry: 0

# Default: 10s
# Validate: register_interval >= 1s
# Description: интервал между повторными регистрациями в Config API
register_interval: ""

# Default: 30s
# Validate: config_update_interval >= 1s
# Description: интервал между двумя проверками наличия новой конфигурации в Config
API
config_update_interval: ""

# Default: 30s
# Validate: instance_send_interval >= 1s
# Description: интервал отправки текущей информации об инстансе в Config API
instance_send_interval: ""

# Default: 90s
# Validate: request_timeout >= 1s
# Description: таймаут HTTP запроса в Config API
request_timeout: ""

exporters:
- name: node_exporter
  start_type: args
  args: --log.format=json --web.listen-address=0.0.0.0:9100
  address: 127.0.0.1:9100

# Default: false
# Description: перетереть конфигурацию в Config API локальной конфигурацией из
этого файла
force_local: false

### Настройки логов бинарника
logger:
# Default: info
# Validate: trace, debug, info, warn, error, disabled
# Description: уровень логирования
level: ""

# Default: "", нет логирования в файл
# Validate: ^(\.\.\/)*(\.\/|\/)?[\w-]*(\.\w+)*$
# Description: путь к файлу, в который будут записаны логи бинарника
file: ""

# Default: "", обычное логирования zerolog console format с цветами
# Validate: "", json, nocolor, disabled

```

```
# Description: формат логирования в stderr
stderr: ""
```

3.2.5 Экспортёры доступные для установки через агент

Список бинарных файлов и экспортёров доступных для запуска и управления через агент astra monitoring

Экспортёры без дашбордов доступны для установки но корректность их работы не тестировалась. Такие экспортёры в данный момент не имеют готовых дашбордов с метриками, но они появятся в ближайших релизах.

Имя	Есть дашборд	Интеграция	Ссылка	Пример запуска через config.yml
ls_exporter	нет	1c	LazarenkoA/prometheus_1C_exporter ³⁰	
blackbox_exporter	да	http	prometheus/blackbox_exporter ³¹	
cadvisor_exporter	нет	docker, k8s	google/cadvisor ³²	<pre>exporters: - name: "cadvisor_exporter" start_type: "args" args: "-- port=9110" address: "127.0.0.1:9110" labels: group: "all"</pre>
ebpf_exporter	нет	ebpf	cloudflare/ebpf_exporter ³³	

30 https://github.com/LazarenkoA/prometheus_1C_exporter

31 https://github.com/prometheus/blackbox_exporter

32 <https://github.com/google/cadvisor>

33 https://github.com/cloudflare/ebpf_exporter

Имя	Есть дашборд	Интеграция	Ссылка	Пример запуска через config.yml
ipmi_exporter	да	ipmi	prometheus-community/ipmi_exporter ³⁴	
kafka_exporter	нет	kafka	danielqsj/kafka_exporter ³⁵	
libvirt_exporter	нет	libvirt	Tinkoff/libvirt-exporter ³⁶	<pre>exporters: - name: "libvirt_exporter" start_type: "args" args: "-- web.listen- address=0.0.0.0:917 7" address: "127.0.0.1:9177" labels: group: "all"</pre>
memcached_exporter	нет	memcached	prometheus/memcached_exporter ³⁷	<pre>exporters: - name: "memcached_exporter " start_type: "args" args: "-- memcached.address=l ocalhost:11211" address: "127.0.0.1:9150" labels: group: "all"</pre>

34 https://github.com/prometheus-community/ipmi_exporter

35 https://github.com/danielqsj/kafka_exporter

36 <https://github.com/Tinkoff/libvirt-exporter>

37 https://github.com/prometheus/memcached_exporter

Имя	Есть дашборд	Интеграция	Ссылка	Пример запуска через config.yml
node_exporter	да	linux host	prometheus/node_exporter ³⁸	<pre>exporters: - name: "node_exporter" start_type: "args" args: "-- log.format=json -- web.listen- address=0.0.0.0:9100" address: "127.0.0.1:9100" labels: group: "all"</pre>
nvidia_gpu_exporter	нет	nvidia	utkuozdemir/nvidia_gpu_exporter ³⁹	
postgres_exporter	да	postgres	prometheus-community/postgres_exporter ⁴⁰	<pre>exporters: - name: "postgres_exporter" start_type: "args" args: "-- web.listen- address=:9187" address: "127.0.0.1:9187" labels: group: "all"</pre>

38 https://github.com/prometheus/node_exporter

39 https://github.com/utkuozdemir/nvidia_gpu_exporter

40 https://github.com/prometheus-community/postgres_exporter

Имя	Есть дашборд	Интеграция	Ссылка	Пример запуска через config.yml
<code>process_exporter</code>	нет	<code>linux</code> <code>proces</code>	ncabatoff/process-exporter ⁴¹	
<code>rabbitmq_exporter</code>	нет	<code>rabbitmq</code>	kbudde/rabbitmq_exporter ⁴²	
<code>redis_exporter</code>	нет	<code>redis</code>	oliver006/redis_exporter ⁴³	

41 <https://github.com/ncabatoff/process-exporter>

42 https://github.com/kbudde/rabbitmq_exporter

43 https://github.com/oliver006/redis_exporter

Имя	Есть дашборд	Интеграция	Ссылка	Пример запуска через config.yml
snmp_exporter	да	snmp	prometheus/snmp_exporter ⁴⁴	<pre>exporters: - name: "snmp_exporter" start_type: "args" args: "-- config.file configs/snmp.yml" address: "127.0.0.1:9116" labels: group: "all"</pre> <p>Требуется конфиг файл. Таргеты для сбора метрик указываются отдельно.</p> <pre>metrics: custom_targets: - name: "cisco_example" target: "127.0.0.1:9116/ snmp? target=192.168.0.1" metrics_path: "/metrics" scrape_interval: "20s" labels: group: "snmp"</pre>

⁴⁴ https://github.com/prometheus/snmp_exporter

Имя	Есть дашборд	Интеграция	Ссылка	Пример запуска через config.yml
sql_exporter	да	sql	burningalchemist/sql_exporter ⁴⁵	<pre>exporters: - name: "sql_exporter" start_type: "args" args: "- config.file /etc/ sql-exporter.yml -web.listen-address 0.0.0.0:9399" address: "127.0.0.1:9399" labels: group: "all"</pre> <p>Требуется конфиг файл</p>
ssl_exporter	да	ssl	ribbybibby/ssl_exporter ⁴⁶	
vector	да	logs	vectordotdev/vector ⁴⁷	<pre>logs: enabled: true sources: - agent - collector - exporters - vmagent</pre>
vmagent	-	metrics	VictoriaMetrics ⁴⁸	

⁴⁵ https://github.com/burningalchemist/sql_exporter

⁴⁶ https://github.com/ribbybibby/ssl_exporter

⁴⁷ <https://github.com/vectordev/vector>

⁴⁸ <https://github.com/VictoriaMetrics/VictoriaMetrics>

Имя	Есть дашборд	Интеграция	Ссылка	Пример запуска через config.yml
vmauth	-	проху	VictoriaMetrics / VictoriaMetrics ⁴⁹	
windows_exporter	да	windows host	prometheus-community/windows_exporter ⁵⁰	<pre> exporters: - name: "windows_exporter" start_type: "args" args: "-- web.listen- address=0.0.0.0:918 2" address: "127.0.0.1:9182" labels: group: "all" hostname: "windows-hostname" </pre>
freeipa_exporter	да	freeipa		
one_exporter	да	one		
rubackup_exporter	да	rubackup		

⁴⁹ <https://github.com/VictoriaMetrics/VictoriaMetrics>

⁵⁰ https://github.com/prometheus-community/windows_exporter

Имя	Есть дашборд	Интеграция	Ссылка	Пример запуска через config.yml
rupost_exporter	да	rupost		<pre>exporters: - name: "rupost_exporter" start_type: "args" args: "-- config=/etc/rupost- exporter.yml" address: "127.0.0.1:9777" labels: group: "all"</pre> <p>Требуется конфиг файл</p>
systemd_exporter	да	systemd		<pre>exporters: - name: "systemd_exporter" start_type: "args" args: "-- config=/etc/ systemd- exporter.conf" address: "127.0.0.1:9750" labels: group: "all"</pre> <p>Требуется конфиг файл</p>
termidesk_exporter	да	termidesk		

Бинарные файлы экспортёров можно скачать в ручную с публичного репозитория dl.astralinux.

Ссылка имеет фиксированную структуру:

[https://dl.astralinux.ru/am/generic/exporters/\\${ИМЯ}/\\${ВЕРСИЯ}/\\${ОС}/\\${АРХИТЕКТУРА}/\\${ИМЯ}.tgz](https://dl.astralinux.ru/am/generic/exporters/${ИМЯ}/${ВЕРСИЯ}/${ОС}/${АРХИТЕКТУРА}/${ИМЯ}.tgz)

Значение экспортёры публикуются с 3 доступными версиями. 1) Численная версия 2) latest 3) release-0.5.0 (версия релиза инкрементируется). Таким образом рекомендуется устанавливать release-0.5.0 версию экспортёров, так как она тестируется для конкретного релиза.

Ниже представлен пример для для node_exporter, доступные NAME можно посмотреть в таблице выше:

```
> export NAME=node_exporter
> curl -sLo ${NAME}.tgz https://dl.astralinux.ru/am/generic/exporters/${NAME}/latest/linux/amd64/${NAME}.tgz
> tar zxvf ${NAME}.tgz
```

Далее необходимо сконфигурировать требуемые экспортеры (см. соответствующие разделы).

3.3 Установка внешних экспортёров для продуктов астры

Рекомендуется использовать агенты для установки экспортёров!

Установка внешних экспортёров не позволяет управлять их конфигурацией через astra monitoring!

3.3.1 Установка клиентской части для ALD Pro / FreeIPA



При создании объектов мониторинга из административной консоли ознакомьтесь с [разделом 6.4 \(see page 134\)](#) и [разделом 6.5 \(see page 135\)](#).

Клиентская часть мониторинга ALD Pro / FreeIPA состоит из нескольких компонентов.

Установка freeipa-exporter

Для установки необходимо скачать и распаковать архив с настройками и экспортером:

```
# Скачиваем опубликованный архив
> curl -sLo freeipa-exporter.tgz https://dl.astralinux.ru/am/generic/freeipa-exporter/latest/freeipa-exporter.tgz
> tar zxvf freeipa-exporter.tgz
> cd ./freeipa-exporter/

# Копируем исполняемый файл в каталог
> sudo cp ./freeipa-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/freeipa-exporter
> sudo chown root:root /usr/local/bin/freeipa-exporter

# Копируем конфигурацию сервиса в системный каталог
> sudo cp ./freeipa-exporter.service /usr/lib/systemd/system/freeipa-exporter.service

# Редактируем параметры сервиса (описание параметров далее по тексту)
> vi ./freeipa-exporter.conf
> sudo cp ./freeipa-exporter.conf /etc/freeipa-exporter.conf
```

```
# Обновляем конфигурацию и запускаем сервис
> sudo systemctl enable freeipa-exporter.service
> sudo systemctl start freeipa-exporter.service
> sudo systemctl status freeipa-exporter.service
```

Редактирование файла параметров сервиса:

/etc/freeipa-exporter.conf

```
username: ""
password: ""
users_dn: ""
```

В FreeIPA необходимо создать пользователя (см. [2.4 Требования к учётной записи](#)) и подставить его имя вместо <user name> (например, testuser) . Пароль зашифровать base64 и заменить в поле <user passwd>. В User_dn поместить правильный dn для пользователей (например, cn=users,cn=accounts,dc=example,dc=loc).

/etc/freeipa-exporter.conf

```
DNScheck:
- DNSserver: 77.88.8.8                (адрес DNS сервера для отправки запросов.
  Может быть localhost для проверок локального DNS)
  hostname: artifactory.astralinux.ru  (FQDN имя сервера, для которого
  необходимо получить ip-адрес)
  hostip: 51.250.55.40                (ip-адрес, который мы ожидаем получить в
  результате DNS-запроса)
- DNSServer: localhost
  hostname: artifactory.astralinux.ru
  hostip: 51.250.55.40
- DNSServer: localhost
  hostname: dc01.example.loc
  hostip: localhost
```

Для проверок DNS необходимо указать требуемые параметры в разделе DNScheck. Проверок может быть несколько для каждого из типов external/internal.

Конфигурация сборщика логов

Приложение Vector должно быть установлено на всех необходимых серверах (см. 3.5.2):

```
# Скачиваем опубликованный архив с конфигурацией Vector для ALD Pro
> curl -sLo freeipa-vector.tgz https://dl.astralinux.ru/am/generic/freeipa-exporter/
latest/freeipa-vector.tgz
> tar zxvf freeipa-vector.tgz
> cd ./freeipa-vector/

# Копируем конфигурационные файлы в каталог Vector
```

```
> sudo cp *.yaml /etc/vector/conf.d/

# Обновляем конфигурацию и запускаем сервис
> sudo systemctl restart vector.service
```

3.3.2 Установка клиентской части для ПК СВ Брест

Клиентская часть мониторинга Брест состоит из нескольких компонентов.

Установка one-exporter

Установка one-exporter выполняется на **одном** сервере, у которого есть сетевой доступ к API Брест и который будет доступен для Платформы мониторинга напрямую, либо через публикацию порта экспортера. Для установки также требуется создать сервисную учетную запись в Брест с использованием драйвера авторизации "core", с указанием основной группы - brestadmins, дополнительной - brestusers.

Для установки необходимо скачать и распаковать архив с настройками и экспортером:

```
# Скачиваем опубликованный архив
> curl -sLo one-exporter.tgz https://dl.astralinux.ru/am/generic/one-exporter/latest/one-exporter.tgz
> tar zxvf one-exporter.tgz
> cd ./one-exporter/

# Копируем исполняемый файл в каталог
> sudo cp ./one-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/one-exporter
> sudo chown root:root /usr/local/bin/one-exporter

# Копируем конфигурацию сервиса в системный каталог
> sudo cp ./one-exporter.service /usr/lib/systemd/system/one-exporter.service

# Редактируем параметры сервиса (описание параметров далее по тексту)
> vi ./one-exporter.conf
> sudo cp ./one-exporter.conf /etc/one-exporter.conf
```

Изменить конфигурационный файл с названием /etc/one-exporter.conf со следующим содержимым, изменив в нем параметры, отмеченные <.> на свои:

```
---

base:
  # address to run the exporter on, mandatory
  address: 0.0.0.0:9621

  # exporter uri to publish on, mandatory
  metrics_path: /metrics

  # frequency to retrieve metrics in seconds, mandatory
```

```

interval: 60s

# file path to write logs (default: empty and disabled log to file)
log_file: /var/log/astra-monitoring/astra-one-exporter.log

# minimum log level (default: debug)
log_level: info

# disable printing log to stderr (default: false)
# disable_console_log: true

```

Установка одинакова для мониторинга любого из продуктов "Группы Астра".

Допускается использовать более новую версию Vector, но рекомендуется версия новее чем **0.33.1** (изменение формата конфигурационных файлов).

Установить и запустить Vector для сбора требуемых логов можно разными способами:

```

# enable default go metrics (default: false)
go_metrics: true

# enable default process metrics (default: false)
proc_metrics: true

# credentials to access OpenNebula
#user: Brest/OpenNebula service account name
#password: base64 encrypted password
user: <user name>
password: <password>

# OpenNebula frontend endpoint
# an empty endpoint will default to http://localhost:2633/RPC2
# endpoint:
endpoint: http://<Brest/Opennebula API url>:2633/RPC2

# Include VMs state and performance metrics (true/false). Disabled by default.
vminfo: false

# OpenNebula Web console
webconsole: https://<Brest/Opennebula webconsole url>/login

# enabling monitoring web console for users
# leave it empty if not (ex. webuser: )
# webpasswd: base64 encrypted password
webuser: <web user name>
webpasswd: <web password>

```

где `user` - имя сервисной учетной записи API Брест, созданной ранее (см. [2.4 Требования к учётной записи](#)),

`password` - пароль учетной записи, закодированный в base64,

`endpoint` - адрес API Брест,

`webconsole` - адрес интерфейса управления Брест,

`webuser` - пользователь для проверки доступа к консоли управления (драйвер авторизации `public`),

`webpasswd` - пароль пользователя, закодированный в `base64`.

Затем разрешить запуск сервиса после перезагрузки и запустить сервис:

```
sudo systemctl enable one-exporter.service
sudo systemctl start one-exporter.service
sudo systemctl status one-exporter.service
```

При размещении сервера с установленным `one-exporter` за сетевым экраном, NAT или другим подобным компонентом, необходимо опубликовать порт 9621/TCP и убедиться в его доступности для Платформы мониторинга.

Установка `libvirt-exporter`

Установка `libvirt-exporter` выполняется на всех серверах-гипервизорах Брест с работающим компонентом `libvirt`.

Для установки необходимо скачать и распаковать архив с настройками и экспортером:

```
# Скачиваем опубликованный архив
> curl -sLo libvirt-exporter.tgz https://dl.astralinux.ru/am/generic/libvirt/libvirt-exporter.tgz
> tar zxvf libvirt-exporter.tgz
> cd ./libvirt-exporter/

# Копируем исполняемый файл в каталог
> sudo cp ./libvirt_exporter_improved /usr/local/bin/
> sudo chmod 755 /usr/local/bin/libvirt_exporter_improved
> sudo chown root:root /usr/local/bin/libvirt_exporter_improved

# Копируем конфигурацию сервиса в системный каталог
> sudo cp ./libvirt-exporter.service /usr/lib/systemd/system/libvirt-exporter.service

# Разрешаем и включаем сервис
> sudo systemctl enable libvirt-exporter.service
> sudo systemctl start libvirt-exporter.service
> sudo systemctl status libvirt-exporter.service
```

При размещении сервера с установленным `libvirt-exporter` за сетевым экраном, NAT или другим подобным компонентом, необходимо опубликовать порт 9177/TCP и убедиться в его доступности для Платформы мониторинга. Если серверов несколько, то порты могут быть опубликованы, например, как 19177, 29177 и т.п.

Установка `postgres-exporter` и `sql-exporter`

Для сбора метрик с базы данных Брест необходимо установить `postgres-exporter` (см. 3.5.6) и `sql-exporter` (см. 3.5.7)

Для sql-exporter в файле `postgre_standard.collector.yml` необходимо раскомментировать названия нужных метрик (в названии используется слово `brestr`).

Конфигурация сборщика логов

Приложение Vector должно быть установлено на всех необходимых серверах (см. 3.5.2):

```
# Скачиваем опубликованный архив с конфигурацией Vector для Бреста
> curl -sLo one-vector.tgz https://dl.astralinux.ru/am/generic/one-exporter/latest/
one-vector.tgz
> tar zxvf one-vector.tgz
> cd ./one-vector/

# Копируем конфигурационные файлы в каталог Vector
> sudo cp *.yaml /etc/vector/conf.d/

# Обновляем конфигурацию и запускаем сервис
> sudo systemctl restart vector.service
```

3.3.3 Установка клиентской части для RuPost

Клиентская часть мониторинга RuPost состоит из нескольких компонентов.

Установка `rupost-exporter`

Установка `rupost-exporter` выполняется на сервере, где расположен компонент Postfix.

Для установки необходимо скачать и распаковать архив с настройками и экспортером:

```
# Скачиваем опубликованный архив
> curl -sLo rupost-exporter.tgz https://dl.astralinux.ru/am/generic/rupost-exporter/
latest/rupost-exporter.tgz
> tar zxvf rupost-exporter.tgz
> cd ./rupost-exporter/
# Копируем исполняемый файл в каталог
> sudo cp ./rupost-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/rupost-exporter
> sudo chown root:root /usr/local/bin/rupost-exporter
# Копируем конфигурацию сервиса в системный каталог
> sudo cp ./rupost-exporter.service /usr/lib/systemd/system/rupost-exporter.service
# Редактируем параметры сервиса (описание параметров далее по тексту)
> vi ./rupost-exporter.conf
> sudo cp ./rupost-exporter.conf /etc/rupost-exporter.yaml
# Разрешаем и включаем сервис
> sudo systemctl enable rupost-exporter.service
> sudo systemctl start rupost-exporter.service
> sudo systemctl status rupost-exporter.service
```

В конфигурационном файле необходимо заменить параметры, отмеченные `<.>` на свои:

```

base:
  # host and port to run the exporter on (default: 0.0.0.0:9777)
  address: 0.0.0.0:9777

  # exporter uri to publish on (default: /metrics)
  metrics_path: /metrics

  # frequency to retrieve metrics in seconds (default: 60s)
  interval: 60s

  # file path to write logs (default: empty and disabled log to file)
  log_file: /var/log/astra-monitoring/astra-rupost-exporter.log

  # minimum log level (default: debug)
  log_level: debug

  # disable printing log to stderr (default: false)
  # disable_console_log: true

  # metric name will be "namespace_subsystem_metricname" (default: empty)
  # namespace: mon
  subsystem: rupost

  # enable default go metrics (default: false)
  go_metrics: true

  # enable default process metrics (default: false)
  proc_metrics: true

  # name of queues what should be checked
  queue_names:
  - incoming
  - active
  - deferred
  - hold

  # path to directory with queues of Rupost
  # example: "/var/spool/postfix/d8d05a26-2521-416e-8acf-3a110b50e8d2"
  base_dir: <..>

```

Base_dir для каждой ноды RuPost своя. Определить ее можно из консоли управления RuPost.

Мониторинг

Экземпляры RuPost

i Действия с экземплярами приложения

Обновить информацию об экземплярах приложения

mail01.aquila.astralinux.ru Узел доступен Экземпляр активен Обновлено в 15.02.2024

Ввод в эксплуатацию Вывод из эксплуатации Перезапуск Статус

UID экземпляра RuPost: **d8d05a26-2521-416e-8acf-3a110b50e8d2**

Компонент	Статус	Ошибка	Время изменения статуса
haproxy	Запущен		15.02.2024 09:34 +03:00
nginx	Запущен		15.02.2024 09:34 +03:00

Настройка haproxy-exporter

Настройка haproxy-exporter выполняется на сервере, где расположен компонент Haproxy. Этот компонент может самостоятельно выдавать метрики в формате prometheus. Для этого необходимо выполнить настройку через конфигурационный файл `/etc/haproxy/haproxy.cfg`. В связи с особенностями работы приложения RuPost необходимо делать настройку через создание шаблона конфигурации RuPost.

В конец файла добавляем:

```
frontend stats
  bind *:8000
  mode http
  http-request use-service prometheus-exporter if { path /metrics }
  stats enable
  stats uri /stats
  stats refresh 15s
  stats show-node
  # stats auth admin:password
```

Строка `# stats auth admin:password` используется, если компонент Haproxy настроен с использованием логина и пароля.

После настройки необходимо выполнить restart службы Haproxy, и метрики будут доступны на порту 8000.

Настройка dovecot-exporter

Настройка dovecot-exporter выполняется на сервере, где расположены компонент Dovecot. Этот компонент может самостоятельно выдавать метрики в формате prometheus. Для этого необходимо выполнить настройку через конфигурационный файл `/etc/dovecot/dovecot.conf`. В связи с особенностями работы приложения RuPost необходимо делать настройку через создание шаблона конфигурации RuPost.

Находим раздел `service stats`:

```
service stats {
    client_limit = 20000
    vsz_limit = 128M
}
```

и добавляем в него строки

```
service stats {
    client_limit = 20000
    vsz_limit = 128M
    inet_listener http {
        port = 9900
    }
}
```

В конец конфигурационного файла добавляем строки:

```
metric auth_success {
    filter = event=auth_request_finished AND success=yes
}
metric auth_failures {
    filter = event=auth_request_finished AND NOT success=yes
}
metric imap_command {
    filter = event=imap_command_finished
    group_by = cmd_name tagged_reply_state
}
metric smtp_command {
    filter = event=smtp_server_command_finished
    group_by = cmd_name status_code
}
metric mail_delivery {
    filter = event=mail_delivery_finished
}
metric sql_query {
    filter = event=sql_connection_finished
}
metric index_recreate {
    filter = event=mail_index_recreated
}
```

После настройки необходимо выполнить restart службы Dovecot, и метрики будут доступны на порту 9900.

Установка postgres-exporter и sql-exporter

Для сбора метрик с базы данных RuPost необходимо установить postgres-exporter (см. 3.5.6) и sql-exporter (см. 3.5.7)

Для sql-exporter в файле `postgre_standard.collector.yml` необходимо раскомментировать названия нужных метрик (в названии используется слово `rupost`). Также необходимо полностью раскомментировать метрики, которые созданы только для RuPost.

```
# - metric_name: rupost_top10_users_by_quota_bytes
#   type: gauge
#   help: 'TOP 10 users who spend more bytes on storage'
#   key_labels:
#     - username
#   values: [quota_bytes]
#   query: |
#     select distinct username, quota_bytes from rp_user_quota_state group by 1, 2 order by quota_bytes desc fetch first 10 rows only;
- metric_name: rupost_top10_users_by_messages
#   type: gauge
#   help: 'TOP 10 users who have more messages'
#   key_labels:
#     - username
#   values: [messages]
#   query: |
#     select distinct username, messages from rp_user_quota_state group by 1, 2 order by messages desc fetch first 10 rows only;
```

Конфигурация сборщика логов

Приложение Vector должно быть установлено на всех необходимых серверах (см. 3.5.2):

```
# Скачиваем опубликованный архив с конфигурацией Vector для RuPost
> curl -sLo rupost-vector.tgz https://dl.astralinux.ru/am/generic/rupost-exporter/latest/rupost-vector.tgz
> tar zxvf rupost-vector.tgz
> cd ./rupost-vector/

# Копируем конфигурационные файлы в каталог Vector
> sudo cp *.yaml /etc/vector/conf.d/

# Обновляем конфигурацию и запускаем сервис
> sudo systemctl restart vector.service
```

3.3.4 Установка клиентской части для RuBackup

Клиентская часть мониторинга RuBackup состоит из нескольких компонентов.

Установка `rubackup-exporter`

Установка `rubackup-exporter` выполняется на сервере, где расположен компонент `Rubackup-server`.

Для установки необходимо скачать и распаковать архив с настройками и экспортером:

```
# Скачиваем опубликованный архив
> curl -sLo rubackup-exporter.tgz https://dl.astralinux.ru/am/generic/rubackup-exporter/latest/rubackup-exporter.tgz
> tar zxvf rubackup-exporter.tgz
> cd ./rubackup-exporter/
# Копируем исполняемый файл в каталог
> sudo cp ./rubackup-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/rubackup-exporter
> sudo chown root:root /usr/local/bin/rubackup-exporter
# Копируем конфигурацию сервиса в системный каталог
```

```

> sudo cp ./rubackup-exporter.service /usr/lib/systemd/system/rubackup-exporter.servi
ce
# Редактируем параметры сервиса (описание параметров далее по тексту)
> vi ./rubackup-exporter.conf
> sudo cp ./rubackup-exporter.conf /etc/rubackup-exporter.conf
# Разрешаем и включаем сервис
> sudo systemctl enable rubackup-exporter.service
> sudo systemctl start rubackup-exporter.service
> sudo systemctl status rubackup-exporter.service

```

В конфигурационном файле необходимо заменить параметры, отмеченные <.> на свои:

```

# Базовые обязательные настройки
base:
# Адрес, на котором запускается экспортер
address: 127.0.0.1:9444

# Эндпоинт, на котором происходит сбор метрик
metrics_path: /metrics

# Частота опроса метрик
interval: 60s

# Настройки логгера экспортера
logger:
# Уровень логирования
level: info
# Путь до файла лога
file: /var/log/astra-monitoring/rubackup-exporter.log

# Выключение логирования в консоль (default: false)
disable_console_log: true

# Включение стандартных go метрик (default: false)
go_metrics: false

# Включение стандартных proc метрик (default: false)
proc_metrics: false

# Path to RuBackup API (default: "")
api_path: "https://localhost:5656/api/v1/"

# User for API of RuBackup (default: "")
api_user: "user"

# Password for API user of RuBackup (default: "")
api_pass: "password"

# Authentication type (default: "database")
api_type: "database"

```

Конфигурация сборщика логов

Приложение Vector должно быть установлено на всех необходимых серверах (см. 3.5.2):

```
# Скачиваем опубликованный архив с конфигурацией Vector для RuBackup
> curl -sLo rubackup-vector.tgz https://dl.astralinux.ru/am/generic/rubackup-
exporter/latest/rubackup-vector.tgz
> tar zxvf rubackup-vector.tgz
> cd ./rubackup-vector/

# Копируем конфигурационные файлы в каталог Vector
> sudo cp *.yaml /etc/vector/conf.d/

# Обновляем конфигурацию и запускаем сервис
> sudo systemctl restart vector.service
```

3.3.5 Установка клиентской части для Termidesk

Установка termidesk-exporter

Установка termidesk-exporter выполняется на сервере, где расположен компонент VDI.

Для установки необходимо скачать и распаковать архив с настройками и экспортером:

```
# Скачиваем опубликованный архив
> curl -sLo termidesk-exporter.tgz https://dl.astralinux.ru/am/generic/termidesk-
exporter/latest/termidesk-exporter.tgz
> tar zxvf termidesk-exporter.tgz
> cd ./termidesk-exporter/
# Копируем исполняемый файл в каталог
> sudo cp ./termidesk-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/termidesk-exporter
> sudo chown root:root /usr/local/bin/termidesk-exporter
# Копируем конфигурацию сервиса в системный каталог
> sudo cp ./termidesk-exporter.service /usr/lib/systemd/system/termidesk-exporter.ser
vice
# Редактируем параметры сервиса (описание параметров далее по тексту)
> vi ./termidesk-exporter.conf
> sudo cp ./termidesk-exporter.conf /etc/termidesk-exporter.yaml
# Разрешаем и включаем сервис
> sudo systemctl enable termidesk-exporter.service
> sudo systemctl start termidesk-exporter.service
> sudo systemctl status termidesk-exporter.service
```

В конфигурационном файле необходимо заменить параметры, отмеченные <..> на свои:

```
base:
  # host and port to run the exporter on (default: 0.0.0.0:9777)
```

```
address: 0.0.0.0:9555

# exporter uri to publish on (default: /metrics)
metrics_path: /metrics

# frequency to retrieve metrics in seconds (default: 60s)
interval: 60s

# file path to write logs (default: empty and disabled log to file)
log_file: /var/log/astra-monitoring/astra-termidesk-exporter.log

# minimum log level (default: debug)
log_level: debug

# disable printing log to stderr (default: false)
# disable_console_log: true

# metric name will be "namespace_subsystem_metricname" (default: empty)
# namespace:
# subsystem:

# enable default go metrics (default: false)
go_metrics: true

# enable default process metrics (default: false)
proc_metrics: true

# Path to Termidesk web console (default: "") (example - "http://127.0.0.1")
path: ""

# User for API of RuBackup (default: "")
api_user: ""

# Password for API user of RuBackup (default: "")
api_pass: ""

# Auth name of authenticator domain for user credentials above (/admin/auth-domains
your termidesk web console) (default: "")
auth_name: ""

# URL for discover of modules API versions (default: "/api/discover")
discover_api: "/api/discover"

# Token for Health API requests (/etc/opt/termidesk-vdi/termidesk.conf,
HEALTH_CHECK_ACCESS_KEY variable) (default: "")
health_token: ""

# Taskman port for Health API requests (default: "8100")
taskman_port: "8100"

# WSProxy port for Health API requests (default: "8101")
wsproxy_port: "8101"
```

Конфигурация сборщика логов

Приложение Vector должно быть установлено на всех необходимых серверах (см. 3.5.2):

```
# Скачиваем опубликованный архив с конфигурацией Vector для RuBackup
> curl -sLo termidesk-vector.tgz https://dl.astralinux.ru/am/generic/termidesk-
exporter/latest/termidesk-vector.tgz
> tar zxvf termidesk-vector.tgz
> cd ./termidesk-vector/

# Копируем конфигурационные файлы в каталог Vector
> sudo cp *.yaml /etc/vector/conf.d/

# Обновляем конфигурацию и запускаем сервис
> sudo systemctl restart vector.service
```

3.3.6 Установка клиентской части для Billmanager

Установка billm-exporter

Установка billm-exporter выполняется на сервере, с которого есть доступ до API Billmanager (по умолчанию сервер с установленным Billmanager).

Для установки необходимо скачать и распаковать архив с экспортером:

```
# Скачиваем опубликованный архив
> curl -sLo billm-exporter.tgz https://dl.astralinux.ru/am/generic/billm-exporter/
latest/billm-exporter.tgz
> tar zxvf billm-exporter.tgz
> cd ./billm-exporter/
# Копируем исполняемый файл в каталог
> sudo cp ./billm-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/billm-exporter
> sudo chown root:root /usr/local/bin/billm-exporter
# Создаем и копируем конфигурацию сервиса в системный каталог
> vim billm-exporter.service
> sudo cp ./billm-exporter.service /usr/lib/systemd/system/billm-exporter.service
# Создаем файл конфигурации
> vim ./billm-exporter.conf
> sudo cp ./billm-exporter.conf /etc/billm-exporter.yaml
# Разрешаем и включаем сервис
> sudo systemctl enable billm-exporter.service
> sudo systemctl start billm-exporter.service
> sudo systemctl status billm-exporter.service
```

Пример сервисного файла:

```
[Unit]
```

```

Description=Astra Monitoring BillManager exporter
Wants=basic.target
After=basic.target network.target

[Service]
ExecStart=/usr/local/bin/billm-exporter -c /etc/billm-exporter.conf -l "0.0.0.0:9696"

ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=always

[Install]
WantedBy=multi-user.target

```

Пример конфигурационного файла:

```

base:
  # host and port to run the exporter on (default: 0.0.0.0:9696)
  address: 0.0.0.0:9696

  ### Optional fields
  # exporter uri to publish on (default: /metrics)
  metrics_path: /metrics

  # health uri to check exporter health
  health_path: /health

  # frequency to retrieve metrics in seconds (default: 60s)
  interval: 60s

  # enable default go and proc metrics (default: false)
  go_metrics: false
  proc_metrics: false

  #logger options
  logger:
    level: debug
    file: /var/log/astra-monitoring/billm-exporter.log
    # stderr: disabled

  #Path to Billmanager API (Example: "https://192.168.0.1:1500/billmgr?")
  api_path: ""

  #User login for monitoring (default: "")
  api_user: ""

  #Password for API user in BASE64 encoding (default: "")
  api_pass: ""

  #List of providers to check them status
  processing_list:
    #ID of provider

```

```

- number: 1
#Name for human readable in grafana dashboards
  name: Brest
- number: 2
  name: Test

```

3.4 Установка внешних компонентов удаленного мониторинга

Рекомендуется использовать агенты для установки экспортеров!

Установка внешних экспортеров не позволяет управлять их конфигурацией через astra monitoring!

3.4.1 Установка snmp-exporter

Ниже рассматривается вариант установки экспортера как сервис. Для запуска через агент мониторинга см. [раздел 6.12 \(see page 154\)](#).

[SNMP-exporter](#)⁵¹ устанавливается на специальные выделенные хосты, для удаленного опроса целевых хостов (железные сервера, сетевые устройства и пр.).

```

# Скачиваем опубликованный архив
> curl -sLo snmp-exporter.tgz https://dl.astralinux.ru/am/generic/snmp-exporter/0.25.0/snmp-exporter.tgz
> tar zxvf snmp-exporter.tgz

# Копируем исполняемый файл в каталог
> sudo cp ./snmp-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/snmp-exporter
> sudo chown root:root /usr/local/bin/snmp-exporter

# Копируем конфигурацию сервиса в системный каталог
> sudo cp ./snmp-exporter.service /usr/lib/systemd/system/snmp-exporter.service

```

Конфигурационный файл для snmp-exporter генерируется специальной утилитой generator. Она так же содержится в архиве экспортера.

В архив с snmp-exporter добавлен каталог стандартных mib файлов, для генерации конфигов.

Необходимые для мониторинга устройств mib файлы размещаем в mibs директорию.

```

# Подготавливаем файл конфигурации для generator (пример далее)
> vim generator.yml

# Генерируем конфигурационный файл для snmp-exporter
> ./generator --fail-on-parse-errors generate -m mibs/ -g generator.yml

```

⁵¹ https://github.com/prometheus/snmp_exporter

В результате генерируется snmp.yml. Размещаем его по пути, который указан в unit file для сервиса snmp-exporter.

```
# Копируем конфигурационный файл
> sudo cp ./snmp.yml /etc/snmp.yml

# Разрешаем и включаем сервис
> sudo systemctl enable snmp-exporter.service
> sudo systemctl start snmp-exporter.service
> sudo systemctl status snmp-exporter.service
```

Пример generator.yml:

```
auths:
  public_v1:
    community: public
    security_level: noAuthNoPriv
    auth_protocol: MD5
    priv_protocol: DES
    version: 1
  public_v2:
    community: public
    security_level: noAuthNoPriv
    auth_protocol: MD5
    priv_protocol: DES
    version: 2
modules:
  arcup:
    walk:
      - 1.3.6.1.4.1
  linux:
    walk:
      - 1.3.6.1.4.1.2021
```

3.4.2 Установка ipmi-exporter

Ниже рассматривается вариант установки экспортера как сервиса. Для запуска через агент мониторинга см. [раздел 6.13 \(see page 155\)](#).

IPMI-exporter⁵² устанавливается на специальные выделенные хосты, для удаленного опроса целевых хостов (железные сервера, другие устройства), которые поддерживают протокол IPMI.

Для работы ipmi-exporter необходим установленный набор утилит **FreeIPMI**⁵³.

```
# Устанавливаем дополнительные пакеты
> sudo apt-get -y install build-essential libgrypt20-dev
```

⁵² https://github.com/prometheus-community/ipmi_exporter

⁵³ <https://www.gnu.org/software/freeipmi>

```
# Качаем архив с набором утилит
> wget https://ftp.gnu.org/gnu/freeipmi/freeipmi-1.6.9.tar.gz
# Распаковываем
> tar -xvf freeipmi-1.6.9.tar.gz
> cd freeipmi-1.6.9/
# собираем и конфигурируем пакет
> ./configure
> make
> sudo make install
> sudo apt -y install freeipmi
```

Устанавливаем экспортер:

```
# Скачиваем опубликованный архив
> curl -sLo ipmi-exporter.tgz https://dl.astralinux.ru/am/generic/ipmi-exporter/1.8.0/ipmi-exporter.tgz
> tar zxvf ipmi-exporter.tgz

# Копируем исполняемый файл в каталог
> sudo cp ./ipmi-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/ipmi-exporter
> sudo chown root:root /usr/local/bin/ipmi-exporter

# Копируем конфигурацию сервиса в системный каталог
> sudo cp ./ipmi-exporter.service /usr/lib/systemd/system/ipmi-exporter.service

# Копируем конфигурационный файл
> sudo cp ./ipmi-exporter.yml /etc/ipmi-exporter.yml

# Разрешаем и включаем сервис
> sudo systemctl enable ipmi-exporter.service
> sudo systemctl start ipmi-exporter.service
> sudo systemctl status ipmi-exporter.service
```

Пример конфигурационного файла:

```
modules:
  default:
    user: "demo"
    pass: "demo"
    driver: "LAN_2_0"
    privilege: "user"
    collectors:
      - bmc
      - ipmi
      - chassis
    collector_cmd:
      ipmi: sudo
    custom_args:
      ipmi:
        - "ipmimonitoring"
```

3.4.3 Установка ssl-exporter

Ниже рассматривается вариант установки экспортера как сервис. Для запуска через агент мониторинга см. [раздел 6.15 \(see page 185\)](#).

В зависимости от типа сертификатов [ssl-exporter](#)⁵⁴ может устанавливаться как на хост с сертификатами, так и на удаленном хосте (например, коллектор).

Экспортер позволяет собирать данные о сертификатах из файлов, web ssl сертификаты сайтов, секреты kubernetes.

Устанавливаем экспортер:

⁵⁴ https://github.com/ribbybibby/ssl_exporter/

```

# Скачиваем опубликованный архив
> curl -sLo ssl-exporter.tgz https://dl.astralinux.ru/am/generic/ssl-exporter/2.4.3/ssl-exporter.tgz
> tar zxvf ssl-exporter.tgz

# Копируем исполняемый файл в каталог
> sudo cp ./ssl-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/ssl-exporter
> sudo chown root:root /usr/local/bin/ssl-exporter

# Копируем конфигурацию сервиса в системный каталог
> sudo cp ./ssl-exporter.service /usr/lib/systemd/system/ssl-exporter.service

# Копируем конфигурационный файл
> sudo cp ./ssl-exporter.yml /etc/ssl-exporter.yml

# Разрешаем и включаем сервис
> sudo systemctl enable ssl-exporter.service
> sudo systemctl start ssl-exporter.service
> sudo systemctl status ssl-exporter.service

```

Пример конфигурационного файла:

```

default_module: https
modules:
  https:
    prober: https
  https_insecure:
    prober: https
    tls_config:
      insecure_skip_verify: true
  tcp:
    prober: tcp
  tcp_client_auth:
    prober: tcp
    tls_config:
      ca_file: /etc/ssl/GlobalSignRSA2018.crt
  tcp_client_auth_error:
    prober: tcp
    tls_config:
      ca_file: /etc/ssl/ca.crt
  file:
    prober: file
  file_ca_certificates:
    prober: file
    target: /etc/ssl/agent/agent.crt

```

3.4.4 Установка SNMP Агента

Агент позволяет получать и обрабатывать SNMP трапы.

Ниже рассматривается вариант установки агента как сервис.

```
# Скачиваем опубликованный архив
> curl -sLo snmp-agent.tgz https://dl.astralinux.ru/am/generic/snmp-agent/0.0.1/snmp-agent.tgz
> tar zxvf snmp-agent.tgz
> cd snmp-agent
# Для упрощенной установки агента запускаем скрипт с правами администратора
> sudo ./install_service.sh
> sudo systemctl status snmp-agent.service
```

Пример запуска агента непосредственно из консоли:

```
# Запускаем из бинарного файла
> snmp-agent trap --config="./config.yaml"
```

Пример конфигурационного файла:

```
snmp: # SNMP трап коллектор
  address: "0.0.0.0" # Адрес на котором слушает коллектор (0.0.0.0 - любой адрес)
  port: 162 # Порт на который получаем трапы
  protocol: "udp" # Протокол (udp или tcp)
  version: "2c" # Версия SNMP протокола (1, 2c, 3)
  community: "public"
  timeout: 3

interpreter: # Нормализатор
  timeout: 30 # Таймаут обработки файла правил
  type: file # Тип правил (file - путь к файлу, plain - текст, код файла правил в конфигурационном файле)
  rules: "./rules.js" # Правила, в зависимости от типа выбранного выше может быть путем к файлу правил или текстом

database: # Подключение к базе данных вызова хранимой процедуре. Если не включать этот сегмент в конфигурационный файл то результат будет записываться только в файл логов
  host: "127.0.0.1"
  port: 5432
  username: "postgres"
  password: "pass"
  database: "postgres" # Имя базы данных в которой вызываем хранимую процедуру

log: # Логирование
  output: "file" # Направление вывода лога (file, stdout, discard)
```

```
file: "./logs/snmptrap.log" # Путь к файлу логов
```

Также агент может быть запущен в виде docker-контейнера (необходим установленный и работающий docker). Обратите внимание, что в созданном из образа контейнере необходимо отредактировать файл конфигурации `/etc/snmp-agent/snmp-agent.yaml`. Другой вариант - создать на хосте директорию с файлом конфигурации и правилами обработки трапов и примонтировать ее в контейнер как `/etc/snmp-agent`.

```
docker pull aicl-registry.artifactory.astralinux.ru/snmp-agent:latest
docker run -d aicl-registry.artifactory.astralinux.ru/snmp-agent

# Если на хосте создана директория /opt/snmp-agent с файлом конфигурации и правилами
docker run -d -v /opt/snmp-agent:/etc/snmp-agent aicl-
registry.artifactory.astralinux.ru/snmp-agent
```

3.4.5 Установка Vector

Установка одинакова для мониторинга любого из продуктов "Группы Астра".

Примечание к релизу

Допускается использовать более новую версию Vector, но рекомендуется версия новее чем 0.33.1 (изменение формата конфигурационных файлов).

Установить и запустить Vector для сбора требуемых логов можно разными способами:

- Установка через Агент. В конфигурационном файле агента необходимо включить сбор логов:

```
logs:
  enabled: True
  all_sources: True
```

После выполнения данной команды агент скачает vector и запустит автоматически сконфигурировав точку отправки логов в платформу am. В качестве целей сбора логов будут выступать сам агент и запущенные через него экспортёры.

- Ручная установка.

```
# Установить DEB-пакет
> curl -sLo /tmp/vector_0.33.1-1_amd64.deb 'https://packages.timber.io/vector/0.33.1/vector_0.33.1-1_amd64.deb'
> sudo dpkg -i /tmp/vector_0.33.1-1_amd64.deb
> rm -f /tmp/vector_0.33.1-1_amd64.deb

# Внести изменения в файл /lib/systemd/system/vector.service для того, чтобы:
```

```
# Vector запускался с root правами. Необходимо закомментировать строки User
и Group в разделе Service.
# Vector писал логи в файл (для отделения логов Vector от логов основных
систем).
> sudo vi /lib/systemd/system/vector.service

#/lib/systemd/system/vector.service
...
[Service]
#User=vector
#Group=vector
StandardOutput=file:/var/log/astra-monitoring/vector.log
ExecStartPre=/usr/bin/vector validate
...

# Создать правила для ротации файлов логов Vector.
> sudo vi /etc/logrotate.d/astra-monitoring

# /etc/logrotate.d/astra-monitoring
/var/log/astra-monitoring/*.log {
    daily
    rotate 3
}

# Измените файл, чтобы указать для Vector директорию, которая будет содержать
правила обработки.
> sudo vi /etc/default/vector

# /etc/default/vector
VECTOR_CONFIG_DIR=/etc/vector/conf.d/
VECTOR_LOG=warn

# Создайте (если директория создана, то предварительно удалите все файлы toml)
директорию /etc/vector/conf.d/ и скопируйте предоставленные конфигурационные
файлы (vector.yaml и все *.yaml) по пути /etc/vector/conf.d/.
> sudo mkdir /etc/vector/conf.d/

# Создайте директорию /var/log/astra-monitoring для хранения логов Vector.
> sudo mkdir /var/log/astra-monitoring

# Внести изменение в конфигурационный файл вектора, указав принимающий логи
сервер и порт
> sudo vi /etc/vector/conf.d/vector.yaml

#/etc/vector/conf.d/vector.yaml
...
[sinks.vector_cached]
type = "vector"
inputs = [ "aldpro_*", "vector_internal_logs_transformed" ]
address = "<IP_address>:<port>"
```

```
# Выполните команду reload для systemd.
> sudo systemctl daemon-reload

# Перезапустите сервис vector:
> sudo systemctl restart vector.service
```

3.4.5.1 Обзор конфигурационного файла Vector

Конфигурация Vector в am-agent осуществляется через YAML-файлы. Основная структура конфигурационного файла включает следующие секции:

- **sources:** Определяет источники данных, которые Vector собирает (например, файлы журналов, системные метрики).
- **transforms:** Определяет цепочки преобразований, применяемые к данным, собранным из источников.
- **sinks:** Определяет целевые системы, в которые отправляются обработанные данные.

3.4.5.2 Особенности конфигурации Vector в am-agent

В отличие от стандартной установки Vector, конфигурация Vector внутри am-agent имеет следующие ограничения и требования:

- Требования к именам transform: Имя transform, которое должно быть отправлено в sinks, должно начинаться с префикса final. Это необходимо для того, чтобы am-agent правильно маршрутизировал данные.

Создание конфигурационного файла Vector

Рассмотрим пример создания конфигурационного файла Vector для am-agent, который собирает журналы из файла /var/log/application.log, добавляет поле hostname и отправляет их в платформу.

3.4.5.2.1 Шаг 1: Определение источника (sources)

Секция sources определяет, откуда Vector будет собирать данные. В нашем примере мы будем собирать журналы из файла /var/log/application.log.

```
sources:
  application_logs:
    type: file
    include:
      - /var/log/application.log
    read_from: beginning
```

В этом примере:

- **application_logs** – уникальное имя для источника.
- **type: file** – указывает, что источник данных – файл.
- **path: /var/log/application.log** – определяет путь к файлу журнала.
- **read_from: beginning** - указывает Vector читать файл с самого начала, а не только новые строки.

3.4.5.2.2 Шаг 2: Определение преобразования (transforms)

Секция `transforms` определяет, как данные будут преобразованы перед отправкой в целевую систему. В нашем примере мы добавим поле `hostname` к каждому событию. Важно: имя `transform`, которое отправляется в `sinks`, должно начинаться с `final`.

```
transforms:
  final_add_hostname:
    type: remap
    inputs:
      - application_logs
    source: |
      .hostname = "${HOSTNAME}"
      . = .
```

В этом примере:

- `final_add_hostname` – уникальное имя для преобразования. Обратите внимание на префикс `final`.
- `type: remap` – указывает, что тип преобразования – переназначение (`remap`).
- `inputs: [application_logs]` – определяет, что этот `transform` будет применять преобразования к данным из источника `application_logs`.
- `source: | ...` – содержит код преобразования на языке VRL (Vector Remap Language). Здесь мы добавляем поле `hostname` к каждому событию, используя переменную окружения `HOSTNAME`.

3.4.5.2.3 Шаг 3: Определение целевой системы (sinks)

Глобально секцией `sinks` управляет агент, управлять ею отдельно в большинстве случаев не требуется.

3.4.5.2.4 Развертывание конфигурации в am-agent

- Сохраните конфигурационный файл в формате YAML внутри платформы `astra-monitoring` (например, `application.yaml`).

```
sources:
  application_logs:
    type: file
    include:
      - /var/log/application.log
    read_from: beginning
transforms:
  final_add_hostname:
    type: remap
    inputs:
      - application_logs
    source: |
      .hostname = "${HOSTNAME}"
```

```
. = .
```

- Добавить созданный файл в конфигурации агента

Сбор логов 91

+

Имя файла	Дата	Описание	Действия
application.yaml	2/13/2025, 11:28:19 AM		↓ ↻ ✎ ✖

Редактировать объект ✕

Объект **Настройки агента** Настройки коллектора

Имя объекта:

Агент **Логи**

Vector включен:

Версия vector:

Сбор логов с агента включен:

Vector конфиги:

Отменить

Сохранить

Устранение неполадок

- Проверьте логи am-agent на наличие ошибок, связанных с конфигурацией Vector.
- Убедитесь, что синтаксис YAML-файла верен.
- Проверьте, правильно ли указаны пути к файлам журналов и URL-адреса целевых систем.
- Убедитесь, что переменная окружения HOSTNAME определена.
- Используйте vector validate для проверки конфигурации.

3.4.6 Установка node-exporter

Node-exporter устанавливается на все хосты, независимо от продукта "Группы Астра".

Пакет node-exporter может быть установлен из репозитория:

```
> sudo apt install prometheus-node-exporter
```

При размещении сервера с установленным node-exporter за сетевым экраном, NAT или другим подобным компонентом, необходимо опубликовать порт 9100/TCP и убедиться в его доступности для Платформы мониторинга. Если серверов несколько, то порты могут быть опубликованы, например, как 19100, 29100 и т.п.

3.4.7 Установка windows-exporter

Windows-exporter устанавливается на хосты, на которых установлена Windows OS.

Для установки необходимо скачать и распаковать архив с экспортером (предварительно необходимо установить/скачать утилиту [wget](https://eternallybored.org/misc/wget/)⁵⁵ для Windows):

```
# Скачиваем опубликованный архив
> wget https://dl.astralinux.ru/am/generic/windows-exporter/0.25.1/windows-exporter.z
ip
# Распаковываем архив с помощью средств Windows
# Запускаем установку экспортера
> ./windows_exporter-0.25.1-amd64.msi
#Экспортер установится как сервис со стандартным набором коллекторов и будет доступен
на порту 9182
```

3.4.8 Установка systemd-exporter

Systemd-exporter устанавливается на все хосты, независимо от продукта "Группы Астра".

```
# Скачиваем опубликованный архив
> curl -sLo systemd-exporter.tgz https://dl.astralinux.ru/am/generic/systemd-
exporter/latest/systemd-exporter.tgz
> tar zxvf systemd-exporter.tgz
> cd ./systemd-exporter/

# Копируем исполняемый файл в каталог
> sudo cp ./systemd-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/systemd-exporter
> sudo chown root:root /usr/local/bin/systemd-exporter

# Копируем конфигурацию сервиса в системный каталог
```

⁵⁵ <https://eternallybored.org/misc/wget/>

```
> sudo cp ./systemd-exporter.service /usr/lib/systemd/system/systemd-exporter.service

# Редактируем параметры сервиса (описание параметров далее по тексту)
> vi ./systemd-exporter.conf
> sudo cp ./systemd-exporter.conf /etc/systemd-exporter.conf
```

Изменить конфигурационный файл с названием /etc/systemd-exporter.conf со следующим содержимым, изменив в нем параметры:

```
Services:
- includes: .*\.service # выбор всех сервисов
General:
Interval: 60 # интервал опроса 60 секунд
```

Пример конфигурационного файла:

```
base:
# address to run the exporter on, mandatory
address: 0.0.0.0:9558

# exporter uri to publish on, mandatory
metrics_path: /metrics

# frequency to retrieve metrics in seconds, mandatory
interval: 60s

# file path to write logs (default: empty and disabled log to file)
log_file: /var/log/astra-monitoring/astra-systemd-exporter.log

# minimum log level (default: debug)
log_level: error

# disable printing log to stderr (default: false)
# disable_console_log: true

# enable default go metrics (default: false)
go_metrics: false

# enable default process metrics (default: false)
proc_metrics: false

services:
- includes: .*\.service
```

3.4.9 Установка postgres-exporter

Установка postgres-exporter выполняется на сервере, где расположен компонент Postgres DB.

Для установки необходимо скачать и распаковать архив с настройками и экспортером:

```
# Скачиваем опубликованный архив
> curl -sLo postgres-exporter.tgz https://dl.astralinux.ru/am/generic/postgres-exporter/0.15.0/postgres-exporter.tgz
> tar zxvf postgres-exporter.tgz > cd ./postgres-exporter/
# Копируем исполняемый файл в каталог
> sudo cp ./postgres-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/postgres-exporter
> sudo chown root:root /usr/local/bin/postgres-exporter
# Копируем конфигурацию сервиса в системный каталог
> sudo cp ./postgres-exporter.service /usr/lib/systemd/system/postgres-exporter.service
# Создаем файл со строкой подключения к postgres
> sudo vim /etc/default/postgres-exporter.default
# Разрешаем и включаем сервис
> sudo systemctl enable postgres-exporter.service
> sudo systemctl start postgres-exporter.service
> sudo systemctl status postgres-exporter.service
```

Файл /etc/default/postgres-exporter выглядит следующим образом:

```
# /etc/default/postgres-exporter.default
# This file can theoretically contain a bunch of environment variables
DATA_SOURCE_NAME="postgres://rupost:rupost@?
sslmode=disable&dbname=rupost&host=localhost&port=5432"
```

Замените имя пользователя, пароль и имя БД на необходимые.

После запуска службы, метрики будут доступны на порту 9187.

3.4.10 Установка sql-exporter

Установка sql-exporter выполняется на сервере, где расположен компонент Postgres DB. Этот экспортер позволяет формировать метрики с любыми sql запросами к базе.

Для установки необходимо скачать и распаковать архив с настройками и экспортером:

```
# Скачиваем опубликованный архив
> curl -sLo sql-exporter.tgz https://dl.astralinux.ru/am/generic/sql-exporter/0.14.1/sql-exporter.tgz
> tar zxvf sql-exporter.tgz > cd ./sql-exporter/
# Копируем исполняемый файл в каталог
> sudo cp ./sql-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/sql-exporter
> sudo chown root:root /usr/local/bin/sql-exporter
# Копируем конфигурацию сервиса в системный каталог
> sudo cp ./sql-exporter.service /usr/lib/systemd/system/sql-exporter.service
# Копируем конфигурационный файл
```

```

> sudo cp ./sql-exporter.yml /etc/sql-exporter.yml
# Копируем файл метриками
> sudo cp ./postgres_standard.collector.yml /etc/postgre_standard.collector.yml
# Разрешаем и включаем сервис
> sudo systemctl enable sql-exporter.service
> sudo systemctl start sql-exporter.service
> sudo systemctl status sql-exporter.service

```

В конфигурационном файле `sql-exporter.yml` необходимо обратить внимание на следующие параметры (заменить имя пользователя, пароль и имя БД на необходимые):

```

target:
  # Data source name always has a URI schema that matches the driver name. In some
  cases (e.g. MySQL)
  # the schema gets dropped or replaced to match the driver expected DSN format.
  data_source_name: 'postgres://rupost:rupost@localhost:5432/rupost?sslmode=disable'

```

После запуска службы, метрики будут доступны на порту 9399.

3.4.11 Установка `script_exporter`

Ниже рассматривается вариант установки экспортера как сервис. Для запуска через Агент см. [раздел 6.16](#) (see page 190).

Установка `script-exporter` выполняется на сервере, где необходимо выполнить запуск пользовательского сценария (скрипта) для сбора мониторинговых данных.

Для установки необходимо скачать и распаковать архив с настройками и экспортером:

```

# Скачиваем опубликованный архив
> curl -sLo script-exporter.tgz https://dl.astralinux.ru/am/generic/sql-exporter/2.18
.0/script-exporter.tgz
> tar zxvf script-exporter.tgz > cd ./script-exporter/
# Копируем исполняемый файл в каталог
> sudo cp ./script-exporter /usr/local/bin/
> sudo chmod 755 /usr/local/bin/script-exporter
> sudo chown root:root /usr/local/bin/script-exporter
# Копируем конфигурацию сервиса в системный каталог
> sudo cp ./script-exporter.service /usr/lib/systemd/system/script-exporter.service
# Копируем конфигурационный файл
> sudo cp ./script-exporter.yml /etc/script-exporter.yml
# Разрешаем и включаем сервис
> sudo systemctl enable script-exporter.service
> sudo systemctl start script-exporter.service
> sudo systemctl status script-exporter.service

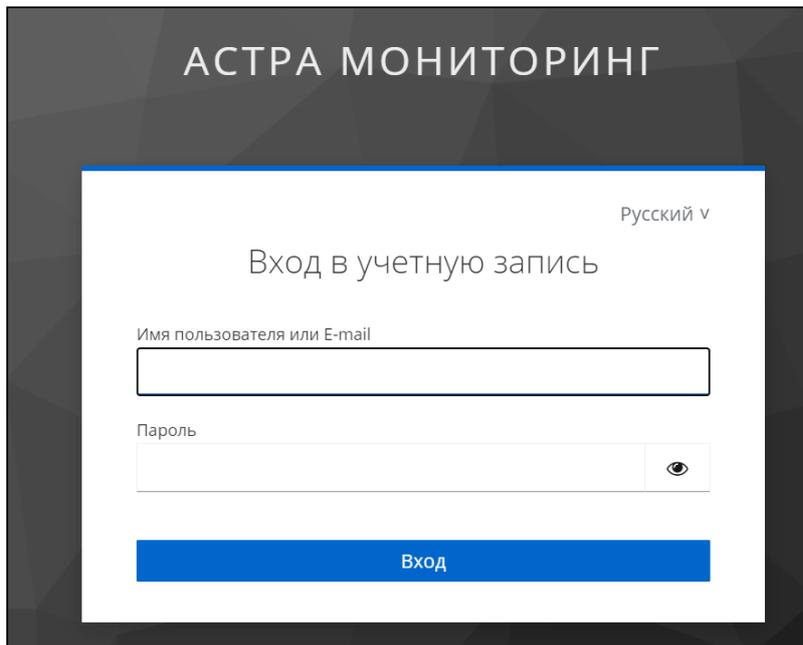
```

После запуска службы, метрики будут доступны на порту 9469.

4 4. Настройка Платформы АМ

4.1 Вход в Платформу

При обращении к URL AdminUI пользователь автоматически будет перекинут на сервис Keycloak, где он должен выполнить вход. В случае настроенной интеграции с каталогом пользователей LDAP, пользователь должен использовать свои персональные УЗ.



По умолчанию, в Платформе создаётся локальный (с точки зрения Keycloak) пользователь `admin-internal` с паролем, заданным при установке (через указание пароля `admin_ui.password` или через указание имени секрета `admin_ui.secretName`)

Для смены пароля в Keycloak:

1. В:
 - a. Кластере: необходимо перейти по адресу <http://адрес сервера мониторинга⁵⁶/oidc/> (пример: <https://admin-ui-mon-mon-mon-mon.10-177-181-74.sslip.io/oidc/>⁵⁷).
 - b. `docker-compose` необходимо перейти по адресу <http://адрес сервера мониторинга⁵⁸:8081/> (пример <http://192.168.0.103:8081/>).
2. После открытия окна с Keycloak нажать "Administration Console".
3. Ввести логин и пароль (по умолчанию `admin:password`).

⁵⁶ <http://flhtc/>

⁵⁷ <https://admin-ui-mon-dev.10-177-181-13.sslip.io/oidc/>

⁵⁸ <http://flhtc/>

4. Если необходимо сменить:
 - a. логин и пароль пользователя для входа в Keycloak, то выбрать в верхнем левом углу realm "master" → Перейти на вкладку "Users" → Кликнуть по "Username=admin" → Перейти на вкладку "Credentials" → Нажать "Reset password" → Далее сбросить пароль на новый → можно осуществлять вход в Keycloak по новому паролю.
 - b. логин и пароль для входа в Admin UI, то выбрать в верхнем левом углу realm "astra-monitoring" → Перейти на вкладку "Users" → Кликнуть по "Username=admin-internal" → Перейти на вкладку "Credentials" → Нажать "Reset password" → Далее сбросить пароль на новый → можно осуществлять вход в Admin UI по новому паролю.
5. Если необходимо создать:
 - a. новый логин и пароль пользователя для входа в Keycloak, то выбрать в верхнем левом углу realm "master" → Перейти на вкладку "Users" → Нажать "Add user" → Заполнить поле "Username" → Нажать кнопку "Create" → Перейти на вкладку "Credentials" → Нажать "Set password" → Создать пароль и сохранить его → Перейти на вкладку "Role mapping" → Нажать "Assign role" и выбрать "admin" и "default-roles-master" (можно выбрать те роли, которые необходимы) → можно осуществлять вход в Keycloak по новому логину и паролю.
 - b. новый логин и пароль для входа в Admin UI, то выбрать в верхнем левом углу realm "astra-monitoring" → Перейти на вкладку "Users" → Нажать "Add user" → Заполнить поле "Username" → Нажать кнопку "Create" → Перейти на вкладку "Credentials" → Нажать "Set password" → Создать пароль и сохранить его → можно осуществлять вход в Admin UI по новому логину и паролю.

Для увеличения времени действия сессии в Keycloak:

1. Перейти на адрес keycloak.
2. Изменить параметры в разделе `/admin/master/console/#/astra-monitoring/realm-settings/sessions`:
 - a. SSO Session Idle (12 часов)
 - b. SSO Session Max (12 часов)
3. Изменить параметры в разделе `admin/master/console/#/astra-monitoring/realm-settings/tokens`:
 - a. Access Token Lifespan (12 часов)
 - b. Access Token Lifespan For Implicit Flow (12 часов)
 - c. Client Login Timeout (5 минут)

KEYCLOAK

master

General Login Email Themes Keys Events Localization Security defenses Sessions **Tokens** Client policies User registration

General

Default Signature Algorithm: RS256

OAuth 2.0 Device Code Lifespan: 10 Minutes

OAuth 2.0 Device Polling Interval: - 5 +

Short verification_uri in Device Authorization flow: Short verification_uri in Device Authorization flow

Refresh tokens

Revoke Refresh Token: Disabled

Access tokens

Access Token Lifespan: 12 Hours
It is recommended for this value to be shorter than the SSO session idle timeout: 30 minutes

Access Token Lifespan For Implicit Flow: 12 Hours

astra-monitoring

Realm settings are settings that control the options for users, applications, roles, and groups in the current realm. [Learn more](#)

General Login Email Themes Keys Events Localization Security defenses **Sessions** Tok

SSO Session Settings

SSO Session Idle: 12 Hours

SSO Session Max: 12 Hours

SSO Session Idle Remember Me: Minutes

SSO Session Max Remember Me: Minutes

4.2 Настройка интеграции с LDAP

Интеграцию Keycloak с LDAP каталогом пользователей (ALD Pro, FreeIPA, MS AD) можно настроить двумя способами:

1) При установке Платформы - для этого необходимо раскомментировать блок `keycloak.ldap` в `values.yml` и при выполнении команды `helm upgrade --install ...` задать параметр

```
keycloak.ldap.bindCredential
```

```
keycloak:
  ldap:
    connectionUrl: ldap://ipaserver:389
    bindDn: "uid=integration,cn=users,cn=accounts,dc=example,dc=loc"
    usersDn: "cn=users,cn=accounts,dc=example,dc=loc"
    usernameLDAPAttribute: "uid"
```

В случае, если эти опции добавляются после установки Платформы, то необходимо прежде удалить `deployments` и `rvc`, относящиеся к `keycloak` - в случае установки в `Kubernetes`. Для настройки при развертывании `docker compose` необходимо удалить каталог `keycloak/data/pgdata`, где хранится стейт `Keycloak`.

2) Настроить самостоятельно через интерфейс `Keycloak`, согласно официальной [инструкции](#)⁵⁹.

4.3 Добавление объектов мониторинга



Добавление объекта в Платформу AM не влечет за собой автоматическую установку Агента.

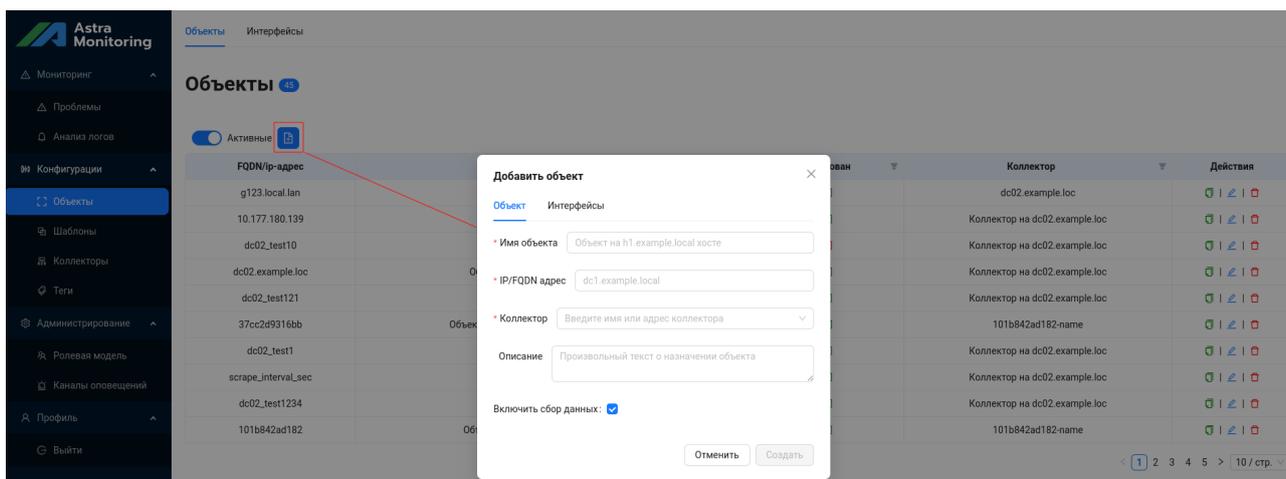
Необходимо заранее убедиться, что Агент, если он требуется для объекта, уже установлен на хосте объекта, запущен и работает, а также доступен для Платформы мониторинга.

Все требуемые Агенту управляемые им экспортеры, а также необходимые сторонние экспортеры, неуправляемые Агентом, могут в дальнейшем устанавливаться и конфигурироваться автоматически, обеспечивая сбор и выдачу соответствующих метрик.

Также необходимо заранее убедиться, что установлен, запущен, работает и доступен для Платформы мониторинга соответствующий Коллектор, через который будут передаваться диагностические данные с интерфейсов объекта.

Добавление объектов мониторинга производится в пользовательском web-интерфейсе, раздел Конфигурации → Объекты → кнопка добавления нового объекта:

⁵⁹ https://www.keycloak.org/docs/latest/server_admin/#_ldap



В открывшейся форме на вкладке "Объект" необходимо ввести:

	Параметр	Описание
1	Имя объекта	Имя объекта мониторинга, идентифицирующее объект в AM
2	FQDN/Ip-адрес	Адрес объекта мониторинга
3	Коллектор	Имя и адрес коллектора, на который интерфейсы объекта передают диагностические данные. Выбирается из списка зарегистрированных коллекторов
4	Описание	Описание объекта
5	Включить сбор данных	Показывает, включен сбор данных от объекта или нет

Далее необходимо перейти на вкладку "Интерфейсы" для настройки интерфейсов объекта. Если у объекта нет Агента, то доступны к конфигурированию только сторонние экспортеры, которые передают данные напрямую в коллектор.

Добавить объект ✕

Объект **Интерфейсы**

Имя объекта:

Источники данных/сторонние экспортеры

Добавить экспортер

По кнопке "Добавить экспортер" открывается форма для настройки стороннего экспортера:

Добавить экспортер ✕

* Имя стороннего экспортера:

* Адрес:порт сервера метрик: * URL путь метрик:

* Интервал сбора метрик, сек:

Лейблы для собираемых метрик:

Задаются следующие параметры:

	Параметр	Описание
1	Имя стороннего экспортера	Имя стороннего экспортера, вводится в поле руками

	Параметр	Описание
2	Адрес сервера метрик	В формате <ip-адрес>:<порт>
3	URL путь метрик	Начинается с "/"
4	Интервал сбора метрик, сек	Целое неотрицательное число
5	Лейблы для собираемых метрик	Формат Ключ:Значение. По этим лейблам данные метрик будут фильтроваться для показа графиков на дашбордах (см. пояснение ниже)

При наличии установленного Агента у объекта запись объекта создается автоматически с параметрами Агента и Коллектора, на который Агент сконфигурирован при установке:

Редактировать объект ✕

Объект Интерфейсы

* Имя объекта

* IP/FQDN адрес

* Коллектор ▾

Описание

Включить сбор данных:

На вкладке "Интерфейсы" автоматически отобразятся 2 субвкладки "Агент" и "Exporter". Субвкладка "Агент" при этом открыта дефолтно. На ней уже представлены управляемые Агентом экспортеры, которые установлены и сконфигурированы вместе с ним:

Редактировать объект ✕

Объект Интерфейсы

Агент Exporter

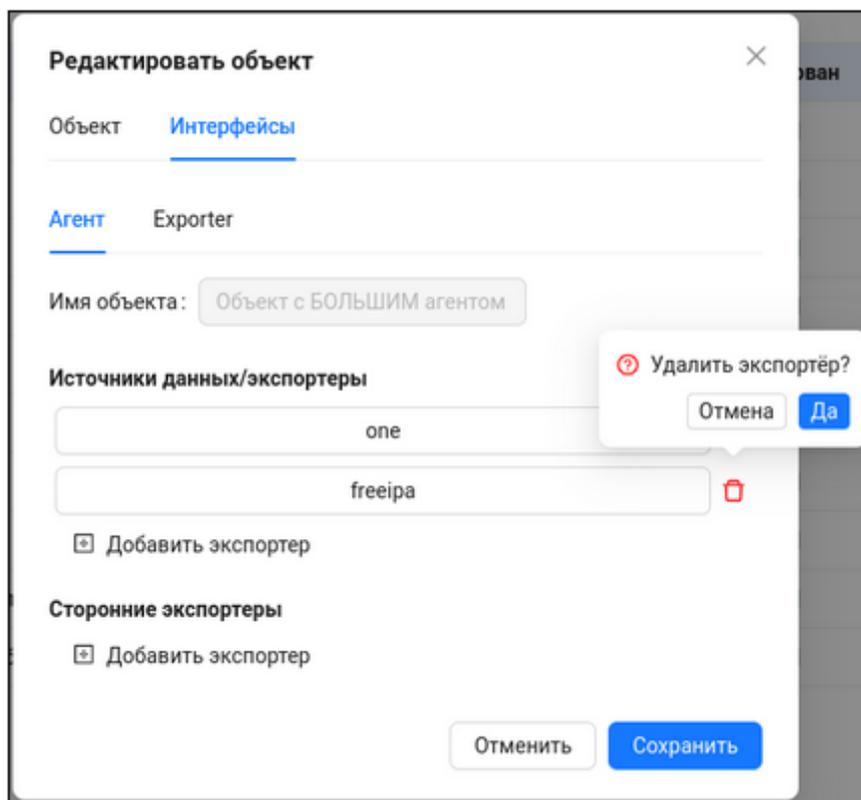
Имя объекта:

Источники данных/экспортеры

<input type="text" value="one"/>	
<input type="text" value="freeipa"/>	

Сторонние экспортеры

Если в списке экспортеров присутствуют лишние, их можно удалить, последовательно кликая по соответствующим кнопкам "Удалить" и подтверждая операцию:



При необходимости добавить экспортер под управление Агентом, кликнуть по кнопке "Добавить экспортер" в секции "Источники данных/экспортеры" (для внутренних экспортеров) либо в секции "Сторонние экспортеры" (для сторонних экспортеров).

Для редактирования настроек уже имеющихся экспортеров нужно кликнуть по имени соответствующего экспортера в списке, откроется форма с полями:

- для внутреннего экспортера

Редактировать агент ✕

* Имя экспортера:

Labels:

:

:

Config:

:

Интервал, сек: go_metrics:

Редактировать агент ✕

* Имя экспортера:

Labels:

:

:

Config:

:

Интервал, сек: go_metrics:

- systemd
- freeipa
- one**
- rupost
- rubackup
- termidesk
- node

	Параметр	Описание
1	Имя экспортера	Имя внутреннего экспортера, выбирается из списка доступных экспортеров
2	Labels	Лейблы для собираемых метрик. Формат Ключ:Значение. По этим лейблам данные метрик будут фильтроваться для показа графиков на дашбордах (см. пояснение ниже)
3	Config	Параметры конфигурационного файла экспортера. Формат Ключ:Значение
4	Интервал, сек	Интервал сбора метрик в секундах. Целое неотрицательное число
5	go_metrics	Включает дефолтные метрики, собираемые пакетом Prometheus на go, в которых информация о ресурсах, потребляемых экспортером

- для стороннего экспортера форма аналогична описанной выше для настроек стороннего экспортера вне Агента.

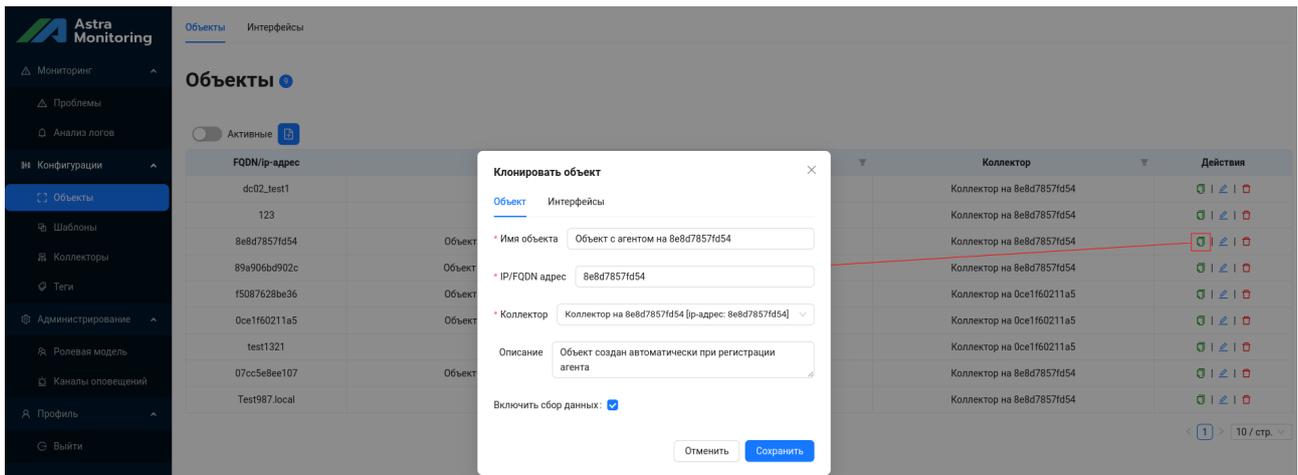
! Добавленные через UI экспортеры в дальнейшем автоматически устанавливаются из централизованного хранилища Платформы на хост объекта.

i Для корректного использования предустановленных дашбордов в Grafana необходимо использовать специальные теги `product` и `component`. Они позволяют корректно фильтровать необходимые объекты и показывать графики для нужных продуктов. В данный момент такие теги необходимо создавать и назначать вручную на соответствующие объекты. В главе 6 документации для каждого продукта даны необходимые пояснения по созданию и использованию данных тегов. Указанная особенность касается только использования этих двух тегов.

После ввода всех необходимых параметров необходимо нажать кнопку "Сохранить", в результате чего новый созданный объект появится в общем списке объектов мониторинга. В дальнейшем параметры объекта можно изменить с помощью пользовательского интерфейса.

Для простого добавления новых объектов на основе уже существующих предусмотрена кнопка клонирования объекта. В открывшейся форме достаточно поменять какие-то из параметров вместо полного описания всех полей. После нажатия кнопки "Сохранить", новый объект появится в интерфейсе Платформы АМ.

! Платформа не допускает дубликатов объектов с одинаковыми именами и FQDN/Ip-адресами.



Из итогового списка объектов мониторинга затем формируется список targets в формате json, включающий в себя имя (hostname), адрес инстанса (instance) и теги (labels) каждого объекта мониторинга. Этот список доступен по специальному адресу `<admin-ui-url>/api/endpoints?format=prometheus`. Prometheus обращается к указанному адресу, считывает список объектов и, посредством механизма [HTTP Service Discovery](#)⁶⁰, добавляет их в свою конфигурацию, после чего начинает опрос и получение метрик от объектов мониторинга.



Примечание к релизу

Управление объектами мониторинга в рамках подхода IaC (Infrastructure-as-Code) не поддерживается на данном этапе и будет рассмотрено в следующих релизах.

4.4 Настройка триггеров для создания событий

Правила триггеров для создания событий описаны в конфигурационных файлах:

- в директории `./alert-rules/` в случае установки в кластер Kubernetes (см. [Установка и обновление Astra Monitoring в кластер Kubernetes](#)).
- в директории `./vmalert/config/` в случае установки в виде Docker Compose (см. [Установка и обновление Astra Monitoring с использованием Docker Compose](#)).

Файлы логически разделены по функциональным группам, например, `ald-pro.yaml`, `brest.yaml` или `node-exporter.yaml`.

Правила описываются в формате YAML следующим образом:

```
groups:
  - name: ald-pro
    interval: 30s
    concurrency: 2
```

⁶⁰ https://prometheus.io/docs/prometheus/latest/http_sd/

```

rules:
  - alert: IPATestUserConnection
    expr: |
      ipa_user_connect{} == 0
    for: 5m
    labels:
      severity: critical
      team: ipa_support
    annotations:
      summary: "Server {{ $labels.hostname }} has an issue with test user authentication"
      description: "Test user {{ $labels.user_check }} can't connect to domain on server {{ $labels.hostname }} ({{ $labels.instance }})"

  - alert: IPAUserConnectionTime
    expr: |
      avg_over_time(ipa_time_for_connect[5m]) > 2 *
      avg_over_time(ipa_time_for_connect[5m] offset 5m)
    labels:
      severity: warning
      team: ipa_support
    annotations:
      summary: "Server {{ $labels.hostname }} ({{ $labels.instance }}) has highly increased test user connection time ({{ humanize $value }} ms.)"
      description: "Test user {{ $labels.user_check }} has an issue with very long test user connection time ({{ humanize $value }} ms.) on server {{ $labels.hostname }} ({{ $labels.instance }})"

```

где:

- name - имя группы правил,
- interval - интервал обработки правил группы,
- alert - имя триггера,
- expr - условие для срабатывания триггера в формате [PromQL](#)⁶¹,
- for - минимальная длительность состояния перед срабатыванием триггера,
- labels - дополнительные лейблы для триггера:
 - severity - уровень критичности,
 - team - маршрут для отправки уведомлений о срабатывании триггера, например, имя команды поддержки системы/продукта. (дополнительное опциональное поле, приведенное в качестве примера для конкретного набора правил),
- annotations - дополнительные информационные сообщения для получателей сообщений:
 - summary - краткое описание проблемы в заголовке сообщения,
 - description - общее описание проблемы в тексте сообщения.

Набор полей, указанных в списке labels может отличаться, в зависимости от правила, например, включать в себя имя сервиса systemd, точку монтирования файловой системы mountpoint, имя пользователя, название метрики и т.д., но при этом рекомендуется всегда указывать уровень критичности severity.

⁶¹ <https://prometheus.io/docs/prometheus/latest/querying/basics/>

Для добавления нового правила к уже существующей группе, необходимо определить имя триггера, условия для его срабатывания, критичность и получателя сообщений о срабатывании триггера. Далее добавить описание триггера в формате YAML в список **rules** требуемой группы правил, например, группы `ald-pro` в файле `./alert-rules/ald-pro.yaml` (`./vmaalert/config/ald-pro.yaml`).

Для создания нового набора правил необходимо создать новый файл в директории `./alert-rules/` (`./vmaalert/config/`) с требуемой группой (например, файл `rupost.yaml` с группой правил `rupost`), описать в группе требуемый набор правил и сохранить файл.

Также правила могут быть размещены в файле `values.tpl`, в секции `vmaalert / additionalPrometheusRulesMap`, аналогично описанному выше.

После окончания добавления или редактирования правил, необходимо обновить компоненты Платформы с помощью инструментов `docker-compose / helm-chart` в соответствии с описанием в разделах [Установка и обновление Astra Monitoring с использованием Docker Compose](#) и [Установка и обновление Astra Monitoring в кластер Kubernetes](#).

4.5 Настройка оповещений

Правила оповещений о событиях описаны в конфигурационном файле:

- в директории `./templates/alertmanager-ctm.yaml` в случае установки в кластер Kubernetes (см. [Установка и обновление Astra Monitoring в кластер Kubernetes](#)).
- в директории `./alertmanager/config/alertmanager.yml` в случае установки в виде Docker Compose (см. [Установка и обновление Astra Monitoring с использованием Docker Compose](#)).

Настройки описываются в формате YAML. Например, для интеграции с Mattermost необходимо подставить `<webhookid>` и `<channel-name>` в блок `slack_configs` группы `receivers` и т.д. Имя канала оповещений (`receiver`) задается произвольно. Неиспользуемые каналы можно удалить из конфигурации.

```
global:
templates:
- '/etc/alertmanager-templates/*.tpl'
route:
  receiver: alert-null # канал для отправки всех сообщений по умолчанию.
  # В примере указана пустая заглушка для отправки сообщений только по указанным
  # далее маршрутам, но может быть использован любой другой канал.
  group_by: ['alertname', 'hostname', 'group', 'severity', 'metric']
  group_wait: 3m
  group_interval: 5m
  repeat_interval: 1d
  routes:
  - receiver: alert-mmmost
    matchers: # задаем условия для отправки сообщений через этот канал
    - team = ipa_support
  - receiver: alert-telegram1
    matchers:
    - group = infra
  - receiver: send-to-telegram2
    matchers:
```

```

- group = middleware
- severity =~ "warning|critical"
- receiver: email-alerts
  matchers:
  - group = middleware
  - severity = critical

receivers:
- name: alert-mmost1
  slack_configs:
  - api_url: https://im.astralinux.ru/hooks/<webhookid>
    channel: '<channel-name>' # ИМЯ КАНАЛА В mattermost
    send_resolved: true
    title: '{{ template "slack.title" . }}'
    color: '{{ template "slack.color" . }}'
    text: '{{ template "slack.text" . }}'
- name: alert-telegram1
  # Оповещения в первую группу telegram
  telegram_configs:
  - bot_token: '<bot_token>' # https://core.telegram.org/bots/features#botfather
    api_url: 'https://api.telegram.org'
    # Не забудьте добавить бота в группу, в которую должны приходить сообщения
    chat_id: -1000000000001 # <указать id группы в формате -1234567890, в начале
    # СТОИТ МИНУС
    send_resolved: true
    message: '{{ template "telegram.message" . }}'
- name: send-to-telegram2
  # Оповещения во вторую группу telegram (при необходимости)
  telegram_configs:
  - bot_token: '<bot_token>'
    api_url: 'https://api.telegram.org'
    chat_id: -1000000000002
    send_resolved: true
    message: '{{ template "telegram.message" . }}'
- name: email-alerts
  email_configs:
  - to: email@example-domain.ru
    send_resolved: false
    from: astra-monitoring@example-domain.ru
    smarthost: <smtp-relay-address>:25
    require_tls: false
- name: alert-null # пустой канал-заглушка

inhibit_rules:
- source_matchers: [severity="critical"]
  target_matchers: [severity="warning"]
  # Apply inhibition if the alertname is the same.
  # CAUTION:
  # If all label names listed in `equal` are missing
  # from both the source and target alerts,
  # the inhibition rule will apply!
  equal: [alertname, hostname, instance, group]

```

Из "коробки" поддерживается много разных типов каналов оповещений, например:

- email
- Telegram
- Discord
- Jira
- Mattermost (Slack)
- MS Teams
- OpsGenie
- PagerDuty
- VictorOps
- SNS (Amazon AWS)
- Webex
- Webhook

Указанный в списке механизм Webhook позволяет расширить список возможностей интеграции, если требуемый канал оповещений не поддерживается нативно Alertmanager. Так, в списке совместимых интеграций заявлены GitLab, Ansible Tower, IRC, Zoom, SMS (с использованием web-сервисов провайдеров) и прочие продукты (<https://prometheus.io/docs/operating/integrations/#alertmanager-webhook-receiver>)

Набор конкретных используемых каналов оповещений задается в блоке **receivers**. Далее для каждого канала в отдельности задается имя и специфичные параметры, такие как адрес получателя email, token бота, id группы Telegram и так далее. Может быть несколько каналов одного типа, но с разными параметрами, например, разными группами в Telegram или получателями email.

Выбор того или иного канала оповещений зависит от условий, заданных в секции **route**, включая канал "по умолчанию" (в примере выше это alert-null), куда будут отправляться все оповещения. Для остальных маршрутов условия задаются в блоке **matchers** маршрута. Например, это может быть проверка на значение каких-либо лейблов - severity, group и т.д. Оповещение может быть отправлено в несколько разных каналов. Так, в примере выше описана отправка в telegram2 алертов, у которых лейбл group = "middleware", а severity = "warning" или "critical". При этом, алерты с лейблами group = "middleware" и severity = "critical" будут также дублироваться на электронную почту из канала email-alerts.

Оповещения можно группировать по различным признакам для уменьшения их количества, например, объединить все алерты от одного хоста в одно сообщение. Группировка задается параметром **group_by**, который определяет лейблы, по совпадению значений которых будут группироваться алерты. Дополнительно применяются параметры **group_wait** - задержка для начальной группировки алертов перед первым оповещением, а также **group_interval** - через какое время отправлять дополнительные оповещения, если в эту же группу добавлены новые алерты. Параметр **repeat_interval** задает время для повторной отправки оповещения о все еще активной проблеме, как напоминание.

Секция **inhibit_rules** позволяет задать правила "подавления" одних алертов другими. Например, при наличии алерта с лейблом severity="critical", Alertmanager не будет реагировать на приходящие алерты с такими же лейблами [alertname, hostname, instance, group], но с severity="warning".

Активные настройки Alertmanager, его состояние и список активных алертов можно проверить в web-интерфейсе, по доменному имени, заданному при развертывании в среде Kubernetes (например, вида <https://alertmanager.am.domain.local/>) или по порту 9093 при развертывании в среде Docker Compose (вида http://<server_address>:9093/). Также, используя API Alertmanager возможно проверять его состояние GET-запросами по пути `/-/healthy`, `/-/ready` или инициировать перезагрузку конфигурации POST-запросом по пути `/-/reload`.

Более подробно о возможностях настройки подсистемы оповещения можно найти в [официальной документации](#)⁶².

Конфигурация `webhook_configs` должна быть сохранена для наполнения базы событий и отображения актуальной информации в Admin UI.

После внесения изменений, необходимо обновить компоненты Платформы с помощью инструментов `docker-compose` / `helm-chart` в соответствии с описанием в разделах [Установка и обновление Astra Monitoring с использованием Docker Compose](#) и [Установка и обновление Astra Monitoring в кластер Kubernetes](#).

4.6 Настройка (добавление) дополнительных дашбордов Grafana

Дополнительные пользовательские дашборды Grafana могут быть добавлены в Платформу AM на этапе развертывания. Для этого заранее подготовленный код дашборда в формате json необходимо добавить в файл `values.yaml` и затем провести обновление компонентов AM с помощью инструментов `helm-chart` в соответствии с описанием в разделе [Установка и обновление Astra Monitoring в кластер Kubernetes](#).

Формат описания дополнительных дашбордов в `values.yaml` состоит из двух частей: 1) провайдера и 2) группы дашбордов с содержимым в виде непосредственно json-кода или в виде ссылки на него (в том же формате json). Добавление дашбордов с несуществующим провайдером может вызвать ошибки при развертывании Платформы.

Примерный фрагмент файла `values.yaml`:

```
grafana:
  dashboardProviders:
    dashboardproviders.yaml:
      apiVersion: 1
      providers:
        - name: 'custom-dashboards' # имя провайдера дашбордов
          orgId: 1
          folder: ''
          type: file
          disableDeletion: false
          updateIntervalSeconds: 60
          editable: true
          allowUiUpdates: true
          options:
            path: /src/grafana/dashboards/
            foldersFromFilesStructure: true
        - name: 'web-dashboards'
          orgId: 1
          folder: ''
          type: file
          disableDeletion: false
          updateIntervalSeconds: 60
          editable: true
          allowUiUpdates: true
```

⁶² <https://prometheus.io/docs/alerting/latest/configuration/>

```

options:
  path: /var/lib/grafana/dashboards/Web-dashboards
  foldersFromFilesStructure: true
dashboards:
  custom-dashboards: # совпадает с именем провайдера в dashboardProviders
  mydashboard: # название дашборда
    json: |
      { } # код дашборда в формате json
  json-dashboard-2:
    json: |
      { }
  web-dashboards: # совпадает с
именем провайдера в dashboardProviders
  mydashboard-3: # название
дашборда
  url: http://externalurl:8080/grafana/mydashboard.json # путь к
загружаемому дашборду

```

Подробное описание провайдеров и дашбордов можно найти в [документации Grafana](#)⁶³ и на [странице GitHub](#)⁶⁴.

Дашборды, загружаемые с помощью url, должны быть доступны для Платформы во время развертывания, так как при этом происходит загрузка (скачивание) файла во внутреннюю директорию Grafana, указанную в настройках соответствующего провайдера.

Примечание: стоит учесть, что после удаления дашбордов или изменения путей у провайдеров, при повторном развертывании Платформы без удаления рvc Grafana может сложиться ситуация, когда дашборды появляются в новой директории и остаются в старой. Решением может оказаться ручное удаление рvc Grafana перед повторным развертыванием Платформы AM.

4.7 Использование API

Для того, чтобы иметь доступ к выполнению API запросов необходимо аутентифицироваться в системе аутентификации (Keycloak), при этом будет получен токен доступа (access_token), обеспечивающий дальнейшую работу с API.

```

> export USER_NAME="<your_user_name>"
> export USER_PASSWORD="<your_user_password>"
> export ADMINUI_ENDPOINT="<url_to_your_admin_ui>"

> curl --silent -k -X POST -d "grant_type=password&username=${USER_NAME}&password=${
USER_PASSWORD}&client_id=admin-ui" -H "Content-Type:application/x-www-form-
urlencoded" ${ADMINUI_ENDPOINT}/oauth/login | jq
{
  "token_type": "Bearer",

```

63 <http://docs.grafana.org/administration/provisioning/#dashboards>

64 <https://github.com/grafana/helm-charts/blob/main/charts/grafana/values.yaml>


```
Tan7vPfpWj+4wFtKnQPg9ijJGcpazVrrmpDZVU7YS5TFOwnA3K4kz9dlsK6kWjokWP9PCzz/
chkz0Ld1WOUFOk30eL6j33j2vKc9cwnXOPTcTE3cI3ORBxu3eA
qc4ed+iq3JQAaOoHmFcEWwXQ5l4dSl8M/
mLbRkf50Op8nvFtRfbAu0sdAg5FpkLvprCk5iFJfYX5DCzGflsbu210BeXrIzc5ILLreG8cxLkbBDKRbPLJSh
+s+ykKQH0gv35Ci90NVXjoHx0nsBlXB7G7z5BAY8YNGNer9gkjsrWSSgbW4gYPGtdC1upIW/
cFvuv35intZbZocLzxx
HbPmaDE/3n/caiXI5cxwmRz5FqvNz8i/ly4ZjWSpk+gnC2zlrpEn382L65WTSQ",
  "expires_in": 300,
  "scope": "openid audience-admin-ui profile email"
}
```

После получения токена можно выполнять запросы к API, например:

```
> export USER_NAME="<your_user_name>"
> export USER_PASSWORD="<your_user_password>"
> export ADMINUI_ENDPOINT="<url_to_your_admin_ui>"

> export access_token=`curl --silent k -X POST -d "grant_type=password&username=${
USER_NAME}&password=${USER_PASSWORD}&client_id=admin-ui" -H "Content-
Type:application/x-www-form-urlencoded" ${ADMINUI_ENDPOINT}/oauth/login | jq -r
'.access_token'`

> curl --silent -k -H "Authorization: Bearer ${access_token}" ${ADMINUI_ENDPOINT}/
api/endpoints | jq
[
  {
    "id": 19,
    "instance": "<ip_address>:9100",
    "fqdn": "<fqdn_host>",
    "tags": [
      {
        "id": 16,
        "name": "group",
        "value": "ald-pro",
        "color": "#5aafed",
        "endpoint_count": null
      },
      {
        "id": 20,
        "name": "subsystem",
        "value": "controller",
        "color": "#f7afb0",
        "endpoint_count": null
      }
    ],
    "type": {
      "id": 4,
      "name": "Astra Linux",
      "agent_config": "",
      "template_url": "http://{{ .grafana_base_url }}/d/rYddd1PWkwwkkD453/node-
exporter-full?orgId=1&var-hostname={{ .hostname }}&var-group={{ .group }}",
      "default_port": 9100
    }
  }
]
```

```

    },
    "deleted_at": null,
    "url": "http://grafana-mon-dev.10-177-181-13.sslip.io/d/rYdddlPWkwwkkD453/node-
exporter-full?orgId=1&var-hostname=<fqdn_host>&var-group=ald-pro"
  },
  ...
]

```

Описание существующего API можно найти по адресу `${ADMINUI_ENDPOINT}/api/v1/swagger/index.html`, для предлагаемых endpoint'ов необходимо добавить `/api/v1/` в начале.

The screenshot shows the Swagger UI for an API named 'doc.json'. The interface is organized into sections: 'general', 'Endpoints', and 'Events'. Each section contains a list of API endpoints with their respective HTTP methods and descriptions.

Method	Endpoint	Description
GET	/about	Info page
Endpoints		
POST	/endpoints/new	Create a new endpoint
GET	/endpoints/types	Retrieves list of types
DELETE	/endpoints/{id}	Delete an existing endpoint
PATCH	/endpoints/{id}	modify an existing endpoint
POST	/endpoints/{id}/hardDelete	Perform a hard delete but only if the component has been soft deleted. If the component is active the hard delete will fail with a component not deleted error response.
POST	/endpoints/{id}/undelete	Undo a soft delete of a component. Restores the component marked as deleted
Events		
GET	/events	Retrieves alerts

5 Резервное копирование

5.1 Резервное копирование базы данных ClickHouse



Примечание к релизу

Автоматическое резервное копирование реализовано только при установке в кластер Kubernetes.

В Платформе Astra Monitoring при установке в кластер Kubernetes предусмотрено создание автоматических резервных копий базы данных Clickhouse (CronJob).

По умолчанию, резервные копии создаются каждый день в 2:00 UTC. Полные резервные копии создаются каждую неделю (на 1-ый, 8-ой и т.д. день), далее создаются инкрементальные резервные копии.

Параметр `base_backup_copies_number` определяет, какое количество полных резервных копий и, следовательно, инкрементальных резервных копий хранить, т.е. глубину восстановления. Если `base_backup_copies_number=1`, то удастся восстановить данные за последние 7 дней, если `base_backup_copies_number=2`, то за последние 14.

Резервная копия сохраняется в PVC Kubernetes `"pvc-{{ $.Release.Name }}-clickhouse-backup"`.

Параметры работы сервиса по созданию резервных копий задаются в файле `values.yaml` ([Установка и обновление Astra Monitoring в кластер Kubernetes](#)):

```
...
# Описание конфигурации Clickhouse
clickhouse:
  ...
  # Включить регулярный бекап бд
  db_backup_enabled: "true"
  # Название и путь до образа
  db_backup_image: registry.astralinux.ru/am/ch-backup-db
  # Версия образа
  db_backup_tag: 0.21
  # Количество хранимых базовых бекапов
  base_backup_copies_number: 1
  # Таймаут для подключения к экземпляру бд (минуты)
  db_management_timeout: 5
  # Порт для подключения к экземпляру бд
  db_management_port: 9000
...
```

При установке с использованием Docker Compose, резервное копирование настраивается запуском по расписанию скрипта `ch-backup.sh`, находящегося в директории `astra-monitoring/clickhouse/tools`. Предварительно необходимо создать директорию для размещения архивных копий и указать ее полный путь в переменной `BACKUP_DIR` скрипта `ch-backup.sh`, например, "`BACKUP_DIR=/opt/am/astra-monitoring/clickhouse/backups`". Пример настройки запуска по расписанию:

```
mkdir /opt/am/astra-monitoring/clickhouse/backups
chmod +x /opt/am/astra-monitoring/clickhouse/tools/ch-backup.sh

crontab -e
# В открывшемся редакторе настроить расписание запуска скрипта и сохранить
0 2 * * * /opt/am/astra-monitoring/clickhouse/tools/ch-backup.sh
```

Алгоритм работы резервного копирования аналогичен описанному выше для кластера Kubernetes.

5.2 Резервное копирование базы данных PostgreSQL



Примечание к релизу

Автоматическое резервное копирование реализовано только при установке в кластер Kubernetes.

В Платформе Astra Monitoring при установке в кластер Kubernetes предусмотрено создание автоматических резервных копий базы данных PostgreSQL (CronJob).

По умолчанию, резервные копии создаются каждый день в 2:00 UTC, при этом создаются полные резервные копии (с использованием `pg_basebackup`), хранящиеся 3 дня. Создание резервных копий может быть отключено выставлением параметра `postgresql.db_backup_enabled` в `false`.

Резервная копия сохраняется в PVC Kubernetes "`pvc-{{ $.Release.Name }}-postgresql-backup`".

Параметры работы сервиса по созданию резервных копий задаются в файле `values.yaml` ([Установка и обновление Astra Monitoring в кластер Kubernetes](#)):

```
...
# Описание конфигурации PostgreSQL
postgresql:
  ...
  # Включить регулярный бекап бд
  db_backup_enabled: "true"
  # Таймаут для подключения к экземпляру бд (минуты)
  db_management_timeout: 5
  ...
```

6 6. Мониторинг продуктов

- [6.1 Мониторинг ОС Astra Linux \(node-exporter\)](#) (see page 116)
- [6.2 Мониторинг Windows Server](#) (see page 117)
- [6.3 Мониторинг PostgreSQL](#) (see page 134)
- [6.4 Мониторинг FreeIPA](#) (see page 134)
- [6.5 Мониторинг ALD Pro](#) (see page 135)
- [6.6 Мониторинг ПК СВ Брест](#) (see page 138)
- [6.7 Мониторинг RuPost](#) (see page 145)
- [6.8 Мониторинг RuBackup](#) (see page 146)
- [6.9 Мониторинг Termidesk](#) (see page 147)
- [6.10 Мониторинг Billmanager](#) (see page 149)
- [6.10 Триггеры для создания событий по логам](#) (see page 149)
- [6.11 Мониторинг Astra Kube](#) (see page 152)
- [6.12 Мониторинг SNMP устройств \(агент мониторинга\)](#) (see page 154)
- [6.13 Мониторинг устройств с помощью IPMI \(агент мониторинга\)](#) (see page 155)
- [6.14 Базовые правила для создания событий](#) (see page 158)
- [6.15 Мониторинг SSL сертификатов \(агент мониторинга\)](#) (see page 185)
- [6.16 Запуск и мониторинг локальных скриптов \(агент мониторинга\)](#) (see page 190)
- [6.17 Мониторинг SNMP трапов](#) (see page 194)
- [6.18 Приложения. Метрики, используемые для мониторинга](#) (see page 204)

6.1 6.1 Мониторинг ОС Astra Linux (node-exporter)



Данные с node-exporter могут использоваться для различных продуктов. Обратите внимание на использование тегов `product` и `component` в документации к конкретному продукту.

Для мониторинга системных показателей используется [node-exporter](#)⁶⁵.

Экспортер так же используется на дашбордах продуктов, для отображения системных показателей (например, ЦПУ, память) рядом с продуктовыми для удобства анализа.

Для отображения метрик node-exporter взят [дашборд](#)⁶⁶. Добавлена переменная `group`, для фильтрации хостов по группам, изменен ID дашборда, чтобы не конфликтовать с оригиналом.

Метрики, предоставляемые node-exporter, не описываются в данном документе.

⁶⁵ https://github.com/prometheus/node_exporter

⁶⁶ <https://grafana.com/grafana/dashboards/1860-node-exporter-full/>

6.2 6.2 Мониторинг Windows Server



Данные экспортера должны тегироваться тегом `product: windows`

Экспортер предназначен для мониторинга основных показателей хоста, на котором установлена Windows OS. Кроме этого, он может собирать данные с приложений, характерных для использования на серверных версиях данной операционной системы (IIS, Exchange, HyperV и пр.). Так же осуществляет мониторинг ролей Windows Server (AD, DNS, DHCP и пр.).

Экспортером поддерживаются версии Windows Server 2016 и выше, десктопные версии Windows 10 и 11. Для Windows Server более низких версий работа экспортера не гарантируется.

Экспортер поставляется в виде кода на [github](#)⁶⁷, exe или msi дистрибутива. При установке из msi файла экспортер устанавливается как windows service с коллекторами (получателями/сборщиками метрик от экспортера) по умолчанию.

При установке дополнительно можно указать следующие параметры:

Наименование	Описание
<code>ENABLED_COLLECTORS</code>	As the <code>--collectors.enabled</code> flag, provide a comma-separated list of enabled collectors
<code>LISTEN_ADDR</code>	The IP address to bind to. Defaults to 0.0.0.0
<code>LISTEN_PORT</code>	The port to bind to. Defaults to 9182.
<code>METRICS_PATH</code>	The path at which to serve metrics. Defaults to <code>/metrics</code>
<code>TEXTFILE_DIRS</code>	As the <code>--collector.textfile.directories</code> flag, provide a directory to read text files with metrics from
<code>REMOTE_ADDR</code>	Allows setting comma separated remote IP addresses for the Windows Firewall exception (allow list). Defaults to an empty string (any remote address).
<code>EXTRA_FLAGS</code>	Allows passing full CLI flags. Defaults to an empty string.

⁶⁷ https://github.com/prometheus-community/windows_exporter

Например, так:

```
msiexec /i <path-to-msi-file> ENABLED_COLLECTORS=os,iis LISTEN_PORT=5000
```

В установку, по умолчанию, входят сборщики метрик:

Наименование	Описание	Комментарий
cpu ⁶⁸	CPU usage	
cs ⁶⁹	"Computer System" metrics (system properties, num cpus/total memory)	
logical_disk ⁷⁰	Logical disks, disk I/O	
net ⁷¹	Network interface I/O	
os ⁷²	OS metrics (memory, processes, users)	
service ⁷³	Service state metrics	
system ⁷⁴	System calls	
textfile ⁷⁵	Read prometheus metrics from a text file	

Метрики, которые позволяют собирать дефолтные коллекторы:

Сборщик метрик CPU

68 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.cpu.md

69 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.cs.md

70 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.logical_disk.md

71 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.net.md

72 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.os.md

73 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.service.md

74 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.system.md

75 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.textfile.md

Наименование	Описание	Тип	Теги
windows_cpu_cstate_seconds_total	Time spent in low-power idle states	counter	core, state
windows_cpu_time_total	Time that processor spent in different modes (dpc, idle, interrupt, privileged, user)	counter	core, mode
windows_cpu_interrupts_total	Total number of received and serviced hardware interrupts	counter	core
windows_cpu_dpcs_total	Total number of received and serviced deferred procedure calls (DPCs)	counter	core
windows_cpu_clock_interrupts_total	Total number of received and serviced clock tick interrupts	counter	core
windows_cpu_idle_break_events_total	Total number of time processor was woken from idle	counter	core
windows_cpu_parking_status	Parking Status represents whether a processor is parked or not	gauge	core

Наименование	Описание	Тип	Теги
windows_cpu_core_frequency_mhz	Core frequency in megahertz	gauge	core
windows_cpu_processor_performance_total	Processor Performance is the number of CPU cycles executing instructions by each core; it is believed to be similar to the value that the APERF MSR would show, were it exposed	counter	core
windows_cpu_processor_mperf_total	Processor MPerf Total is proportionl to the number of TSC ticks each core has accumulated while executing instructions. Due to the manner in which it is presented, it should be scaled by 1e2 to properly line up with Processor Performance Total. As above, it is believed to be closely related to the MPERF MSR.	counter	core
windows_cpu_processor_rtc_total	RTC total is assumed to represent the 64Hz tick rate in Windows. It is not by itself useful, but can be used with <code>windows_cpu_processor_utility_total</code> to more accurately measure CPU utilisation than with <code>windows_cpu_time_total</code>	counter	core
windows_cpu_processor_utility_total	Processor Utility Total is a newer, more accurate measure of CPU utilization, in particular handling modern CPUs with variant CPU frequencies. The rate of this counter divided by the rate of <code>windows_cpu_processor_rtc_total</code> should provide an accurate view of CPU utilisation on modern systems, as observed in Task Manager.	counter	core
windows_cpu_processor_privileged_utility_total	Processor Privileged Utility Total, when used in a similar fashion to <code>windows_cpu_processor_utility_total</code> will show the portion of CPU utilization which is happening in privileged mode.	counter	core

Сборщик метрик CS

Наименование	Описание	Тип	Теги
windows_cs_logical_processors	Number of installed logical processors	gauge	None
windows_cs_physical_memory_bytes	Total installed physical memory	gauge	None
windows_cs_hostname	Labeled system hostname information	gauge	hostname, domain, fqdn

Сборщик метрик Logical Disk

Наименование	Описание	Тип	Теги
windows_logical_disk_requests_queued	Number of requests outstanding on the disk at the time the performance data is collected	gauge	volume
windows_logical_disk_avg_read_requests_queued	Average number of read requests that were queued for the selected disk during the sample interval	gauge	volume

Наименование	Описание	Тип	Теги
windows_logical_disk_avg_write_requests_queued	Average number of write requests that were queued for the selected disk during the sample interval	gauge	volume
windows_logical_disk_read_bytes_total	Rate at which bytes are transferred from the disk during read operations	counter	volume
windows_logical_disk_reads_total	Rate of read operations on the disk	counter	volume
windows_logical_disk_write_bytes_total	Rate at which bytes are transferred to the disk during write operations	counter	volume
windows_logical_disk_writes_total	Rate of write operations on the disk	counter	volume
windows_logical_disk_read_seconds_total	Seconds the disk was busy servicing read requests	counter	volume
windows_logical_disk_write_seconds_total	Seconds the disk was busy servicing write requests	counter	volume

Наименование	Описание	Тип	Теги
windows_logical_disk_free_bytes	Unused space of the disk in bytes (not real time, updates every 10-15 min)	gauge	volume
windows_logical_disk_size_bytes	Total size of the disk in bytes (not real time, updates every 10-15 min)	gauge	volume
windows_logical_disk_idle_seconds_total	Seconds the disk was idle (not servicing read/write requests)	counter	volume
windows_logical_disk_split_ios_total	Number of I/Os to the disk split into multiple I/Os	counter	volume

Сборщик метрик Net

Наименование	Описание	Тип	Теги
windows_net_bytes_received_total	Total bytes received by interface	counter	nic
windows_net_bytes_sent_total	Total bytes transmitted by interface	counter	nic

Наименование	Описание	Тип	Теги
windows_net_bytes_total	Total bytes received and transmitted by interface	counter	nic
windows_net_output_queue_length_packets	Length of the output packet queue (in packets). If this is longer than 2, delays occur.	gauge	nic
windows_net_packets_outbound_discarded_total	Total outbound packets that were chosen to be discarded even though no errors had been detected to prevent transmission	counter	nic
windows_net_packets_outbound_errors_total	Total packets that could not be transmitted due to errors	counter	nic
windows_net_packets_received_discarded_total	Total inbound packets that were chosen to be discarded even though no errors had been detected to prevent delivery	counter	nic
windows_net_packets_received_errors_total	Total packets that could not be received due to errors	counter	nic
windows_net_packets_received_total	Total packets received by interface	counter	nic

Наименование	Описание	Тип	Теги
windows_net_packets_received_unknown_total	Total packets received by interface that were discarded because of an unknown or unsupported protocol	counter	nic
windows_net_packets_total	Total packets received and transmitted by interface	counter	nic
windows_net_packets_sent_total	Total packets transmitted by interface	counter	nic
windows_net_current_bandwidth_bytes	Estimate of the interface's current bandwidth in bytes per second	gauge	nic

Сборщик метрик OS

Наименование	Описание	Тип	Теги
windows_os_info	Contains full product name & version in labels. Note that the <code>major_version</code> for Windows 11 is "10"; a build number greater than 22000 represents Windows 11.	gauge	product, version, major_version, minor_version, build_number

Наименование	Описание	Тип	Теги
windows_os_paging_limit_bytes	Total number of bytes that can be stored in the operating system paging files. 0 (zero) indicates that there are no paging files	gauge	None
windows_os_paging_free_bytes	Number of bytes that can be mapped into the operating system paging files without causing any other pages to be swapped out	gauge	None
windows_os_physical_memory_free_bytes	Bytes of physical memory currently unused and available	gauge	None
windows_os_time	Current time as reported by the operating system, in Unix time ⁷⁶ . See time.Unix() ⁷⁷ for details	gauge	None
windows_os_timezone	Current timezone as reported by the operating system. See time.Zone() ⁷⁸ for details	gauge	timezone
windows_os_processes	Number of process contexts currently loaded or running on the operating system	gauge	None

76 https://en.wikipedia.org/wiki/Unix_time

77 <https://golang.org/pkg/time/#Unix>

78 <https://golang.org/pkg/time/#Time.Zone>

Наименование	Описание	Тип	Теги
windows_os_processes_limit	Maximum number of process contexts the operating system can support. The default value set by the provider is 4294967295 (0xFFFFFFFF)	gauge	None
windows_os_process_memory_limit_bytes	Maximum number of bytes of memory that can be allocated to a process	gauge	None
windows_os_users	Number of user sessions for which the operating system is storing state information currently. For a list of current active logon sessions, see Logon ⁷⁹	gauge	None
windows_os_virtual_memory_bytes	Bytes of virtual memory	gauge	None
windows_os_visible_memory_bytes	Total bytes of physical memory available to the operating system. This value does not necessarily indicate the true amount of physical memory, but what is reported to the operating system as available to it	gauge	None
windows_os_virtual_memory_free_bytes	Bytes of virtual memory currently unused and available	gauge	None

Сборщик метрик Service

⁷⁹ https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.logon.md

Наименование	Описание	Тип	Теги
windows_service_info	Contains service information in labels, constant 1	gauge	name, display_name, process_id, run_as
windows_service_state	The state of the service, 1 if the current state, 0 otherwise	gauge	name, state
windows_service_start_mode	The start mode of the service, 1 if the current start mode, 0 otherwise	gauge	name, start_mode
windows_service_status	The status of the service, 1 if the current status, 0 otherwise	gauge	name, status

Сборщик метрик System

Наименование	Описание	Тип	Теги
windows_system_context_switches_total	Total number of context switches ⁸⁰	counter	None

⁸⁰ https://en.wikipedia.org/wiki/Context_switch

Наименование	Описание	Тип	Теги
windows_system_exception_dispatcher_total	Total exceptions dispatched by the system	counter	None
windows_system_processor_queue_length	Number of threads in the processor queue. There is a single queue for processor time even on computers with multiple processors.	gauge	None
windows_system_system_calls_total	Total combined calls to Windows NT system service routines by all processes running on the computer	counter	None
windows_system_system_uptime	Time of last boot of system	gauge	None
windows_system_threads	Number of Windows system threads ⁸¹	gauge	None

Сборщик метрик Textfile

⁸¹ [https://en.wikipedia.org/wiki/Thread_\(computing\)](https://en.wikipedia.org/wiki/Thread_(computing))

Наименование	Описание	Тип	Теги
windows_textfile_scrape_error	1 if there was an error opening or reading a file, 0 otherwise	gauge	None
windows_textfile_mtime_seconds	Unix epoch-formatted mtime (modified time) of textfiles successfully read	gauge	file

Дополнительно можно подключить следующие сборщики метрик:

Наименование	Описание	Комментарий
ad ⁸²	Active Directory Domain Services	
adcs ⁸³	Active Directory Certificate Services	
adfs ⁸⁴	Active Directory Federation Services	
cache ⁸⁵	Cache metrics	
cpu_info ⁸⁶	CPU Information	
container ⁸⁷	Container metrics	
dfs ⁸⁸	DFSR metrics	
dhcp ⁸⁹	DHCP Server	

82 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.ad.md

83 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.adcs.md

84 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.adfs.md

85 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.cache.md

86 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.cpu_info.md

87 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.container.md

88 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.dfsr.md

89 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.dhcp.md

Наименование	Описание	Комментарий
dns ⁹⁰	DNS Server	
exchange ⁹¹	Exchange metrics	
fsrmquota ⁹²	Microsoft File Server Resource Manager (FSRM) Quotas collector	
hyperv ⁹³	Hyper-V hosts	
iis ⁹⁴	IIS sites and applications	
logon ⁹⁵	User logon sessions	
memory ⁹⁶	Memory usage metrics	
mscluster_cluster ⁹⁷	MSCluster cluster metrics	
mscluster_network ⁹⁸	MSCluster network metrics	
mscluster_node ⁹⁹	MSCluster Node metrics	
mscluster_resource ¹⁰⁰	MSCluster Resource metrics	
mscluster_resourcegroup ¹⁰¹	MSCluster ResourceGroup metrics	
msmq ¹⁰²	MSMQ queues	

90 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.dns.md

91 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.exchange.md

92 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.fsrmquota.md

93 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.hyperv.md

94 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.iis.md

95 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.logon.md

96 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.memory.md

97 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.msccluster_cluster.md

98 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.msccluster_network.md

99 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.msccluster_node.md

100 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.msccluster_resource.md

101 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.msccluster_resourcegroup.md

102 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.msmq.md

Наименование	Описание	Комментарий
mssql ¹⁰³	SQL Server Performance Objects ¹⁰⁴ metrics	
netframework_clrexceptions ¹⁰⁵	.NET Framework CLR Exceptions	
netframework_clrinterop ¹⁰⁶	.NET Framework Interop Metrics	
netframework_clrjit ¹⁰⁷	.NET Framework JIT metrics	
netframework_clrloading ¹⁰⁸	.NET Framework CLR Loading metrics	
netframework_clrlocksandthreads ¹⁰⁹	.NET Framework locks and metrics threads	
netframework_clrmemory ¹¹⁰	.NET Framework Memory metrics	
netframework_clrremoting ¹¹¹	.NET Framework Remoting metrics	
netframework_clrsecurity ¹¹²	.NET Framework Security Check metrics	
process ¹¹³	Per-process metrics	
remote_fx ¹¹⁴	RemoteFX protocol (RDP) metrics	
scheduled_task ¹¹⁵	Scheduled Tasks metrics	
smb ¹¹⁶	SMB Server	

103 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.mssql.md

104 <https://docs.microsoft.com/en-us/sql/relational-databases/performance-monitor/use-sql-server-objects#SQLServerPOs>

105 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.netframework_clrexceptions.md

106 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.netframework_clrinterop.md

107 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.netframework_clrjit.md

108 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.netframework_clrloading.md

109 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.netframework_clrlocksandthreads.md

110 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.netframework_clrmemory.md

111 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.netframework_clrremoting.md

112 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.netframework_clrsecurity.md

113 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.process.md

114 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.remote_fx.md

115 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.scheduled_task.md

116 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.smb.md

Наименование	Описание	Комментарий
smtp ¹¹⁷	IIS SMTP Server	
tcp ¹¹⁸	TCP connections	
teradici_pcoip ¹¹⁹	Teradici PCoIP ¹²⁰ session metrics	
time ¹²¹	Windows Time Service	
thermalzone ¹²²	Thermal information	
terminal_services ¹²³	Terminal services (RDS)	
vmware_blast ¹²⁴	VMware Blast session metrics	
vmware ¹²⁵	Performance counters installed by the Vmware Guest agent	

Для отображения метрик, предоставляемых windows-exporter, взят [дашборд](#)¹²⁶. Изменен ID дашборда, удалены лейблы job, instance, добавлен лейбл group для фильтрации хостов на уровне дашборда, удалены часть графиков.

Подробная информация по используемым метрикам windows-exporter приведена в [разделе 6.18, приложение 1](#) (see page 204).

117 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.smtp.md

118 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.tcp.md

119 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.teradici_pcoip.md

120 https://www.teradici.com/web-help/pcoip_wmi_specs/

121 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.time.md

122 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.thermalzone.md

123 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.terminal_services.md

124 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.vmware_blast.md

125 https://github.com/prometheus-community/windows_exporter/blob/master/docs/collector.vmware.md

126 <https://grafana.com/grafana/dashboards/13466-windows-exporter-for-prometheus-dashboard/>

6.3 6.3 Мониторинг PostgreSQL

i Данные с экспортеров могут использоваться для различных продуктов. Обратите внимание на использование тегов `product` и `component` в документации к конкретному продукту.

Для создания метрик PostgreSQL используются следующие экспортеры:

- [postgres-exporter](#)¹²⁷; для визуализации метрик используется [дашборд](#)¹²⁸,
- [sql-exporter](#)¹²⁹.

Метрики, предоставляемые postgres-exporter, не описываются в данном документе.

6.4 6.4 Мониторинг FreeIPA

i Данные с freeipa-exporter необходимо тегировать тегом `product: freeipa` (если Freeipa мониторится как отдельный продукт). В данном случае в мониторинг будут добавлены только хосты с ролью контроллер домена.
Если экспортер используется в продукте АД Про, то его необходимо тегировать `product: ald-pro, component: freeipa`

Подробная информация по используемым метрикам для FreeIPA приведена в [разделе 6.18, приложение 2 \(see page 208\)](#).

¹²⁷ https://github.com/prometheus-community/postgres_exporter/tree/master

¹²⁸ <https://grafana.com/grafana/dashboards/9628-postgresql-database/>

¹²⁹ https://github.com/burningalchemist/sql_exporter

6.5 6.5 Мониторинг ALD Pro

6.5.1 Используемые метрики



Приложение ALD Pro состоит из нескольких подсистем, для мониторинга которых созданы отдельные дашборды (подразумевается, что подсистемы устанавливаются на разные хосты). Для корректной фильтрации хостов в grafana необходимо создавать объекты мониторинга в административной консоли с дополнительными тегами product и component. Необходимо каждый объект, который отвечает за мониторинг подсистемы в ALD Pro, отметить тегами product: ald-pro и component из таблицы ниже. Для мониторинга необходимы экспортеры: systemd, freeipa, node.

Подсистема	Название дашборда	Дополнительные лейблы
Служба каталогов	ALD-Pro directory service	component: controller
Динамическая настройка узлов	ALD-Pro dynamic node configuration	component: dhcp
Общий доступ к файлам	ALD-Pro file sharing	component: filesharing
Установка ОС по сети	ALD-Pro installing the OS over the network	component: osinstalling
Портал управления	ALD-Pro management portal	component: controller
Мониторинг	ALD-Pro monitoring	component: monitoring
Разрешение имен	ALD-Pro name resolution	component: controller
Печать	ALD-Pro printing system	component: printing
Репозитории программного обеспечения	ALD-Pro software repositories	component: repositoring
Синхронизация времени	ALD-Pro time synchronization	component: controller

Подсистема	Название дашборда	Дополнительные лейблы
Компонент Freeipa	ALD-Pro state with history data	component: freeipa

Подробное описание метрик, используемых для мониторинга ALD Pro, приведено в [разделе 6.18](#), [приложение 3](#) (see page 219).

6.5.2 Конфигурирование Vector для сбора логов по продукту

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
	Лог файлы			
1	/var/log/ipa/*.log	.timestamp .labels.component = "FreeIPA" .severity = "ERROR" .labels.product = "ALDPro" .labels.file = .file	logs_aldpro.yaml	
2	/var/log/ipa/*backup.log	.timestamp .labels.component = "FreeIPA" .severity = "ERROR" .labels.product = "ALDPro" .labels.file = .file	logs_aldpro.yaml	
3	/var/log/dirsrv/*/audit	.timestamp .labels.component = "FreeIPA" .severity = "ERROR" .labels.product = "ALDPro" .labels.file = .file	logs_aldpro.yaml	

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
4	/var/log/dirsrv/*/access	.labels.ldap_operation .labels.valid_cred .timestamp .labels.component = "FreelPA" .severity = "ERROR" .labels.product = "ALDPro" .labels.file = .file	logs_aldpro.yaml	
5	/var/log/dirsrv/*/errors	.severity .timestamp .labels.component = "FreelPA" .severity = "ERROR" .labels.product = "ALDPro" .labels.file = .file	logs_aldpro.yaml	
6	/var/log/auth.log	.labels.username .timestamp .labels.component = "FreelPA" .severity = "ERROR" .labels.product = "ALDPro" .labels.file = .file	logs_aldpro.yaml	
7	/var/log/sss/*.*.log	.labels.component = "SSSD" .timestamp .severity = "ERROR" .labels.product = "ALDPro" .labels.file = .file	logs_aldpro.yaml	
8	/var/log/apache2/error.log	.timestamp .labels.component = "Apache" .labels.product = "ALDPro"	logs_apache.yaml	
9	"/var/log/apache2/ access.log", "/var/log/apache2/ other_vhosts_access.log"	.timestamp .labels.component = "Apache" .labels.product = "ALDPro"	logs_apache.yaml	

6.6 6.6 Мониторинг ПК СВ Брест

6.6.1 Используемые метрики

i Приложение Брест состоит из нескольких составных частей (хосты виртуализации, фронт сервера, сервер для доступа к API Брест), для мониторинга которых созданы отдельные дашборды. Для корректной фильтрации хостов в grafana необходимо создавать объекты мониторинга в административной консоли с дополнительными тегами `product` и `component`. Необходимо каждый объект, который отвечает за мониторинг Брест, отметить тегом `product: brest`. На данный момент тегом `component: virthost` необходимо протегировать хосты виртуализации. Если есть необходимость замониторить базу данных PostgreSQL, то объект необходимо отметить тегом `component: BD`. Остальные части Брест данным тегом не помечаются.

Для мониторинга необходимы экспортеры: `systemd-exporter`, `one-exporter`, `node-exporter`, `libvirt-exporter`.

Подробная информация по метрикам, используемым для мониторинга ПК СВ Брест, приведена в [разделе 6.18, приложение 4 \(see page 242\)](#).

6.6.2 Конфигурирование Vector для сбора логов по продукту

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
	Лог файлы			
1	/var/log/apache2/error.log	.timestamp .labels.component = "Apache" .labels.product = "Brest"	logs_apache.yaml	
2	/var/log/apache2/access.log	.timestamp .labels.component = "Apache" .labels.product = "Brest"	logs_apache.yaml	

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
3	/var/log/apache2/ other_vhosts_access.log	timestamp .labels.component = "Apache" .labels.product = "Brest"	logs_apache.yaml	
4	/var/log/libvirt/qemu/*.log	.severity = "ERROR" .timestamp .labels.component = "Libvirt" .labels.product = "Brest"	logs_libvirt.yaml	
5	/var/log/one/oned.log	.timestamp .severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .labels.uid .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	
6	/var/log/one/monitor.log	.timestamp .severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .labels.uid .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	
7	/var/log/one/oneflow.log	.timestamp .severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .labels.uid .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
8	/var/log/one/onegate.log	.timestamp .severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .labels.uid .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	
9	/var/log/one/onehem.log	.timestamp .severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .labels.uid .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	
10	/var/log/one/sched.log	.timestamp .severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .labels.uid .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	
11	/var/log/one/sunstone.log	.timestamp .severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .labels.uid .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
12	/var/log/one/novnc.log	.severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .timestamp .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	
13	/var/log/one/host_enable.log	.severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .timestamp .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	
14	/var/log/one/host_error.log	.severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .timestamp .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	
15	/var/log/one/host_maintenance.log	.severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .timestamp .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
16	/var/log/one/m-autom.log	.severity = "ERROR" .severity = "INFO" .severity = "DEBUG" .timestamp .labels.component = "One" .labels.product = "Brest" .labels.file = .file	logs_one.yaml	
17	/var/lib/postgresql/11/main/pg_log/*.log	.timestamp .labels.file = .file .labels.component = "PostgreSQL" .labels.product = "Brest"	logs_postgresql.yaml	
18	/var/log/postgresql/postgresql-11-main.log	.timestamp .labels.file = .file .labels.component = "PostgreSQL" .labels.product = "Brest"	logs_postgresql.yaml	
19	"/var/log/sss/*.log"	.timestamp .labels.file = .file .severity = "ERROR" .labels.component = "SSSD" .labels.product = "Brest"	logs_sssd.yaml	
	Логи сервисов для leader (journald)			

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
1	"opennebula-flow.service", "opennebula-gate.service", "opennebula-hem.service", "opennebula-showback.timer"	.labels.component = "One" .timestamp .labels.product = "Brest" .labels.systemdunit = _SYSTEMD_UNIT .severity = "DEBUG" .severity = "INFO" .severity = "NOTICE" .severity = "WARN" .severity = "ERROR" .severity = "CRIT" .severity = "EMERG"	journald_brest.yaml journald_leader.yaml	
Логи сервисов для follower (journald)				
1	"opennebula.service", "opennebula-sunstone.service", "opennebula-pdp-datastores-incron.service", "opennebula-ssh-socks-cleaner.timer", "opennebula-apache-auth.service", "opennebula-cli-session.service", "opennebula-fireedge.service", "opennebula-guacd.service", "opennebula-novnc.service", "opennebula-onedisk-incron.service", "opennebula-scheduler.service", "opennebula-ssh-agent.service"	.labels.component = "One" .timestamp .labels.product = "Brest" .labels.systemdunit = _SYSTEMD_UNIT .severity = "DEBUG" .severity = "INFO" .severity = "NOTICE" .severity = "WARN" .severity = "ERROR" .severity = "CRIT" .severity = "EMERG"	journald_brest.yaml journald_follower.yaml 	

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
2	"chrony.service"	.labels.component = "Chrony" .timestamp .labels.product = "Brest" .labels.systemdunit = ._SYSTEMD_UNIT .severity = "DEBUG" .severity = "INFO" .severity = "NOTICE" .severity = "WARN" .severity = "ERROR" .severity = "CRIT" .severity = "EMERG"	journald_brest.yaml journald_follower.yaml 	
3	"sssd.service"	.labels.component = "SSSD" .timestamp .labels.product = "Brest" .labels.systemdunit = ._SYSTEMD_UNIT .severity = "DEBUG" .severity = "INFO" .severity = "NOTICE" .severity = "WARN" .severity = "ERROR" .severity = "CRIT" .severity = "EMERG"	journald_brest.yaml journald_follower.yaml 	

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
4	"postgresql@11-main.service", "postgresql.service"	.labels.component = "PostgreSQL" .timestamp .labels.product = "Brest" .labels.systemdunit = _SYSTEMD_UNIT .severity = "DEBUG" .severity = "INFO" .severity = "NOTICE" .severity = "WARN" .severity = "ERROR" .severity = "CRIT" .severity = "EMERG"	journald_brest.yaml journald_follower.yaml	

6.7 6.7 Мониторинг RuPost

6.7.1 Используемые метрики

Подробное описание метрик, используемых для мониторинга RuPost, приведено в [разделе 6.18, приложение 5 \(see page 278\)](#).

6.7.2 Конфигурирование Vector для сбора логов по продукту

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
	Лог файлы			
1	/var/log/sogo/*.log	.timestamp .labels.component = "SOGO" .severity .labels.product = "RuPost"	logs_postfix.yaml	

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
2	/var/log/mail.log	.timestamp .labels.component = "MAIL" .severity .labels.product = "RuPost"	logs_postfix.yaml	
3	/var/log/rupost/monitor.log	.timestamp .labels.component = "MONITOR" .severity .labels.product = "RuPost"	logs_postfix.yaml	

6.8 6.8 Мониторинг RuBackup

6.8.1 Используемые метрики



На данный момент продукт RuBackup мониторится с помощью опроса API. Для корректной визуализации метрик необходимо пометить тегом product: rubackup все объекты приложения (основной сервер, базу данных). Если необходимо замониторить базу данных, то ее дополнительно надо отметить тегом component: BD.
Для мониторинга необходимы экспортеры: rubackup-exporter, postgres-exporter.

Подробная информация по используемым метрикам для RuBackup приведена в [разделе 6.18, приложение 6](#) (see page 294).

6.8.2 Конфигурирование Vector для сбора логов по продукту

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
	Лог файлы			

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
1	/opt/rubackup/log/task*	.timestamp .labels.product = "RuBackup" .labels.component = "Task"	logs_rubackup.yaml	

6.9 6.9 Мониторинг Termidesk

6.9.1 Используемые метрики



На данный момент продукт Termidesk мониторится с помощью опроса API. Для корректной визуализации метрик необходимо пометить тегом `product: termidesk` все объекты приложения (основной сервер).
Для мониторинга необходимы экспортеры: `termidesk-exporter`.

Подробная информация по используемым метрикам для Termidesk приведена в [разделе 6.18, приложение 7 \(see page 299\)](#).

6.9.2 Конфигурирование Vector для сбора логов по продукту

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
	Лог файлы			

№	Имя лога	Лейблы	Файл вектора (/etc/vector/conf.d/)	Комментарий
1	/var/log/termidesk/ auth.log /var/log/termidesk/ celery-beat.log /var/log/termidesk/ celery-worker.log /var/log/termidesk/ other.log /var/log/termidesk/ services.log /var/log/termidesk/ termidesk.log /var/log/termidesk/ use.log /var/log/termidesk/ wsproxy.log /var/log/termidesk/ sql.log /var/log/termidesk/ workers.log	.timestamp .labels.product = "Termidesk" .labels.component = "VDI"	logs_termidesk.yaml	
2	/var/log/rabbitmq/ rabbit@*.log	.timestamp .labels.product = "Termidesk" .labels.component = "Rabbit"	logs_middleaware.yaml	
3	/var/log/apache2/ error.log /var/log/apache2/ access.log	.timestamp .labels.product = "Termidesk" .labels.component = "Apache"	logs_middleaware.yaml	
4	/var/lib/postgresql/11/ main/pg_log/ postgresql*.log	.timestamp .labels.product = "Termidesk" .labels.component = "PostgreSQL"	logs_middleaware.yaml	

6.10 6.10 Мониторинг Billmanager

6.10.1 Используемые метрики



На данный момент продукт Billmanager мониторится с помощью опроса API. Для корректной визуализации метрик необходимо пометить тегом `product: billm` все объекты приложения (основной сервер).
Для мониторинга необходимы экспортеры: `billm-exporter`.

Подробная информация по используемым метрикам для Billmanager приведена в [разделе 6.18](#), [приложение 8](#) (see page 331).

6.11 6.10 Триггеры для создания событий по логам

Ниже описаны триггеры для создания событий по логам, которые в данный момент настроены в Astra Monitoring.

Продукт	Условие (триггер)	Описание	Лог	Технические детали
ALD Pro	Вызывается критическое уведомление на сообщение в логе 'Not listening for new connections - too many fds open'. Критическое уведомление снимается, если в логе появляется сообщение 'Listening for new connections again'.	Лог сигнализирует о проблеме с количеством открытых файловых дескрипторов (fds) в системе. Эта ошибка обычно говорит о том, что был достигнут лимит операционной системы на количество файловых дескрипторов, которые может открыть процесс. В результате новые соединения не будут прослушиваться, что может вызвать сбой в доступности сервиса.	/var/log/dirsrv/ slapd- <DOMAIN>/ error	Имя алерта - 'Not listening for new connections' Severity = 'critical' Для срабатывания триггера используется регулярное выражение: r'^[(?P<timestamp>\d{2}/[a-zA-Z]{3}/\d{4}:\d{2}:\d{2}:\d{2}\.\d{9} \+\d{4})] - ERR - .*? - Not listening for new connections - too many fds open' Для гашения: r'^[(?P<timestamp>\d{2}/[a-zA-Z]{3}/\d{4}:\d{2}:\d{2}:\d{2}\.\d{9} \+\d{4})] - ERR - .*? - Listening for new connections again'

6.11.1 6.10.1 Сводный список собираемых логов для АИС

1. Серверы общего назначения (Astra Linux)
 - /var/log/audit/audit.log
 - /var/log/auth.log
 - /var/log/sss/*.log
 - /var/log/syslog
 - /parsec/log/astra/events
 - /var/log/astra-monitoring/*.log (исключая vector.log)
 - /var/log/apache2/error.log
 - /var/log/apache2/access.log
 - /var/log/apache2/other_vhosts_access.log
2. Узлы виртуализации и управления
 - /var/log/audit/audit.log

- /var/log/auth.log
- /var/log/sss/*.log
- /var/log/syslog
- /parsec/log/astra/events
- /var/log/astra-monitoring/*.log
- /var/log/libvirt/qemu/*.log
- /var/log/one/oned.log
- /var/log/one/monitor.log
- /var/log/one/oneflow.log
- /var/log/one/onegate.log
- /var/log/one/onehem.log
- /var/log/one/sched.log
- /var/log/one/sunstone.log

3. Узлы ALD Pro

- /var/log/audit/audit.log
- /var/log/auth.log
- /var/log/sss/*.log
- /var/log/syslog
- /parsec/log/astra/events
- /var/log/astra-monitoring/*.log (исключая vector.log)
- /var/log/ipa/*.log
- /var/log/ipa/*backup.log
- /var/log/dirsrv*/audit
- /var/log/dirsrv*/access
- /var/log/dirsrv*/errors
- /var/log/apache2/error.log
- /var/log/apache2/access.log
- /var/log/apache2/other_vhosts_access.log

6.12 6.11 Мониторинг Astra Kube

Объект	Источник	Описание	Технические детали
Логи Kubernetes контейнеров	/var/log/pods	Логи содержат все данные, которые контейнеры отправляют в stdout/stderr	<p>Используется helm чарт Vector.dev, который устанавливается в Kubernetes кластер и читает логи подов (через HostPath) из /var/log/pods. Vector.dev разворачивается как DaemonSet на все узлах кластера (включая мастеров).</p> <p>Логи обогащаются Vector.dev дополнительными данными, далее в labels извлекаются следующие данные:</p> <ul style="list-style-type: none"> • container_id • container_image • container_name • pod_id • pod_name • pod_namespace • pod_node_name • pod_owner • pod_uid • file (имя файла, в котором содержится лог) • лейблы namespace • лейблы узла • лейблы пода <p>Severity</p> <p>Дополнительно собираемые логи парсятся через parse_json (некоторые компоненты Kubernetes отправляют логи в формате json, но, например, пока Cilium так не делает) или parse_klog (формат специальных сообщений от управляющих компонентов Kubernetes - см. System Logs¹³⁰). Такие сообщения содержат поле level, которое преобразуется в severity).</p>

¹³⁰ <https://kubernetes.io/docs/concepts/cluster-administration/system-logs/>

Объект	Источник	Описание	Технические детали
Аудитные логи Kubernetes	/var/log/kubernetes/audit	Логи содержат аудитные данные в соответствии с настройками аудита для Kubernetes кластера	<p>Используется helm чарт Vector.dev, который устанавливается в Kubernetes кластер и читает файлы (через HostPath) из /var/log/kubernetes/audit. Vector.dev разворачивается как DaemonSet на все узлах кластера (включая мастеров). Формат аудитных логов - json.</p> <p>Из логов дополнительно извлекаются в labels следующие данные (см. Audit Events¹³¹):</p> <ul style="list-style-type: none"> • level • stage • verb <p>Severity</p> <p>По умолчанию severity == "INFO". Если в сообщении stage == "Panic", то severity меняется на CRIT.</p>
События Kubernetes кластера		События Kubernetes	<p>Vector.dev не имеет встроенного источника для событий в Kubernetes, поэтому для получения событий может использоваться другие подходы:</p> <ol style="list-style-type: none"> 1. Kubernetes event logger¹³² - под, который подписывается на события Kubernetes и отправляет их в stdout. Vector.dev считывается события как логи конкретного пода (для исключения дублирования событий необходимо этот под исключить из сбора Vector.dev данный под). 2. Компонент, который подписывается на логи Kubernetes, и записывает их в файл. Далее Vector.dev считывает логи из данного файла. <p>Из логов дополнительно извлекаются в labels следующие данные (см. Events¹³³):</p> <ul style="list-style-type: none"> • kind • name • namespace • reason • type <p>Severity</p> <p>Severity выставляется на основе поля type.</p>

131 <https://kubernetes.io/docs/reference/config-api/apiserver-audit.v1/#audit-k8s-io-v1-Event>

132 <https://github.com/max-rocket-internet/k8s-event-logger>

133 <https://kubernetes.io/docs/reference/kubernetes-api/cluster-resources/event-v1/>

6.13 6.12 Мониторинг SNMP устройств (агент мониторинга)



Для корректного отображения устройств на дашборде в grafana следут использовать значение тега component как IP адрес или имя устройства, которое мониторится с помощью snmp-exporter.

6.13.1 SNMP поллинг

Для SNMP мониторинга удаленных устройств применяется [snmp-exporter](#)¹³⁴.

Для генерации конфигурационного файла используется специальная утилита [generator](#)¹³⁵.

В архив с snmp-exporter добавлен каталог стандартных mib файлов, для генерации конфигов.

Эта утилита обрабатывает необходимые MIBs и превращает их в список OID цепочек и названий метрик.

SNMP exporter опрашивает удаленные устройства на основе этих данных и выводит метрики в формате prometheus для дальнейшего анализа другими системами.

6.13.2 Запуск snmp-exporter с помощью агента мониторинга

Для установки экспортера как сервис см. [главу 3](#) руководства.

Для корректного запуска экспортера необходимо выполнить следующие настройки:

В конфигурационном файле агента в разделе exporters указываем имя запускаемого экспортера и параметр is_custom.

```
exporters:
  - name: snmp_exporter
    is_custom: true
    args: "--config.file=/opt/astramon-agent/exporters/config/snmp.yml"
    health_address: 127.0.0.1:9116/metrics
```

- Имя в поле name должно совпадать с названием бинарного файла с применением дополнительных условий именования сторонних экспортеров (пример: astramon-snmp_exporter-custom)
- Параметр is_custom говорит агенту, что это сторонний экспортер, но агент может управлять им (запускать, делать health check, останавливать и сообщать статус работы экспортера в

¹³⁴ https://github.com/prometheus/snmp_exporter

¹³⁵ https://github.com/prometheus/snmp_exporter/tree/main/generator

Config API). Так же это дает понять агенту не использовать другую функциональность, что доступна экспортерам, специально разработанным для AM.

- Параметр `args` сообщает экспортеру параметры запуска, в них указывается путь к файлу конфигурации экспортера.

Следом необходимо выполнить настройку в разделе `metrics`.

```
custom_targets:
  - name: snmp_exporter1
    component: 10.177.248.228
    target: 127.0.0.1:9116
    metrics_path: /snmp?module=apcups&target=10.177.248.228
  - name: snmp_exporter2
    component: 10.177.248.234
    target: 127.0.0.1:9116
    metrics_path: /snmp?module=linux&target=10.177.248.234
```

- Имя в поле `name` должно быть уникальным (особенности работы `vmagent`). В примере мы будем собирать данные с двух таргетов.
- Параметр `component` необходим для добавления лейбла компонент, чтобы можно было различать метрики для разных удаленных устройств.
- `target` описывает адрес и порт, с которого будут считываться метрики.
- `metrics_path` позволяет обратиться к конкретному эндпоинту, для получения метрик. По умолчанию `/metrics`, но в нашем случае этот эндпоинт выводит метрики для разных удаленных устройств. Если необходимо можно создать еще одну запись и собирать непосредственно метрики из `/metrics`.

6.14 6.13 Мониторинг устройств с помощью IPMI (агент мониторинга)



Для корректного отображения устройств на дашборде в `grafana` следует использовать значение тега `component` как IP адрес или имя устройства, которое мониторится с помощью `ipmi-exporter`.

6.14.1 IPMI мониторинг

Для мониторинга удаленных устройств с помощью протокола IPMI применяется [ipmi-exporter](#)¹³⁶.

¹³⁶ https://github.com/prometheus-community/ipmi_exporter

IPMI exporter опрашивает удаленные устройства и выводит метрики в формате prometheus для дальнейшего анализа другими системами.

Ключи запуска экспортера:

КЛЮЧ	ОПИСАНИЕ
<code>--config.file=CONFIG.FILE</code>	Path to configuration file.
<code>--freeipmi.path=FREEIPMI.PATH</code>	Path to FreeIPMI executables (default: rely on \$PATH).
<code>--web.listen-address=:9290</code>	Addresses on which to expose metrics and web interface. Repeatable for multiple addresses.
<code>--web.config.file=""</code>	Path to configuration file that can enable TLS or authentication.
<code>--log.level=info</code>	Only log messages with the given severity or above. One of: [debug, info, warn, error]
<code>--log.format=logfmt</code>	Output format of log messages. One of: [logfmt, json]

Для работы экспортера необходим установленный компонент [FreeIPMI](#)¹³⁷, в составе которого следующие приложения:

- ipmimonitoring/ipmi-sensors
- ipmi-dcmi
- ipmi-raw
- bmc-info
- ipmi-sel
- ipmi-chassis

[Документация по запуску экспортера](#)¹³⁸.

[Документация по метрикам для разных коллекторов](#)¹³⁹.

6.14.2 Запуск ipmi-exporter с помощью агента мониторинга

Для установки экспортера как сервис см. [главу 3](#) руководства.

Для корректного запуска экспортера необходимо выполнить следующие настройки:

1. В конфигурационном файле агента мониторинга в разделе exporters указываем имя запускаемого экспортера и параметр is_custom:

¹³⁷ <https://www.gnu.org/software/freeipmi/>

¹³⁸ https://github.com/prometheus-community/ipmi_exporter/blob/master/docs/configuration.md

¹³⁹ https://github.com/prometheus-community/ipmi_exporter/blob/master/docs/metrics.md

```
exporters:
  - name: ipmi_exporter
    is_custom: true
    args: "--config.file=/opt/astramon-agent/exporters/config/ipmi-conf.yml"
    health_address: 127.0.0.1:9290/metrics
```

- Имя в поле `name` должно совпадать с названием бинарного файла с применением дополнительных условий именования сторонних экспортеров (пример: `astramon-ipmi_exporter-custom`).
- Параметр `is_custom` говорит агенту, что это сторонний экспортер, но агент может управлять им (запускать, делать `health check`, останавливать и сообщать статус работы экспортера в `Config API`). Так же это дает понять агенту не использовать другую функциональность, что доступна экспортерам, специально разработанным для АМ.
- Параметр `args` сообщает экспортеру параметры запуска, в них указывается путь к файлу конфигурации экспортера.

2. Следом необходимо выполнить настройку в разделе `metrics`:

```
custom_targets:
  - name: ipmi_exporter1
    component: 10.177.248.234
    target: 127.0.0.1:9290
    metrics_path: /ipmi?target=10.177.248.234

  - name: ipmi_exporter2
    component: 10.177.248.228
    target: 127.0.0.1:9290
    metrics_path: /ipmi?target=10.177.248.228
```

- Имя в поле `name` должно быть уникальным (особенности работы `vmagent`). В примере мы будем собирать данные с двух таргетов.
- Параметр `component` необходим для добавления лейбла компонент, чтобы можно было различать метрики по экспортерам на одном хосту.
- `target` описывает адрес и порт, с которого будут считываться метрики.
- `metrics_path` позволяет обратиться к конкретному эндпоинту, для получения метрик. По умолчанию `/metrics`, но в нашем случае этот эндпоинт выводит метрики по запуску скриптов, а не метрики самих скриптов. Если необходимо можно создать еще одну запись и собирать непосредственно метрики из `/metrics`.

6.15 6.14 Базовые правила для создания событий

Базовые правила оповещений созданы на основе метрик node-exporter. Соответственно, этот экспортер должен быть установлен, автоматически запускаться и корректно работать.

Базовые правила объединены в группу правил General, файл helm/alert-rules/general.yaml или docker-compose/vmalert/config/general.yaml.

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
Node_Down	Node <hostname> is possibly down	up{instance=~".*:9100"} == 0	5m	critical	Node-exporter on <hostname> (<instance>) does not respond, so the host is possibly down
Node_Restart	Node <hostname> has been restarted	node_time_seconds - node_boot_time_seconds < 600	-	critical	Node <hostname> (<instance>) has been restarted (uptime < 10m)
CPU_Utilization	High CPU utilization on <hostname> (<value>% idle)	avg by (hostname, instance, job, group) (irate(node_cpu_seconds_total{mode="idle"}[1m]) * 100) < 20	5m	warning	<hostname> (<instance>) has high CPU utilization for more than 5 minutes
CPU_Utilization	Critical CPU utilization on <hostname> (<value>% idle)	avg by (hostname, instance, job, group) (irate(node_cpu_seconds_total{mode="idle"}[1m]) * 100) < 10	5m	critical	<hostname> (<instance>) has critical CPU utilization for more than 5 minutes

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
CPU_HighIOWait	High CPU iowait on<hostname> (<value>%)	(avg by (instance, hostname, job, group) (rate(node_cpu_seconds_total{mode="iowait"}[1m])) * 100 > 10) * on(instance, hostname, job, group) group_left (nodename) node_uname_info{nodename=~".+"}	5m	warning	CPU iowait > 10%. A high iowait means that you are disk or network bound
Memory_Utilization	High Memory utilization on <hostname> (<value>% available)	node_memory_MemAvailable_bytes / node_memory_Total_bytes * 100 < 20	5m	warning	<hostname> (<instance>) has high Memory utilization for more than 5 minutes
Memory_Utilization	Critical Memory utilization on <hostname> (<value>% available)	node_memory_MemAvailable_bytes / node_memory_Total_bytes * 100 < 10	5m	critical	<hostname> (<instance>) has critical Memory utilization for more than 5 minutes
DiskSpace_Utilization	Host out of disk space (instance <instance>)	((node_filesystem_available_bytes * 100) / node_filesystem_size_bytes < 20 and ON (instance, device, mountpoint) node_filesystem_readonly == 0) * on(instance, hostname, job, group) group_left (nodename) node_uname_info{nodename=~".+"}	5m	warning	Disk is almost full (< 20% left)

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
DiskSpace_Utilization	Host out of disk space (instance <instance>)	((node_filesystem_ava il_bytes * 100) / node_filesystem_size_ bytes < 10 and ON (instance, device, mountpoint) node_filesystem_read only == 0) * on(instance, hostname, job, group) group_left (nodename) node_uname_info{nod ename=~".+"}	5m	critical	Disk is almost full (< 10% left)
HostOutOfInodes	Host out of inodes (instance <instance>)	(node_filesystem_files _free{fstype! ="msdosfs"} / node_filesystem_files{ fstype!="msdosfs"}) * 100 < 10 and ON (instance, device, mountpoint) node_filesystem_read only == 0) * on(instance, hostname, job, group) group_left (nodename) node_uname_info{nod ename=~".+"}	5m	warning	Disk is almost running out of available inodes (< 10% left)
HostFilesystemDeviceError	Host <hostname> filesystem <mountpoint> device error	node_filesystem_devic e_error == 1	-	critical	<hostname> (<instance>): Device error with the <mountpoint> filesystem

6.15.1 6.14.1 Сводные правила оповещений для АИС

Общие правила (general.yaml)

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
Node_Down	Node <hostname> is possibly down	<code>up{instance=~".*:9100"} == 0</code>	5m	critical	Node-exporter on <hostname> (<instance>) does not respond, so the host is possibly down
Node_Reboot	Node <hostname> has been restarted	<code>node_time_seconds - node_boot_time_seconds < 600</code>	-	critical	Node <hostname> (<instance>) has been restarted (uptime < 10m)
CPU_Utilization	High CPU utilization on <hostname> (<value>% used)	<code>100 - avg by (hostname, instance, job, group) (irate(node_cpu_seconds_total{mode="idle"}[1m]) * 100) > 60</code>	5m	warning	<hostname> (<instance>) has high CPU utilization for more than 5 minutes
CPU_Utilization	Critical CPU utilization on <hostname> (<value>% used)	<code>100 - avg by (hostname, instance, job, group) (irate(node_cpu_seconds_total{mode="idle"}[1m]) * 100) > 85</code>	5m	critical	<hostname> (<instance>) has critical CPU utilization for more than 5 minutes

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
CPU_System_Utilization	Critical System CPU utilization on <hostname> (<value>% used)	avg by (hostname, instance, job, group) (irate(node_cpu_seconds_total{mode="system"}[1m]) * 100) > 50	5m	critical	<hostname> (<instance>) has more than 50% CPU utilization at the system (kernel) level for more than 5 minutes.
CPU_HighIOwait	High CPU iowait on <hostname> (<value>%)	(avg by (instance, hostname, job, group) (rate(node_cpu_seconds_total{mode="iowait"}[1m])) * 100 > 10) * on(instance, hostname, job, group) group_left (nodename) node_uname_info{nodename=~".+"}	5m	warning	CPU iowait > 10%. A high iowait means that you are disk or network bound
CPU_CriticalIOwait	Critical CPU iowait on <hostname> (<value>%)	(avg by (instance, hostname, job, group) (rate(node_cpu_seconds_total{mode="iowait"}[1m])) * 100 > 30) * on(instance, hostname, job, group) group_left (nodename) node_uname_info{nodename=~".+"}	5m	critical	CPU iowait > 30%. A high iowait means that you are disk or network bound.
Memory_Utilization	High Memory utilization on <hostname> (<value>% used)	100 - (node_memory_MemAvailable_bytes / node_memory_MemTotal_bytes * 100) > 80	5m	warning	<hostname> (<instance>) has high Memory utilization for more than 5 minutes

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
Memory_Utilization	Critical Memory utilization on <hostname> (<value>% used)	$100 - (\text{node_memory_MemAvailable_bytes} / \text{node_memory_MemTotal_bytes} * 100) > 95$	5m	critical	<hostname> (<instance>) has critical Memory utilization for more than 5 minutes
DiskSpace_Utilization	Host out of disk space (instance <instance>)	$((\text{node_filesystem_avail_bytes} * 100) / \text{node_filesystem_size_bytes} < 20 \text{ and ON (instance, device, mountpoint) node_filesystem_readonly} == 0) * \text{on(instance, hostname, job, group) group_left (nodename) node_uname_info\{nodename=~\".+\}}$	5m	warning	Disk is almost full (< 20% left)
DiskSpace_Utilization	Host out of disk space (instance <instance>)	$((\text{node_filesystem_avail_bytes} * 100) / \text{node_filesystem_size_bytes} < 10 \text{ and ON (instance, device, mountpoint) node_filesystem_readonly} == 0) * \text{on(instance, hostname, job, group) group_left (nodename) node_uname_info\{nodename=~\".+\}}$	5m	critical	Disk is almost full (< 10% left)
HostOutOfInodes	Host out of inodes (instance <instance>)	$(\text{node_filesystem_files_free}\{\text{fstype!} = \text{"msdosfs"}\} / \text{node_filesystem_files}\{\text{fstype!} = \text{"msdosfs"}\} * 100 < 10 \text{ and ON (instance, device, mountpoint) node_filesystem_readonly} == 0) * \text{on(instance, hostname, job, group) group_left (nodename) node_uname_info\{nodename=~\".+\}}$	5m	warning	Disk is almost running out of available inodes (< 10% left)

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
HostFilesystemDeviceError	Host <hostname> filesystem <mountpoint> device error	node_filesystem_device_error == 1	-	critical	<hostname> (<instance>): Device error with the <mountpoint> filesystem
NetworkInterfaceDown	Network interface <device> on <hostname> is in 'Down' state	node_network_info{operstate="down", device! ~"eno[0-9]+"} == 1	-	critical	<hostname> (<instance>): network interface <device> is in 'Down' state

Правила для Brest (brest.yaml)

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
Brest_vCPU_usage	High vCPU usage on Brest cluster <cluster>: >80%	$(\text{one_cluster_cpuusage} / \text{one_cluster_totalcpu}) * 100 > 80$	5m	critical	Node-exporter on <hostname> (<instance>) does not respond, so the host is possibly down
Brest_virtualization_service_status	Service <service> on Brest virtualization server <hostname> is possibly down	<code>systemd_unit_state_id{name=~"libvirtd.* postgresql@.* chrony.* sssd.* opennebula.*", product="brest"} != 1</code>	-	critical	On Brest virtualization server <hostname> (<instance>) service <service> is not running.
Brest_front_service_status	Service <service> on Brest front server <hostname> is possibly down	<code>systemd_unit_state_id{name=~"libvirtd.* postgresql@.* chrony.* sssd.* opennebula.*", product="brest"} != 1</code>	-	critical	On Brest front server <hostname> (<instance>) service <service> is not running.
Brest_RAFT_status	Brest server <hostname> has an issue with RAFT status	<code>one_zone_raft{} == 10</code>	1m	critical	Brest server <hostname> (<instance>) has an issue with RAFT status.

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
Brest_API_status	Brest server <hostname> has an issue with API connection	one_api_connect{} != 1	1m	critical	Brest server <hostname> (<instance>) has an issue with API connection.
Brest_web_portal_status	Brest has an issue with web portal connection	one_web_connect{} != 200	2m	warning	Brest has an issue with web portal <hostname> connection (using <instance> exporter)
Brest_web_portal_duration	On Brest server <hostname> duration of time for web portal connection is too long	one_web_connect_duration{} >= 2000	5m	warning	On Brest server <hostname> (<instance>) duration of time for web portal connection is more than 2 seconds.
Brest_front_host_status	Brest front server <hostname> is possibly down	node_exporter_build_info{product="brest", component="front"} != 1	5m	critical	Brest front server <hostname> (<instance>) is not responding. It may be down.

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
Brest_virtualization_host_error	Brest virtualization server <hostname> is in ERROR state	one_host_state == 3	2m	critical	Brest virtualization server <hostname> (<instance>) is in ERROR state.
Brest_virtualization_host_init	Brest virtualization server <hostname> is in INIT state	one_host_state == 1	2m	critical	Brest virtualization server <hostname> (<instance>) is in INIT state.
Brest_virtualization_host_disabled	Brest virtualization server <hostname> is in DISABLED state	one_host_state == 4	2m	critical	Brest virtualization server <hostname> (<instance>) is in DISABLED state.
Brest_virtualization_host_offline	Brest virtualization server <hostname> is in OFFLINE state	one_host_state == 8	2m	critical	Brest virtualization server <hostname> (<instance>) is in OFFLINE state.

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
Brest_virtualization_host_monitored	Brest virtualization server <hostname> is in MONITORED state	one_host_state == 2	2m	info	Brest virtualization server <hostname> (<instance>) is in MONITORED state.
Brest_changes_RAFT_status	On Brest server <hostname> status RAFT has changed	sum by() (changes(one_zone_raft){[5m]}) > 0	-	warning	On Brest server <hostname> (<instance>) status RAFT has changed in last 5 minutes.
Brest_new_running_VMs	A lot of new VMs on <hostname> in last 10 minutes	delta(sum(one_vms_states_count{}))[10m] > 50	-	warning	More than 50 new VMs were created on <hostname> in last 10 minutes
Brest_new_running_VMs	More than 500 new VMs on <hostname> in last 10 minutes	delta(sum(one_vms_states_count{}))[10m] > 500	-	warning	More than 500 new VMs were created on <hostname> in last 10 minutes

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
Brest_RAFT_issues	More than 50% of Brest fronts are in error state	$(\text{count}(\text{one_zone_raft}) == 10) \text{ or } \text{vector}(0) / \text{count}(\text{one_zone_raft}) * 100 > 50$	5m	critical	More than 50% of Brest front servers are in 'Issue' RAFT status

Правила IPMI (ipmi_exporter.yaml)

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
IPMI_temperature_celsius	Temperature sensor '<name>' on server <hostname> has high temperature	$\text{ipmi_temperature_celsius} \geq 75$	5m	warning	Temperature sensor '<name>' on server <hostname> has temperature <value> degrees Celsius.

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
IPMI_temperature_celsius	Temperature sensor '<name>' on server <hostname> has critical temperature	ipmi_temperature_celsius >= 90	5m	critical	Temperature sensor '<name>' on server <hostname> has critical temperature <value> degrees Celsius.
IPMI_chassis_power_state	Chassis of server <hostname> is powered off	ipmi_chassis_power_state != 1	-	critical	Chassis power on server <hostname> is switched off or failed
IPMI_collector_status	IPMI collector on server <hostname> possibly is not working	ipmi_up != 1	2m	critical	Status of IPMI collector on server <hostname> is down.
IPMI_current_state	Problem with <name> current state on server <hostname>	ipmi_current_state != 0	-	critical	Status of <name> current on server <hostname> is not OK

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
IPMI_voltage_state	Problem with <name> voltage sensor on server <hostname>	ipmi_voltage_state != 0	-	critical	Status of voltage sensor <name> on server <hostname> is not OK
IPMI_voltage_volts	Voltage failure <name> on server <hostname>	ipmi_voltage_volts == 0	-	critical	<name> voltage input or output on server <hostname> is 0

Правила SNMP (snmp_exporter.yaml)

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
Device_Down	Network device <hostname> <instance> is possibly down	up{instance=~".*:9116"} == 0	2m	critical	Network device <hostname> <instance> does not respond by SNMP, so the host is possibly down
IfOperStatus	Interface <interface> on <instance> is DOWN	(ifOperStatus * on(ifIndex, job) group_left (ifName) ifName != 1) and (ifAdminStatus * on(ifIndex, job) group_left (ifName) ifName == 1)	-	warning	Interface <interface> on <instance> is in DOWN state while AdminState configured as UP
IfErrors	Too many errors on interface <interface> of <instance>	(rate(ifOutErrors[1m]) * on(ifIndex, job) group_left (ifName) ifName > 0) or (rate(ifInErrors[1m]) * on(ifIndex, job) group_left (ifName) ifName > 0)	2m	warning	Too many errors on interface <interface> of <instance> for last 2 minutes
Device_Restart	Network device <instance> was restarted	sysUpTime / 100 < 600	-	critical	Network device <instance> was restarted in last 10 minutes

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
If_Admin_Status	ifAdminStatus on host <hostname> is down	ifAdminStatus{product="SNMP"} != 1	5m	warning	ifAdminStatus on server <hostname> ({{ \$labels.instance }}) is down.

Правила для СХД Tatlin (tatlin.yaml)

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
TatlinDiskState	Disk <disk_id> ERROR in slot <disk_slot> on <device>	(tatlinHwDiskState * on(tatlinHwDiskDiskId, job) group_left (tatlinHwDiskSlot) tatlinHwDiskSlot) * on (tatlinHwDiskDiskId, job) group_left (tatlinHwDiskModel) tatlinHwDiskModel != 1	-	warning	Disk <disk_id> (<disk_model>) in slot <disk_slot> on <device> is in ERROR state
TatlinEthDown	Network interface <port_name> on <device> is DOWN	tatlinHwEthState != 1	-	critical	Network interface <port_name> (<sp_name>) on <device> is in DOWN state

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
TatlinSPDown	Storage processor <sp_name> on <device> is DOWN	tatlinHwSpState != 1	-	critical	Storage processor <sp_name> on <device> is in DOWN state

Правила для UserGate (usergate.yaml)

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
usergate_powerSupply1Status	Power supply 1 on <hostname> is down	usergate_powerSupply1Status != 1	-	critical	Power supply 1 on <hostname> (<instance>) is down.
usergate_powerSupply2Status	Power supply 2 on <hostname> is down	usergate_powerSupply2Status != 1	-	critical	Power supply 2 on <hostname> (<instance>) is down.

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
usergate_haStatus	Usergate <hostname> HA state is changing	changes(usergate_haStatus) > 0	5m	warning	HA status is changing in Usergate <hostname> for 5 minutes
usergate_cpuLoad	CPU load on Usergate <hostname> is too high	usergate_cpuLoad > 60	2m	warning	CPU load on Usergate <hostname> is too high (<value>)
usergate_raidStatus	Problem with RAID status on Usergate <hostname>	usergate_raidStatus != 1	-	warning	Problem with RAID status on Usergate <hostname>
usergate_memoryUsed	Memory usage on Usergate <hostname> is too high	usergate_memoryUsed > 60	2m	warning	Memory usage on Usergate <hostname> is too high (<value>)

Правила Node Exporter (node-exporter.yaml, расширенный набор по сравнению с базовыми правилами, возможны пересечения)

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
HostOutOfMemory	Host <hostname> out of memory	$(\text{node_memory_MemAvailable_bytes} / \text{node_memory_MemTotal_bytes} * 100 < 10) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	2m	warning	Node memory is filling up (< 10% left)
HostMemoryUnderMemoryPressure	Host memory under memory pressure on <hostname>	$(\text{rate}(\text{node_vmstat_pgmajfault}[1\text{m}]) > 1000) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	2m	warning	The node is under heavy memory pressure. High rate of major page faults
HostMemoryIsUnderutilized	Host Memory is underutilized on <hostname>	$(100 - (\text{avg_over_time}(\text{node_memory_MemAvailable_bytes}[30\text{m}]) / \text{node_memory_MemTotal_bytes} * 100) < 20) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	1w	info	Node memory is < 20% for 1 week. Consider reducing memory space. (instance <instance>)
HostUnusualNetworkThroughputIn	Unusual network input throughput on <hostname>	$(\text{rate}(\text{node_network_receive_bytes_total}[2\text{m}]) / 1024 / 1024 > 1000) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	5m	warning	Host network interfaces are probably receiving too much data (> 1000 MB/s) for 5 minutes

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
HostUnusualNetworkThroughputOut	Host unusual network output throughput on <hostname>	$(\text{rate}(\text{node_network_transmit_bytes_total}[2\text{m}]) / 1024 / 1024 > 1000) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	5m	warning	Host network interfaces are probably sending too much data (> 1000 MB/s) for 5 minutes
HostUnusualDiskReadRate	Unusual disk read rate on <hostname>	$(\text{rate}(\text{node_disk_read_bytes_total}[2\text{m}]) / 1024 / 1024 > 200) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	5m	warning	Disk is probably reading too much data (> 200 MB/s) for 5 minutes
HostUnusualDiskWriteRate	Unusual disk write rate on <hostname>	$(\text{rate}(\text{node_disk_written_bytes_total}[2\text{m}]) / 1024 / 1024 > 200) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	5m	warning	Disk is probably writing too much data (> 200 MB/s) for 5 minutes
HostOutOfDiskSpaceWarn	Host <hostname> is near out of disk space	$((\text{node_filesystem_avail_bytes} * 100) / \text{node_filesystem_size_bytes} < 20 \text{ and ON } (\text{instance}, \text{device}, \text{mountpoint}) \text{node_filesystem_readonly} == 0) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	2m	warning	Disk is almost full (< 20% left)
HostOutOfDiskSpaceCrit	Host <hostname> is out of disk space	$((\text{node_filesystem_avail_bytes} * 100) / \text{node_filesystem_size_bytes} < 10 \text{ and ON } (\text{instance}, \text{device}, \text{mountpoint}) \text{node_filesystem_readonly} == 0) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	2m	critical	Disk is almost full (< 10% left)

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
HostDiskWillFillIn24Hours	Host disk will fill in 24 hours on <hostname>	$((\text{node_filesystem_avail_bytes} * 100) / \text{node_filesystem_size_bytes} < 10 \text{ and ON } (\text{instance, device, mountpoint}) \text{predict_linear}(\text{node_filesystem_avail_bytes}\{\text{fstype!}\sim\text{"tmpfs"}\}[1\text{h}], 24 * 3600) < 0 \text{ and ON } (\text{instance, device, mountpoint}) \text{node_filesystem_readonly} == 0) * \text{on}(\text{instance, hostname, job, group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{"."}\})$	2m	warning	Filesystem is predicted to run out of space within the next 24 hours at current write rate
HostOutOfInodes	Host <hostname> is out of inodes	$(\text{node_filesystem_files_free}\{\text{fstype!}=\text{"msdosfs"}\} / \text{node_filesystem_files}\{\text{fstype!}=\text{"msdosfs"}\} * 100 < 10 \text{ and ON } (\text{instance, device, mountpoint}) \text{node_filesystem_readonly} == 0) * \text{on}(\text{instance, hostname, job, group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{"."}\})$	2m	warning	Disk is almost running out of available inodes (< 10% left)
HostNodesWillFillIn24Hours	Host inodes will fill in 24 hours on <hostname>	$(\text{node_filesystem_files_free}\{\text{fstype!}=\text{"msdosfs"}\} / \text{node_filesystem_files}\{\text{fstype!}=\text{"msdosfs"}\} * 100 < 10 \text{ and } \text{predict_linear}(\text{node_filesystem_files_free}\{\text{fstype!}=\text{"msdosfs"}\}[1\text{h}], 24 * 3600) < 0 \text{ and ON } (\text{instance, device, mountpoint}) \text{node_filesystem_readonly}\{\text{fstype!}=\text{"msdosfs"}\} == 0) * \text{on}(\text{instance, hostname, job, group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{"."}\})$	2m	warning	Filesystem is predicted to run out of inodes within the next 24 hours at current write rate

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
HostUnusualDiskReadLatency	Unusual disk read latency on <hostname>	$(\text{rate}(\text{node_disk_read_time_seconds_total}[1\text{m}]) / \text{rate}(\text{node_disk_reads_completed_total}[1\text{m}]) > 0.1 \text{ and } \text{rate}(\text{node_disk_reads_completed_total}[1\text{m}]) > 0) * \text{on}(\text{instance, hostname, job, group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{.}+\}$	2m	warning	Disk latency is growing (read operations > 100ms)
HostUnusualDiskWriteLatency	Unusual disk write latency on <hostname>	$(\text{rate}(\text{node_disk_write_time_seconds_total}[1\text{m}]) / \text{rate}(\text{node_disk_writes_completed_total}[1\text{m}]) > 0.1 \text{ and } \text{rate}(\text{node_disk_writes_completed_total}[1\text{m}]) > 0) * \text{on}(\text{instance, hostname, job, group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{.}+\}$	2m	warning	Disk latency is growing (write operations > 100ms)
HostHighCPUload	High CPU load on <hostname>	$(\text{sum by}(\text{instance, hostname, job, group})(\text{avg by}(\text{mode, instance, hostname})(\text{rate}(\text{node_cpu_seconds_total}\{\text{mode}!\text{="idle"}\}[2\text{m}])))) > 0.8) * \text{on}(\text{instance, hostname, job, group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{.}+\}$	2m	warning	CPU load is > 80%
HostCpulsUnderutilized	CPU is underutilized on <hostname>	$(100 - (\text{rate}(\text{node_cpu_seconds_total}\{\text{mode}=\text{"idle"}\}[30\text{m}]) * 100) < 20) * \text{on}(\text{instance, hostname, job, group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{.}+\}$	1w	warning	CPU load is < 20% for 1 week. Consider reducing the number of CPUs.

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
HostCpuStealNoisyNeighbor	CPU steal noisy neighbor on <hostname>	$(\text{avg by}(\text{instance}, \text{hostname}, \text{job}, \text{group}) (\text{rate}(\text{node_cpu_seconds_total}\{\text{mode}=\text{"steal"}\}[5\text{m}])) * 100 > 10) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	-	warning	CPU steal is > 10%. A noisy neighbor is killing VM performances or a spot instance may be out of credit.
HostCpuHighIowait	High CPU iowait on <hostname>	$(\text{avg by}(\text{instance}, \text{hostname}, \text{job}, \text{group}) (\text{rate}(\text{node_cpu_seconds_total}\{\text{mode}=\text{"iowait"}\}[5\text{m}])) * 100 > 10) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	-	warning	CPU iowait > 10%. A high iowait means that you are disk or network bound.
HostUnusualDiskIO	Unusual disk IO on <hostname>	$(\text{rate}(\text{node_disk_io_time_seconds_total}[1\text{m}]) > 0.5) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	2m	warning	Time spent in IO is too high on <hostname>. Check storage for issues.
HostContextSwitching	High context switching on <hostname>	$((\text{rate}(\text{node_context_switches_total}[5\text{m}])) / (\text{count without}(\text{cpu}, \text{mode}) (\text{node_cpu_seconds_total}\{\text{mode}=\text{"idle"}\}))) > 10000) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	-	warning	Context switching is growing on the node (> 10000 / CPU / s)
HostSwapIsFillingUp	Host swap is filling up on <hostname>	$((1 - (\text{node_memory_SwapFree_bytes} / \text{node_memory_SwapTotal_bytes})) * 100 > 80) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{".+"}\}$	2m	warning	Swap is filling up (>80%)

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
HostSystemdServiceCrashed	Service <name> crashed on <hostname>	<code>(node_systemd_unit_state{state="failed"} == 1) * on(instance, hostname, job, group) group_left (nodename) node_uname_info{nodename=~".+"}</code>	-	warning	systemd service <name> is crashed
HostPhysicalComponentTooHot	Host physical component too hot on <hostname>	<code>((node_hwmon_temp_celsius * ignoring(label) group_left(instance, job, node, sensor) node_hwmon_sensor_label{label!="tctl"} > 75)) * on(instance, hostname, job, group) group_left (nodename) node_uname_info{nodename=~".+"}</code>	2m	warning	Physical hardware component too hot
HostNodeOvertemperatureAlarm	Host node overtemperature alarm on <hostname>	<code>(node_hwmon_temp_crit_alarm_celsius == 1) * on(instance, hostname, job, group) group_left (nodename) node_uname_info{nodename=~".+"}</code>	-	critical	Physical node temperature alarm triggered
HostKernelVersionDeviations	Host kernel version deviations on <hostname>	<code>(count(sum(label_replace(node_uname_info, "kernel", "\$1", "release", "[0-9]+.[0-9]+.[0-9]+.*")) by (kernel)) > 1) * on(instance, hostname, job, group) group_left (nodename) node_uname_info{nodename=~".+"}</code>	6h	warning	Different kernel versions are running
HostOomKillDetected	Host OOM kill detected on <hostname>	<code>(increase(node_vmstat_oom_kill[1m]) > 0) * on(instance, hostname, job, group) group_left (nodename) node_uname_info{nodename=~".+"}</code>	-	warning	OOM kill detected

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
HostEdacCorrectableErrorsDetected	Host EDAC Correctable Errors detected on <hostname>	$(\text{increase}(\text{node_edac_correctable_errors_total}[5\text{m}]) > 0) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{.}+\}$	-	info	Host <hostname> has had <value> correctable memory errors reported by EDAC in the last 5 minutes
HostEdacUncorrectableErrorsDetected	Host EDAC Uncorrectable Errors detected on <hostname>	$(\text{node_edac_uncorrectable_errors_total} > 0) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{.}+\}$	-	warning	Host <hostname> has had <value> uncorrectable memory errors reported by EDAC in the last minutes.
HostNetworkReceiveErrors	Host Network Receive Errors on <hostname>	$(\text{rate}(\text{node_network_receive_errs_total}[2\text{m}]) / \text{rate}(\text{node_network_receive_packets_total}[2\text{m}]) > 0.01) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{.}+\}$	2m	warning	Host <hostname> interface <device> has encountered <value> receive errors in the last two minutes.
HostNetworkTransmitErrors	Host Network Transmit Errors on <hostname>	$(\text{rate}(\text{node_network_transmit_errs_total}[2\text{m}]) / \text{rate}(\text{node_network_transmit_packets_total}[2\text{m}]) > 0.01) * \text{on}(\text{instance}, \text{hostname}, \text{job}, \text{group}) \text{group_left}(\text{nodename}) \text{node_uname_info}\{\text{nodename}=\sim\text{.}+\}$	2m	warning	Host <hostname> interface <device> has encountered <value> transmit errors in the last two minutes.

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
HostCPUCountChanged	Host <hostname> CPU count is changed	changes(count(count(node_cpu_seconds_total != 0) by (cpu, hostname, group, job, instance)) by (hostname, group, job, instance)) > 0	-	critical	CPU count is changed on <hostname> (<instance>)

Правила Vector (обнаружение в логах):

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
event_megaraid_error	MEGARAI D VD <device> failure	создание - при обнаружении в /var/log/syslog выражения: kernel: megaraid_sas .* - VD (.+) is now (PARTIALLY DEGRADED DEGRADED OFFLINE) восстановление по строке kernel: megaraid_sas .* - VD (.+) is now OPTIMAL	-	critical	Изменение состояния RAID-массива для серверов с установленным megaraid_sas

Название события	Краткое описание	Условие создания события	Минимальная длительность	Уровень критичности	Описание
event_aldro_error	Not listening for new connections	создание - при обнаружении в /var/log/dirsrv/slapd-<DOMAIN>/errors выражения: ERR - .*? - Not listening for new connections - too many fds open восстановление по строке ERR - .*? - Listening for new connections again	-	critical	

6.16 6.15 Мониторинг SSL сертификатов (агент мониторинга)



Для корректного отображения устройств на дашборде в grafana следует использовать значение тега component как IP адрес, имя устройства или имя сайта/файла, чей сертификат проверяется с помощью ssl-exporter. Так же необходимо отмечать объект тегом ssl_type как file или web, соответственно проверяемому сертификату (см. пример ниже).

6.16.1 Ключи запуска ssl-exporter

Для мониторинга SSL сертификатов используется [ssl-exporter](https://github.com/ribbybibby/ssl_exporter)¹⁴⁰.

Экспортер позволяет работать с различными источниками: проверка сертификатов сайтов с помощью TCP и HTTPS проверок, проверка файлов ключей локально и по HTTP протоколу, проверка файлов и секретов Kubernetes.

Ключи запуска экспортера:

Ключ	Описание
--config.file=""	SSL exporter configuration file

¹⁴⁰ https://github.com/ribbybibby/ssl_exporter

Ключ	Описание
<code>--web.metrics-path="/metrics"</code>	Path under which to expose metrics
<code>--web.listen-address=:9219</code>	Address to listen on for web interface and telemetry
<code>--web.probe-path="/probe"</code>	Path under which to expose the probe endpoint
<code>--log.level=info</code>	Only log messages with the given severity or above. Valid levels: [debug, info, warn, error, fatal]
<code>--log.format=logfmt</code>	Output format of log messages

6.16.2 Метрики

Metric	Meaning	Labels	Probers
<code>ssl_cert_not_after</code>	The date after which a peer certificate expires. Expressed as a Unix Epoch Time.	serial_no, issuer_cn, cn, dnsnames, ips, emails, ou	tcp, https
<code>ssl_cert_not_before</code>	The date before which a peer certificate is not valid. Expressed as a Unix Epoch Time.	serial_no, issuer_cn, cn, dnsnames, ips, emails, ou	tcp, https
<code>ssl_file_cert_not_after</code>	The date after which a certificate found by the file prober expires. Expressed as a Unix Epoch Time.	file, serial_no, issuer_cn, cn, dnsnames, ips, emails, ou	file
<code>ssl_file_cert_not_before</code>	The date before which a certificate found by the file prober is not valid. Expressed as a Unix Epoch Time.	file, serial_no, issuer_cn, cn, dnsnames, ips, emails, ou	file
<code>ssl_kubernetes_certificate_not_after</code>	The date after which a certificate found by the kubernetes prober expires. Expressed as a Unix Epoch Time.	namespace, secret, key, serial_no, issuer_cn, cn, dnsnames, ips, emails, ou	kubernetes

Metric	Meaning	Labels	Probers
ssl_kubernetes_certificate_not_before	The date before which a certificate found by the kubernetes prober is not valid. Expressed as a Unix Epoch Time.	namespace, secret, key, serial_no, issuer_cn, cn, dnsnames, ips, emails, ou	kubernetes
ssl_kubeconfig_certificate_not_after	The date after which a certificate found by the kubeconfig prober expires. Expressed as a Unix Epoch Time.	kubeconfig, name, type, serial_no, issuer_cn, cn, dnsnames, ips, emails, ou	kubeconfig
ssl_kubeconfig_certificate_not_before	The date before which a certificate found by the kubeconfig prober is not valid. Expressed as a Unix Epoch Time.	kubeconfig, name, type, serial_no, issuer_cn, cn, dnsnames, ips, emails, ou	kubeconfig
ssl_ocsp_response_next_update	The nextUpdate value in the OCSP response. Expressed as a Unix Epoch Time		tcp, https
ssl_ocsp_response_produced_at	The producedAt value in the OCSP response. Expressed as a Unix Epoch Time		tcp, https
ssl_ocsp_response_revoked_at	The revocationTime value in the OCSP response. Expressed as a Unix Epoch Time		tcp, https
ssl_ocsp_response_status	The status in the OCSP response. 0=Good 1=Revoked 2=Unknown		tcp, https
ssl_ocsp_response_stapled	Does the connection state contain a stapled OCSP response? Boolean.		tcp, https
ssl_ocsp_response_this_update	The thisUpdate value in the OCSP response. Expressed as a Unix Epoch Time		tcp, https
ssl_probe_success	Was the probe successful? Boolean.		all
ssl_prober	The prober used by the exporter to connect to the target. Boolean.	prober	all

Metric	Meaning	Labels	Probers
ssl_tls_version_info	The TLS version used. Always 1.	version	tcp, https
ssl_verified_cert_not_after	The date after which a certificate in the verified chain expires. Expressed as a Unix Epoch Time.	chain_no, serial_no, issuer_cn, cn, dnsnames, ips, emails, ou	tcp, https
ssl_verified_cert_not_before	The date before which a certificate in the verified chain is not valid. Expressed as a Unix Epoch Time.	chain_no, serial_no, issuer_cn, cn, dnsnames, ips, emails, ou	tcp, https

[Документация по настройке probes](#)¹⁴¹.

6.16.3 Запуск ssl-exporter с помощью агента мониторинга

Для установки экспортера как сервис см. [главу 3](#) руководства.

Для корректного запуска экспортера необходимо выполнить следующие настройки:

1. В конфигурационном файле в разделе exporters указываем имя запускаемого экспортера и параметр is_custom:

```
exporters:
  - name: ssl-exporter
    is_custom: true
    args: "--config.file=./exporters/ssl-exporter.yml"
    health_address: 127.0.0.1:9219/metrics
```

- Имя в поле name должно совпадать с названием бинарного файла с применением дополнительных условий именования сторонних экспортеров (пример: astramon-ssl-exporter-custom)
- Параметр is_custom говорит агенту, что это сторонний экспортер, но агент может управлять им (запускать, делать health check, останавливать и сообщать статус работы экспортера в Config API). Так же это дает понять агенту не использовать другую функциональность, что доступна экспортерам, специально разработанным для АМ.
- Параметр args сообщает экспортеру параметры запуска, в них указывается путь к файлу конфигурации экспортера.

2. Следом необходимо выполнить настройку в разделе metrics:

¹⁴¹ https://github.com/ribbybibby/ssl_exporter

```

custom_targets:
  - name: ssl_exporter1
    component: mail.ru
    target: 127.0.0.1:9219
    metrics_path: /probe?target=mail.ru:443
    labels:
      ssl_type: web
  - name: ssl_exporter2
    component: local_pem_file
    target: 127.0.0.1:9219
    metrics_path: /probe?module=file&target=/etc/ssl/*.pem
    labels:
      ssl_type: file

```

- Имя в поле `name` должно быть уникальным (особенности работы `vmaagent`). В примере мы будем собирать данные с двух таргетов.
- Параметр `component` необходим для добавления лейбла компонент, чтобы можно было различать метрики по таргетам на одном хосту.
- `target` описывает адрес и порт, с которого будут считываться метрики.
- `metrics_path` позволяет обратиться к конкретному эндпоинту, для получения метрик. По умолчанию `/metrics`, но в нашем случае этот эндпоинт выводит метрики по запуску скриптов, а не метрики самих скриптов. Если необходимо можно создать еще одну запись и собирать непосредственно метрики из `/metrics`.
- `ssl_type` позволяет разделять проверки web ssl сертификатов и сертификатов из локальных файлов на дашборде в grafana (используются типы: `web`, `file`).

6.16.4 Настройка конфигурационного файла

При создании конфигурационного файла стоит обращать внимание на следующие моменты:

Пример конфигурационного файла

```

modules:
  https:
    prober: https
  https_insecure:
    prober: https
    tls_config:
      insecure_skip_verify: true
  tcp:
    prober: tcp
  tcp_client_auth:
    prober: tcp
    tls_config:

```

```

    ca_file: /home/agladkov/temp/agent/GlobalSignRSA2018.crt
tcp_client_auth_error:
  prober: tcp
  tls_config:
    ca_file: /home/agladkov/temp/agent/ca.crt
file:
  prober: file
file_ca_certificates:
  prober: file
  target: /home/agladkov/temp/agent/ca.crt

```

- Каждый модуль содержит в себе настройки запроса при использовании имени этого модуля в probe. (например: /probe?module=https_insecure&target=test.local.ru¹⁴²).
- Модуль `https_insecure` позволяет проверять самоподписные сертификаты, отключая проверку центром сертификации.
- Модуль `tcp_client_auth` позволяет проверять сертификат на подпись конкретным центром сертификации. Если он не будет совпадать, то мы получим `ssl_probe_success 0`.
- При использовании конфигурационного файла в нем должны быть в явном виде все типы probe (которые планируется использовать), иначе проверки не будут работать.

6.17 6.16 Запуск и мониторинг локальных скриптов (агент мониторинга)

6.17.1 Ключи запуска script-exporter

Для запуска и мониторинга локальных скриптов используется `script-exporter`¹⁴³.

Экспортер позволяет запускать и впоследствии мониторить локальные скрипты.

Ключи запуска экспортера:

Ключ	Описание
<code>-config.file file</code>	Configuration file in YAML format. (default "config.yaml")
<code>-config.check</code>	Do not run the exporter. Only check the configuration file and exit (0 if the Configuration file is valid, 1 otherwise)
<code>-web.listen-address string</code>	Address to listen on for web interface and telemetry. (default ":9469")

¹⁴² <http://test.local.ru>

¹⁴³ https://github.com/ricoberger/script_exporter

Ключ	Описание
-create-token	Create bearer token for authentication
-log.level string	Only log messages with the given severity or above. One of: [debug, info, warn, error] (default "info")
-log.format string	Output format of log messages. One of: [logfmt, json] (default "logfmt")
-log.env	Log environment variables used by a script
-noargs	Restrict script to accept arguments, for security issues
-timeout-offset seconds	Offset to subtract from Prometheus-supplied timeout in seconds. (default 0.5)

Дополнительные примеры можно найти на [странице](#)¹⁴⁴.

6.17.2 Метрики

Метрики различаются в зависимости от созданных скриптов.

Эндпоинт /metrics показывает стандартные метрики типа go, proc, promhttp. Кроме этого выводит метрики характеристик работы кастомных скриптов:

Название метрики	Описание метрики	Пример метрики
scripts_duration_seconds	Длительность выполнения кастомного скрипта (в зависимости от квантиля)	scripts_duration_seconds{script="ping",quantile="1"} 2.062227309
scripts_duration_seconds_count	Количество измерений времени для запусков скрипта	scripts_duration_seconds_count{script="ping"} 89
scripts_requests_total	Количество запусков скрипта	scripts_requests_total{script="ping"} 89

¹⁴⁴ https://github.com/ricoberger/script_exporter/tree/main/examples

Эндпоинты запуска скриптов выглядят примерно так (в зависимости от типа скрипта): `/probe?script=ping&prefix=test¶ms=target&target=ya.ru145`. При обращении к ним, непосредственно запускаются кастомные скрипты. Общие метрики на подобных эндпоинтах:

Название метрики	Описание метрики	Пример метрики
<code>script_success</code>	Статус выполнения скрипта	<code>script_success{script="ping"} 1</code>
<code>script_duration_seconds</code>	Длительность выполнения кастомного скрипта (конкретный запуск)	<code>script_duration_seconds{script="ping"} 2.026552</code>
<code>script_exit_code</code>	Итоговый код завершения скрипта	<code>script_exit_code{script="ping"} 0</code>

Кроме этого, если скрипт формирует вывод метрики по стандарту prometheus, то будет выведена эта метрика (далее показано в примере использования).

6.17.3 Запуск script-exporter с помощью агента мониторинга

Для установки экспортера как сервис см. главу 3 руководства.

1. Для запуска экспортера через monitoring-agent необходимо:

- Скопировать бинарник экспортера в папку `exporters` (папка для хранения бинарников экспортеров, которыми управляет агент, задается полем `exporters.directory` в конфигурации агента);
- Скопировать конфигурационный файл в папку, которая будет указана в аргументах запуска экспортера;
- Скопировать файлы кастомных скриптов, которые будут запускаться экспортером, в папку, указанную в конфигурации экспортера. (Нами успешно протестированы shell скрипты и go бинарники).

2. В конфигурационном файле в разделе `exporters` указываем имя запускаемого экспортера и параметр `is_custom`:

```
- name: script_exporter
  is_custom: true
  args: "-config.file /etc/astra-monitoring/exporters/script_config.yaml"
```

- Имя в поле `name` должно совпадать с названием бинарного файла с применением дополнительных условий именования сторонних экспортеров (пример: `astramon-script_exporter-custom`).

¹⁴⁵ <http://ya.ru>

- Параметр `is_custom` говорит агенту, что это сторонний экспортер, но агент может управлять им (запускать, делать health check, останавливать и сообщать статус работы экспортера в Config API). Так же это дает понять агенту не использовать другую функциональность, что доступна экспортерам, специально разработанным для АМ.
- Параметр `args` сообщает экспортеру параметры запуска, в них указывается путь к файлу конфигурации экспортера.

3. Следом необходимо выполнить настройку в разделе `metrics`:

```
custom_targets:
- name: node_exporter
  component: node-exporter
  target: 0.0.0.0:9100
- name: script_exporter_1
  component: script-exporter1
  target: 0.0.0.0:9469
  metrics_path: "/probe?script=ping&prefix=test&params=target&target=ya.ru"
- name: script_exporter_2
  component: script-exporter2
  scrape_interval: 300s
  target: 0.0.0.0:9469
  metrics_path: "/probe?script=show_metrics"
```

- Имя в поле `name` должно быть уникальным (особенности работы `vmagent`). В примере мы будем собирать данные с двух таргетов.
- Параметр `component` необходим для добавления лейбла компонент, чтобы можно было различать метрики по экспортерам на одном хосту.
- `target` описывает адрес и порт, с которого будут считываться метрики.
- `metrics_path` позволяет обратиться к конкретному эндпоинту, для получения метрик. По умолчанию `/metrics`, но в нашем случае этот эндпоинт выводит метрики по запуску скриптов, а не метрики самих скриптов. Если необходимо можно создать еще одну запись и собирать непосредственно метрики из `/metrics`.
- `scrape_interval` дает возможность запускать кастомные скрипты с нужной периодичностью (если общий интервал опроса метрик отличается).

6.17.4 Пример запуска кастомного скрипта

Для примера используем следующий кастомный скрипт:

```
#!/bin/sh
random_number=$((1 + RANDOM % 100))

echo "# HELP test_output_for_custom_metrics"
echo "# TYPE test_output_for_custom_metrics gauge"
echo "test_output_for_custom_metrics{} $random_number"
```

При запуске скрипта он выводит метрику и генерирует случайное ее значение от 1 до 100.

Создаем файл с названием `show_metrics.sh` в папке `exporters` в рабочей файле агента мониторинга. Агент будет искать скрипт в данной папке. Даем файлу разрешение на запуск (`chmod +x`).

Описываем скрипт в конфигурационном файле экспортера:

```
scripts:
  - name: show_metrics
    command: /etc/astra-monitoring/exporters/show_metrics.sh
```

В конфигурационном файле агента создаем необходимые записи (см. выше).

После запуска в браузере переходим на эндпоинт http://localhost:9469/probe?script=show_metrics:

```
# HELP script_success Script exit status (0 = error, 1 = success).
# TYPE script_success gauge
script_success{script="show_metrics"} 1
# HELP script_duration_seconds Script execution time, in seconds.
# TYPE script_duration_seconds gauge
script_duration_seconds{script="show_metrics"} 0.008075
# HELP script_exit_code The exit code of the script.
# TYPE script_exit_code gauge
script_exit_code{script="show_metrics"} 0
# HELP test_output_for_custom_metrics
# TYPE test_output_for_custom_metrics gauge
test_output_for_custom_metrics{} 43
```

Метрика с этого эндпоинта может собираться различными средствами и использоваться в дальнейшем.

6.17.5 Особенность запуска на Windows

Рекомендуется запускать экспортер как службу windows.

Кроме этого скрипты, которые запускает экспортер, должны быть запускаемыми файлами (`exe`). На данный момент `bat`, `vbs` и `powershell` скрипты корректно не запускаются экспортером.

Настройка сбора метрик в разделе `metrics` конфигурационного файла агента мониторинга ничем не отличается от настройки для Linux систем.

6.18 6.17 Мониторинг SNMP трапов

6.18.1 Введение

В текущей реализации поддерживаются версии SNMP v1 и v2c.

Файл правил состоит из пяти основных частей:

- * Header - Документация с определением переменных,
- * Lookup - Таблицы приведения числовых обозначений в человекочитаемый формат,
- * Preparation - Подготовка служебной информации и предварительная обработка трапов,

- * Body - Ветвление правил по SNMP трапам,
- * Footer - Подготовка результата для дальнейшей обработки.

6.18.2 Header - подготовка переменных

```
if (typeof $OID1 == "undefined") $OID1 = "";
if (typeof $OID2 == "undefined") $OID2 = "";
if (typeof $OID3 == "undefined") $OID3 = "";
if (typeof $OID4 == "undefined") $OID4 = "";
if (typeof $OID5 == "undefined") $OID5 = "";
if (typeof $OID6 == "undefined") $OID6 = "";
if (typeof $OID7 == "undefined") $OID7 = "";
if (typeof $OID8 == "undefined") $OID8 = "";
if (typeof $OID9 == "undefined") $OID9 = "";
if (typeof $OID10 == "undefined") $OID10 = "";
if (typeof $OID11 == "undefined") $OID11 = "";
if (typeof $OID12 == "undefined") $OID12 = "";

if (typeof $1 == "undefined") $1 = "";
if (typeof $2 == "undefined") $2 = "";
if (typeof $3 == "undefined") $3 = "";
if (typeof $4 == "undefined") $4 = "";
if (typeof $5 == "undefined") $5 = "";
if (typeof $6 == "undefined") $6 = "";
if (typeof $7 == "undefined") $7 = "";
if (typeof $8 == "undefined") $8 = "";
if (typeof $9 == "undefined") $9 = "";
if (typeof $10 == "undefined") $10 = "";
if (typeof $11 == "undefined") $11 = "";
if (typeof $12 == "undefined") $12 = "";

_Manager = "";
_Agent = "";
_Class = "";
_Node = "";
_NodeAlias = "";
_AlertKey = "";
_AlertGroup = "";
_Summary = "";
_Severity = "";
_Type = "";
_Identifier = "";
_ExpireTime = 0;
_FirstOccurrence = new Date($ReceivedTime * 1000);
_LastOccurrence = _FirstOccurrence;
_Alert = false;
```

6.18.3 Lookup - таблицы приведения

```

IANAifType = {
    "1": "Other", // other - none of the following
    "2": "Regular 1822", // regular1822
    "3": "HDH 1822", // hdh1822
    "4": "DDN X.25", // ddnX25
    "5": "RFC-877 X.25", // rfc877x25
    "6": "Ethernet CSMA-CD", // ethernetCsmacd
    "7": "ISO 8802.3 CSMA-CD", // iso88023Csmacd
    "8": "ISO 8802.4 Token Bus", // iso88024TokenBus
    ...
    "212": "Voice FGD Exchange Access North American", // voiceFGDEANA - voice FGD
Exchange Access North American
    "213": "Voice Direct Inward Dialing" // voiceDID - voice Direct Inward Dialing
}

ifAdminStatus = {
    "1": "Up", // up - ready to pass packets
    "2": "Down", // down
    "3": "Testing" // testing - in some test mode
}

ifOperStatus = {
    "1": "Up", // up - ready to pass packets
    "2": "Down", // down
    "3": "Testing", // testing - in some test mode
    "4": "Unknown", // unknown - status can not be determined for some reason.
    "5": "Dormant", // dormant
    "6": "Not Present", // notPresent - some component is missing
    "7": "Lower Layer Down" // lowerLayerDown - down due to state of lower-layer
interface(s)
}

```

6.18.4 Preparation - подготовка данных

```

// Устанавливаем стандартный Agent, Manager и Class
_Manager = "SNMP Trap adapter on " + hostname()
_Agent = "SNMPagent"
_Class = "300"

// Проверяем если получили SNMPv2 трап и ковертируем его в SNMPv1 стиль
if (typeof $notify !== "undefined") // Если $notify существует то это SNMPv2 трап
{
    _Node = $PeerAddress
    _NodeAlias = $PeerIPAddress
}

```

```

if (regmatch($notify, "\.iso\.")) {
    $notify = ".1." + extract($notify, "\.iso\.(.*)")
}

if (nmatch($notify, ".1.3.6.1.6.3.1.1.5")) // SNMPv2 Generic трап
{
    $Enterprise = extract($notify, "(.*)\[0-9]+\$")
    $specific_trap = "0"
    $generic_trap = (int(extract($notify, ".*\[0-9]+\$")) - 1).toString()
} else // Enterprise Specific трап
{
    if (match(extract($notify, "\.[0-9]+\.[0-9]+\$"), "0")) {
        $Enterprise = extract($notify, "(.*)\[0-9]+\.[0-9]+\$")
    } else {
        $Enterprise = extract($notify, "(.*)\[0-9]+\$")
    }
    $specific_trap = extract($notify, ".*\[0-9]+\$")
    $generic_trap = "6"
}
} else // Это SNMPv1 трап
{
    _Node = $Node
    if (typeof $IPAddress !== "undefined")
    {
        _NodeAlias = $IPAddress
    } else {
        _NodeAlias = $IPaddress
    }
}
}

```

6.18.5 Body - определение трапов и извлечение необходимых данных для создания события

На примере конфигурации для Cisco Generic трапов (это часть файла; полный файл правил для Cisco Generic можно найти в [репозитории](#)¹⁴⁶):

```

if (match($generic_trap, "6")) // Проверяем если Enterprise Specific трап
{
    switch ($Enterprise) {
        case "dummy case statement": // Это позволит избежать синтаксических ошибок
        при отсутствующих кейсай
            break
        // В этом сегменте определяем Enterprise Specific трапы не относящиеся к
        Generic, как в следующем примере
        case ".1.3.6.1.4.1.9.9.43.2": // Если $Enterprises соответствует Cisco и
        трапу ciscoConfigManMIBNotificationPrefix
            switch ($specific_trap) {

```

146 https://gitlab.158-160-60-159.sslip.io/astra-monitoring-icl/snmp-agent/-/blob/main/rules/cisco_generic.js

```

    case "1": // - ciscoConfigManEvent
        // ...
        break
    case "2": // - ccmCLIRunningConfigChanged
        // ...
        break
    case "3": // Если $specific_trap соответствует ccmCTIDRolledOver
        _AlertGroup = "cisco"
        _Agent = "cisco"
        _Summary = "ccmCTIDRolledOver: This notification indicates that
the Config Change Tracking ID has rolled over and will be reset."
        _Severity = 3
        _Type = 1
        _ExpireTime = 10
        _Identifier = _Node + _Summary
        break
    default:
        _Summary = "Unknown specific trap number (" + $specific_trap + ")
received for enterprise " + $Enterprise
        _Identifier = _Node + $Enterprise + $generic_trap +
$specific_trap + _Agent + _Manager
        _Severity = 3
        _Type = 1
    }
    break
default:
    $EventID = $Enterprise + "|" + $generic_trap + "|" + $specific_trap
    switch ($EventID) {
        case "dummy case statement": // ЭТО ПОЗВОЛИТ ИБЕЖАТЬ СИНТАКСИЧЕСКИХ
ошибок при отсутствующих кейсай
            break
        default:
            _Summary = "Enterprise ID Not Found (see details): " +
$Enterprise + " " + $generic_trap + " " + $specific_trap
            _Severity = 2
            _Type = 1
    }
    _Identifier = _Node + " " + $Enterprise + " " + $generic_trap + " " +
$specific_trap + " " + _AlertGroup + " " + _AlertKey + " " + _Agent + " " + _Manager
}
} else // Проверяем если Generic трап
{
    // Стантартно для Generic
    _Agent = "Generic"
    switch ($generic_trap) {
        // В этом сегменте определяем только SNMP Generic трапы
        case "0": // coldStart
            _AlertGroup = "Generic"
            _AlertKey = ""
            _Summary = "Cold Start"
            _Severity = 2
            _Type = 1
    }
}

```

```

        _Identifier = _Node + " " + _AlertGroup + " " + _Agent + " " + _Manager +
" " + $generic_trap
        break
        case "1": // warmStart
            // ...
            break
        case "2": // linkDown
            // ...
            break
        // ...
        default:
            _Summary = "Invalid Generic Trap Number: " + $Enterprise + " " +
$generic_trap + " " + $specific_trap
            _Severity = 2
            _Type = 0
            _Identifier = _Node + " " + $Enterprise + " " + $generic_trap + " " +
$specific_trap + " " + _Agent + " " + _Manager
        }
        switch ($Enterprise) {
            // В этом сегменте определяем SNMP Generic трапы которые относятся к другим MIB
или к Enterprise (Generic трапы имеющие дополнительную Enterprise информацию)
            case ".1.3.6.1.6.3.1.1.5": // RFC1907 - SNMPv2-MIB
                if (nmatch($OID1, ".1.3.6.1.4.1.9") || nmatch($OID2, ".1.3.6.1.4.1.9") ||
nmatch($OID4, ".1.3.6.1.4.1.9")) { // Если трап содержит один из данных OID, значит
это Enterprise Generic трап принадлежащий Cisco
                    _Agent = "Generic-Cisco SNMPv2"
                    _Class = "40057"
                    switch ($generic_trap) {
                        case "0": // При попадании в этот кейс, выполняем следующий что
бы не дублировать код
                            case "1": // Для трапов coldStart и warmStart одинаковая ветка,
так что можно их обрабатывать одинаково
                                $sysUptime = $1
                                $whyReload = $2
                                details($sysUptime, $whyReload)

                                _Summary = _Summary + ": " + $2
                                _Identifier = _Identifier + " " + $2
                                break
                            case "2":
                            case "3": // Так же как и для предыдущих, linkDown и linkUp
обрабатываем одинаково
                                // ...
                                break
                                // ...
                            default:
                                }
                } if (nmatch($OID6, ".1.3.6.1.4.1.9")) { // Если Generic трап содержит
данный OID, значит это Generic-Cisco трап
                    _Agent = "Generic-Cisco SNMPv2"
                    _Class = "40057"
                    switch ($generic_trap) {
                        case "2":

```

```

        case "3": // linkDown, linkUp
            $ifIndex = $1
            $ifAdminStatus = ifAdminStatus[$2] + " ( " + $2 + " )"
            $ifOperStatus = ifOperStatus[$3] + " ( " + $3 + " )"
            $ifDescr = $4
            $ifType = IANAifType[$5] + " ( " + $5 + " )" // Берем
описание из Lookup таблицы
            $locIfReason = $6
            details($ifIndex, $ifAdminStatus, $ifOperStatus, $ifDescr,
$ifType, $locIfReason)

            switch ($2) {
                case "2": // Down
                    _Summary = _Summary + ", Administratively: " + $6 + "
( " + $4 + " )"

                    _Severity = 2
                    break
                case "3": // Testing - in some test mode
                    _Summary = _Summary + ", Testing: " + $6 + " ( " +
$4 + " )"

                    _Severity = 2
                    break
                default:
                    _Summary = _Summary + ": " + $6 + " ( " + $4 + " )"
            }
            _Identifier = _Identifier + " "
            break
        default:
            // ...
    }
} else { // Если не подходит не под онид из перечисленных вариантов,
значит это Generic IETF SNMPv2 Трап
    _Agent = "Generic-IETF-SNMPv2-MIB"

    switch ($generic_trap) {
        case "2":
        case "3": // linkDown, linkUp
            // ...
            break
        // ...
        default:
            // ...
    }
}
break
// ...
default:
    if (match($Enterprise, "")) {
        _Summary = "Dummy Enterprise! If you matched here you have problems!
(see details)"
    }
    else // Обрабатываем неизвестный Enterprise как SNMPv1 Generic трап
    {

```

```

        _Agent = "Generic-Unknown"
        switch ($generic_trap) {
            case "2":
            case "3": // linkDown, linkUp
                //...
                break
            default:
                _Summary = _Summary + " ( Enterprise = " + $Enterprise +
" )"
        }
    }
}
}
}

```

6.18.6 Footer - подготовка данных для передачи в хранимую процедуру

```

set("Manager", _Manager)
set("Agent", _Agent)
set("Class", _Class)
set("Node", _Node)
set("NodeAlias", _NodeAlias)
set("AlertKey", _AlertKey)
set("AlertGroup", _AlertGroup)
set("Summary", _Summary)
set("Severity", _Severity)
set("Type", _Type)
set("Identifier", _Identifier)
set("ExpireTime", _ExpireTime)
set("FirstOccurrence", _FirstOccurrence.getTime() / 1000)
set("LastOccurrence", _LastOccurrence.getTime() / 1000)
set("Alert", _Alert)

```

6.18.7 Алерты

Для отправки алертов необходимо установить параметр **_Alert = true** в соответствующих ветках файла правил где необходимо создать алерт.

Пример ветки трапа с алертом

```

// ...
case "2": // linkDown
    _AlertGroup = "Generic Link Status"
    _AlertKey = "ifEntry." + $1
    _Summary = "Link Down"
    _Severity = 5

```

```

        _Type = 1
        _Identifier = _Node + " " + _AlertKey + " " + _AlertGroup + " " + _Agent
+ " " + _Manager
        _Alert = true // При установке данного параметра в true будет создан
алерт на текущий трап
        break
// ...

```

6.18.8 Доступные параметры в файле правил

Поле	Версии SNMP	Описание
\$Community	V1 и V2c	Строка community SNMP
\$contextEngineID	V3	Context Engine ID
\$Enterprise	V1	Строка enterprise SNMP
\$generic_trap	V1	Целочисленное значение Generic SNMP-трапа
\$IPAddress	V1, V2c и V3	IP-адрес (происхождение SNMP-трапа)
\$Node	V1, V2c и V3	Имя узла (происхождение SNMP-трапа). IP-адрес (если имя узла не может быть разрешено)
\$notify	V2c и V3	Специфическое поле notify V2c
\$PeerAddress	V1, V2c и V3	Имя хоста или IP-адрес, откуда был получен SNMP-трап
\$PeerIPAddress	V1, V2c и V3	IP-адрес, откуда был получен SNMP-трап
\$Protocol	V1, V2c и V3	Протокол полученного трапа. Это может быть либо UDP, либо TCP
\$ReceivedPort	V1, V2c и V3	Номер порта, откуда был получен SNMP-трап. Это определяется свойством Port

Поле	Версии SNMP	Описание
\$ReceivedTime	V1, V2c и V3	Время получения SNMP-пакета с сетевого интерфейса
\$ReqId	V1	Идентификатор запроса SNMP
\$securityEngineID	V3	Security Engine ID авторитетного SNMP-субъекта Для трапов - это Engine ID источника трапа
\$securityLevel	V3	Уровень безопасности trap или inform: noAuth - trap или inform не имеют аутентификации и конфиденциальности, authNoPriv - trap или inform имеют аутентификацию, но без конфиденциальности, authPriv - trap или inform имели аутентификацию и конфиденциальность
\$securityName	V3	Security Name, используемое для аутентификации трапа
\$SNMP_Version	V1, V2c и V3	Имеет значение 1 для трапов SNMP V1 и значение 2 для трапов SNMP V2c
\$specific_trap	V1	Целочисленное значение Specific SNMP-трапа
\$UpTime	V1 и V2c	Аптайм SNMP трапов, выраженное в целых числах
\$n_hex		Шестнадцатеричное представление переменных varbind

6.18.9 Примечания

1. [Инструкция по установке и настройке.](#)
2. В текущей реализации нет генератора файла правил из MIB файлов, поэтому для новых устройств файл правил необходимо создавать/дополнять вручную.

3. В файле правил используется синтаксис JavaScript.

6.19 6.18 Приложения. Метрики, используемые для мониторинга

- [1. Метрики, предоставляемые windows-exporter \(see page 204\)](#)
- [2. Метрики, используемые для мониторинга FreeIPA \(see page 208\)](#)
- [3. Метрики, используемые для мониторинга ALD Pro \(see page 219\)](#)
- [4. Метрики, используемые для мониторинга ПК СВ "Брест" \(see page 242\)](#)
- [5. Метрики, используемые для мониторинга RuPost \(see page 278\)](#)
- [6. Метрики, используемые для мониторинга RuBackup \(see page 294\)](#)
- [7. Метрики, используемые для мониторинга Termidesk \(see page 299\)](#)
- [8. Метрики, используемые для мониторинга Billmanager \(see page 331\)](#)

6.19.1 1. Метрики, предоставляемые windows-exporter

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД
1	Операционная система	Метрика показывает название операционной системы	Windows dashboard (row "Resource overview")/ operating system	windows_os_info
2	Общее использование места на диске C	Метрика показывает сколько всего памяти использовано на диске C	Windows dashboard (row "Resource overview")/ C:/ usage	windows_logical_disk_free_bytes windows_logical_disk_size_bytes
3	Общее использование ЦПУ, %	Метрика показывает общее использование ЦПУ	Windows dashboard (row "Resource overview")/ Total CPU usage	windows_cpu_time_total
4	Общее использование оперативной памяти, %	Метрика показывает итоговое потребление оперативной памяти	Windows dashboard (row "Resource overview")/ Memory usage	windows_os_physical_memory_free_bytes windows_cs_physical_memory_bytes

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД
5	Сетевой трафик	Метрика показывает величину входящего и исходящего сетевого трафика	Windows dashboard (row "Resource overview")/ Network details of the maximum traffic network card	windows_net_bytes_sent_total windows_net_bytes_received_total
6	Максимальная скорость записи и чтения	Метрика показывает максимальную скорость записи и чтения со всех дисков системы	Windows dashboard (row "Resource overview")/ Maximum disk read and write details	windows_logical_disk_read_bytes_total windows_logical_disk_write_bytes_total
7	Максимальное число операций на дисках	Метрика показывает максимальное число операций ввода/вывода на дисках системы	Windows dashboard (row "Resource overview")/ Maximum disk IO details	windows_logical_disk_reads_total windows_logical_disk_writes_total
8	Uptime системы	Метрика показывает время с последней перезагрузки системы	Windows dashboard (row "Resource details")/ Up time	windows_system_system_up_time
9	Количество логических процессоров	Метрика показывает сколько логических процессоров в системе	Windows dashboard (row "Resource details")/ CPUs	windows_cs_logical_processors
10	Общий объем оперативной памяти	Метрика показывает количество оперативной памяти в системе	Windows dashboard (row "Resource details")/ Total memory	windows_cs_physical_memory_bytes
11	Текущее значение загруженности ЦПУ, %	Метрика показывает текущее среднее значение использования процессора за последние две минуты	Windows dashboard (row "Resource details")/ CPU usage	windows_cpu_time_total

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД
12	Текущее значения использования оперативной памяти, %	Метрика показывает текущее значения использования оперативной памяти	Windows dashboard (row "Resource details")/ Memory usage	windows_os_physical_memory_free_bytes windows_cs_physical_memory_bytes
13	Использованное место на всех разделах, %	Метрика показывает сколько места занято на разделах системы	Windows dashboard (row "Resource details")/ Usage space of each partition	windows_logical_disk_free_bytes windows_logical_disk_size_bytes
14	Общий статус всех сервисов	Метрика показывает сколько сервисов находится в различных статусах	Windows dashboard (row "Resource details")/ Service status	windows_service_state
15	Количество запущенных процессов	Метрика показывает сколько запущенно процессов в системе	Windows dashboard (row "Resource details")/ Number of processes	windows_os_processes
16	Использование ЦПУ по времени	Метрика показывает использование процессора по времени	Windows dashboard (row "Resource details")/ CPU usage history	windows_cpu_time_total
17	Использование памяти по типам	Метрика показывает использование памяти в зависимости от типа памяти	Windows dashboard (row "Resource details")/ Memory details	windows_cs_physical_memory_bytes windows_os_physical_memory_free_bytes windows_os_virtual_memory_bytes windows_os_virtual_memory_free_bytes
18	Свободное место на дисках системы	Метрика показывает оставшееся место на дисках системы	Windows dashboard (row "Resource details")/ Free disk space	windows_logical_disk_free_bytes windows_logical_disk_size_bytes

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД
19	Скорость передачи сетевых данных	Метрика показывает скорость входящих и исходящих сетевых данных	Windows dashboard (row "Resource details")/ Network details	windows_net_bytes_sent_total windows_net_bytes_received_total
20	Скорость записи и чтения с дисков	Метрика показывает скорость записи и чтения на дисках системы	Windows dashboard (row "Resource details")/ Disk read and write	windows_logical_disk_read_bytes_total windows_logical_disk_write_bytes_total
21	Количество операций ввода/вывода	Метрика показывает количество операций ввода/вывода на дисках системы	Windows dashboard (row "Resource details")/ Disk IO	windows_logical_disk_reads_total windows_logical_disk_writes_total
22	Использование сетевой пропускной способности, %	Метрика показывает процент использования пропускной способности сетевой карты	Windows dashboard (row "Resource details")/ Network usage	windows_net_bytes_total windows_net_current_bandwidth_bytes
23	Количество ошибок сетевой карты	Метрика показывает сколько ошибок происходит при входящем и исходящем трафике	Windows dashboard (row "Resource details")/ Network discarded/error packets	windows_net_packets_outbound_discarded_total windows_net_packets_outbound_errors_total windows_net_packets_received_discarded_total windows_net_packets_received_errors_total
24	Количество потоков управления ядра	Метрика показывает величину потоков управления ядра	Windows dashboard (row "Resource details")/ System threads	windows_system_threads
25	Величина системных исключений	Метрика показывает количество системных исключений	Windows dashboard (row "Resource details")/ System exception dispatches	windows_system_exception_dispatches_total

6.19.2 2. Метрики, используемые для мониторинга FreeIPA

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
1	Проверка возможности входа в LDAP тестового пользователя	Метрика отрабатывает локальную попытку входа пользователя в контроллер домена	FreeIPA state/ User connection test FreeIPA state with history data/ Test user connection	<p><code>ipa_user_connect</code></p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется <code>freeipa-exporter</code> (github.com/ccin2p3/go-freeipa/freeipa¹⁴⁷). 2. Создан технический пользователь в FreeIPA 3. Пользователь прописан в <code>freeipa-exporter.conf</code> 4. Пароль для пользователя должен быть зашифрован Base64 <p>Специальный технический пользователь локально подключается к LDAP и происходит попытка получить статус этого пользователя. Если LDAP возвращает ошибку, то метрика переходит в NOK и возвращает 0. Если попытка успешная то возвращает 1</p>

¹⁴⁷ <http://github.com/ccin2p3/go-freeipa/freeipa>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
2	Кол-во неуспешных попыток авторизации в LDAP	Метрика подсчитывает кол-во неуспешных попыток аутентификации на контроллере домена	FreeIPA state with history data/ Failed user login attempts	<p>Развернуть</p> <p>Метрика собирается из логов.</p> <p>Пререквизиты:</p> <ol style="list-style-type: none"> 1. Используется сервис vector 2. В /etc/vector/conf.d/freeipa.toml прописан путь к логу /var/log/dirsrv/*/access 3. Там же настроен парсинг этого лога: <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre>(.*(?P<valid_cred>Invalid credentials))</pre> </div> <p>После неуспешной попытки входа в логе появляется запись "Invalid credentials". Vector обрабатывает запись и пишет ее в БД с необходимыми labels</p> <p>Комментарий: Необходимо помнить, что не все системы, которые аутентифицируют пользователей через FreeIPA, возвращают неуспешные попытки контроллеру. Поэтому в логе собирается информация не обо всех неуспешных попытках входа пользователей</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
3	Время (длительность) прохождения аутентификации тестовым пользователем	Метрика подсчитывает время, необходимое для проведения аутентификации пользователя	FreeIPA state/ User connection time FreeIPA state with history data/ User connection time	<p>ipa_time_for_connect</p> <p>Развернуть</p> <p>Пререквизиты:</p> <ol style="list-style-type: none"> 1. Используется freeipa-exporter (github.com/ccin2p3/go-freeipa/freeipa¹⁴⁸). 2. Создан технический пользователь в FreeIPA 3. Пользователь прописан в freeipa-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 <p>Специальный технический пользователь локально подключается к LDAP и происходит попытка получить статус этого пользователя. Замеряется время прохождения этого процесса. Если происходит ошибка, то возвращается -1</p>

148 <http://github.com/ccin2p3/go-freeipa/freeipa>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
4	Статус репликации на контроллере домена Было: Кол-во конфликтов репликации (между кд)	Метрика проверяет статус репликации между контроллерами	FreeIPA state/ Replication FreeIPA state with history data/ Summary replication for each controller; Replication status; Last replication time (time from now)	ipa_ldap_replication_status; ipa_last_ldap_replication_time Развернуть Пререквизиты: 1. Используется freeipa-exporter. 2. Пользователь должен иметь роль с привилегией на чтение Read Replication Agreements ¹⁴⁹ Специальный технический пользователь локально подключается к LDAP, делает запрос на статус репликации. Ответ анализируется на наличие настроенных репликаций к другим хостам, их статус и последнее время репликации между ними Комментарий: Метрика не выдает конкретику по ошибкам репликации

149 <https://ipatest2.astra.dcs.lan/ipa/ui/#/e/permission/details/Read%20Replication%20Agreements>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
5	Кол-во заблокированных пользователей	Метрика подсчитывает кол-во выключенных пользователей	FreeIPA state/ Disabled users FreeIPA state with history data/ Amount of disabled users	<p>ipa_disabled_users</p> <p>Развернуть</p> <p>Пререквизиты:</p> <ol style="list-style-type: none"> 1. Используется freeipa-exporter(gopkg.in/ldap.v2¹⁵⁰). 2. Создан технический пользователь в FreeIPA 3. Пользователь прописан в freeipa-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 <p>Специальный технический пользователь локально подключается к LDAP и собирает данные обо всех пользователях, у которых параметр " nsAccountLock " = TRUE</p> <p>Комментарий: Данные о выключенных пользователях синхронизированы между контроллерами, поэтому кол-во одинаковое на всех контроллерах</p>

150 <http://gopkg.in/ldap.v2>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
6	Проверка соответствия заданному dns имени - ip адресу при запросе через локальные dns настройки	Метрика проверяет работу DNS службы, выполняя преобразование заданного имени в ip-адрес (DNS запрос) в соответствии с локальными настройками DNS.	FreeIPA state/ Internal DNS test	dns_internal_check Развернуть Пререквизиты: <ol style="list-style-type: none"> Используется freeipa-exporter. В freeipa-exporter.conf прописаны необходимые параметры для DNS запроса: имя (fqdn), ожидаемый ip-адрес (может быть localhost), тип проверки (internal) Экспортер делает запрос к внутреннему DNS-серверу для выяснения ip-адреса, соответствующего имени из запроса. Если ip адрес совпадает с ожидаемым, проверка пройдена
			FreeIPA state with history data/ DNS check for expected IP address using local dns settings	
7	Проверка соответствия заданному dns имени - ip адресу при запросе через локальные dns настройки	Метрика проверяет работу DNS службы, выполняя преобразование заданного имени в ip-адрес (DNS запрос) с обращением к заданному внешнему серверу DNS	FreeIPA state/ External DNS test	dns_external_check Развернуть Пререквизиты: <ol style="list-style-type: none"> Используется freeipa-exporter. В freeipa-exporter.conf прописаны необходимые параметры для DNS запроса: имя (fqdn), ожидаемый ip-адрес, адрес внешнего DNS-сервера, тип проверки (external) Экспортер делает запрос к указанному в настройках внешнему DNS-серверу для выяснения ip-адреса, соответствующего имени из запроса. Если ip адрес совпадает с ожидаемым, проверка пройдена
			FreeIPA state with history data/ DNS check for expected IP address using external dns server	

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
8	Проверка доступности dns сервера контроллера домена путём выполнения локального dns запроса	Метрика проверяет доступность DNS сервера, указанного в локальных настройках	FreeIPA state with history data/ DNS resolve check using local dns settings	dns_internal_resolve Развернуть Пререквизиты: 1. Используется freeipa-exporter. 2. В freeipa-exporter.conf прописаны необходимые параметры для DNS запроса: имя (fqdn), ожидаемый ip-адрес (может быть localhost), тип проверки (internal) Экспортер делает запрос к DNS-серверу указанному в локальных настройках для получения ответа на dns запрос. Ожидается ответ от dns сервера
9	Проверка доступности global dns путём выполнения dns запроса	Метрика проверяет доступность external DNS сервера	FreeIPA state with history data/ DNS resolve check using query to external dns server	dns_external_resolve Развернуть Пререквизиты: 1. Используется freeipa-exporter. 2. В freeipa-exporter.conf прописаны необходимые параметры для DNS запроса: имя (fqdn), ожидаемый ip-адрес, адрес внешнего DNS-сервера, тип проверки (external) Экспортер делает запрос к внешнему DNS-серверу для получения ответа на dns запрос. Ожидается ответ от dns сервера

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
10	Статус ntp (ошибок синхронизации с вышестоящим и ntp серверами)	Метрика проверяет ошибки синхронизации контроллера с вышестоящими серверами времени	FreeIPA state/ NTP	<p>ipa_ntp_check</p> <p>Развернуть</p> <p>Пререквизиты:</p> <ol style="list-style-type: none"> Используется локальная служба <code>chrony</code> <p>Проверяется вывод команды <code>chronyc tracking</code>. Проверка осуществляется по полю: <code>"Leap status"</code>. Статус <code>"Normal"</code> считается ОК и возвращается 0. Статус <code>"Insert second"</code> считается NOK и возвращается 1. Статус <code>"Delete second"</code> считается NOK и возвращается 2. Статус <code>"Not synchronized"</code> считается NOK и возвращается 3. Любой другой статус или ошибка получения данных возвращается 4</p> <p>Комментарий: В документации FreeIPA сказано, что при установке контроллера всегда используется служба <code>chrony</code> для синхронизации времени</p>
			FreeIPA state with history data/ NTP Status	
11	Статус служб FreeIPA	Метрика проверяет состояние служб контроллера домена	FreeIPA state/ Services state	<p><code>systemd_unit_state_id</code></p> <p>Развернуть</p> <p>Пререквизиты:</p> <ol style="list-style-type: none"> Используется <code>systemd_exporter</code>. Список сервисов фильтруется в самом запросе.
			FreeIPA state with history data/ FreeIPA summary services state on all controllers; FreeIPA services on \$hostname	

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
12	Кол-во записей (изменений) в базу LDAP за единицу времени	Метрика собирает количество операций в LDAP в единицу времени по их типам	FreeIPA state with history data/ Changes in LDAP	<p>Развернуть</p> <p>Метрика собирается из логов.</p> <p>Пререквизиты:</p> <ol style="list-style-type: none"> 1. Используется сервис vector 2. В /etc/vector/conf.d/freeipa.toml прописан путь к логу /var/log/dirsrv/*/access 3. Там же настроен парсинг этого лога: <pre>(.*conn=(? P<connection_id>[0-9]+) op=(? P<operation_id>[0-9]+) (? P<ldap_operation>[A-Z]+) (.* \$))</pre> <p>В логе собирается информация по типам операций ldap_operation. Все операции собираются, им присваивается labels и vector пишет все в БД. Метрика строится по следующим типам операций: ADD, DEL, MOD</p> <p>Комментарий: Данные об изменениях в LDAP синхронизированы между контроллерами, поэтому кол-во одинаковое на всех контроллерах</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
13	Кол-во логинов в web ui от пользователей с админ правами за единицу времени	Метрика собирает кол-во успешных входов пользователей уровня Администратор через веб консоль FreeIPA	FreeIPA state/ Admin UI logins	<p>Развернуть</p> <p>Метрика собирается из логов.</p> <p>Пререквизиты:</p> <ol style="list-style-type: none"> 1. Используется сервис vector 2. В /etc/vector/conf.d/freeipa.toml прописан путь к логу /var/log/auth.log 3. Там же настроен парсинг этого лога: <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre>^(.*), (?P<username>[a-zA-Z0-9-_. /]+)@[a-zA-Z0-9-_. /]+)</pre> </div> <p>Из лога собираются вход любого пользователя с меткой: "TGS_REQ". На уровне grafana делается выборка пользователей с именем, которое содержит "adm"</p> <p>Комментарий: Необходимо изменение метрики в будущем, чтобы учитывать вход любого пользователя с правами Администратора</p>
14	Кол-во событий уровня error во всех логах системы за единицу времени	Метрика собирает кол-во ошибок уровня error из предоставленных логов	FreeIPA state with history data/ Errors in log files	<p>Развернуть</p> <p>Метрика собирается из логов.</p> <p>Пререквизиты:</p> <ol style="list-style-type: none"> 1. Используется сервис vector 2. В /etc/vector/conf.d/freeipa.toml прописан путь к логам 3. Работает по всем логам, которые собираются на данном хосту <p>Все события уровня ERROR собираются, помечаются label "Severity" для дальнейшего анализа на стороне grafana</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
15	Кол-во Search запросов на контроллере домена	Метрика собирает кол-во Search запросов к контроллеру	FreeIPA state with history data/ Search requests in LDAP	<p>Развернуть</p> <p>Метрика собирается из логов.</p> <p>Пререквизиты:</p> <ol style="list-style-type: none"> 1. Используется сервис vector 2. В /etc/vector/conf.d/freeipa.toml прописан путь к логу /var/log/dirsrv/*/access 3. Там же настроен парсинг этого лога: <pre>(.*conn=(? P<connection_id>[0-9]+) op=(? P<operation_id>[0-9]+) (? P<ldap_operation>[A-Z]+) (.* \$))</pre> <p>В логе собирается информация по типам операций ldap_operation. Все операции собираются, им присваивается labels и vector пишет все в БД. Метрика строится по следующим типам операций: SRCH</p>
16	Нагрузка на процессор на контроллерах	Метрика показывает загрузку процессора на хостах контроллера домена	FreeIPA state with history data/ Total CPU usage, %; CPU	<p>node_cpu_seconds_total</p> <p>Развернуть</p> <p>Пререквизиты:</p> <ol style="list-style-type: none"> 1. Используется prometheus node exporter. <p>Стандартный способ сбора метрики с хостов основанный на стандартном node exporter. Метрика образовывается путем вычитания текущего времени использования ЦПУ и минуту назад</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
17	Использование памяти	Метрика показывает использование памяти на хостах контроллера домена	FreeIPA state with history data/ Total memory usage, %; Memory Stack	<p><code>node_memory_MemTotal_bytes</code>; <code>node_memory_Buffers_bytes</code>; <code>node_memory_MemFree_bytes</code>; <code>node_memory_Cached_bytes</code></p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> Используется <code>prometheus node exporter</code>. <p>Стандартный способ сбора метрики с хостов основанный на стандартном <code>node exporter</code>. Метрика образовывается путем вычитания из максимального количества ОЗУ всей использованной памяти</p>

6.19.3 3. Метрики, используемые для мониторинга ALD Pro

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
1	Статус службы <code>krb5kdc</code>	Метрика показывает состояние сервиса <code>krb5kdc</code>	Служба каталогов	ALD-Pro directory service/ Status of <code>krb5kdc</code> service	<p><code>systemd_unit_state_id</code></p> <p>Прerequisites:</p> <ol style="list-style-type: none"> Используется <code>systemd exporter</code>
2	Статус службы <code>ipa-custodia</code>	Метрика показывает состояние сервиса <code>ipa-custodia</code>	Служба каталогов	ALD-Pro directory service/ Status of <code>ipa-custodia</code> service	<p><code>systemd_unit_state_id</code></p> <p>Прerequisites:</p> <ol style="list-style-type: none"> Используется <code>systemd exporter</code>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
3	Статус службы apache2	Метрика показывает состояние сервиса httpd/apache2	Служба каталогов	ALD-Pro directory service/ Status of apache2 service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
4	Статус службы dirsrv	Метрика показывает состояние сервиса dirsrv	Служба каталогов	ALD-Pro directory service/ Status of directory service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
5	Статус службы bind9	Метрика показывает состояние сервиса bind9	Служба каталогов	ALD-Pro directory service/ Status of named service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
6	Статус службы krb5-admin	Метрика показывает состояние сервиса krb5-admin	Служба каталогов	ALD-Pro directory service/ Status of kadmin service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
7	Статус службы ipa-dnskeysyncd	Метрика показывает состояние сервиса ipa-dnskeysyncd	Служба каталогов	ALD-Pro directory service/ Status of ipa-dnskeysyncd service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
8	Статус службы ipa-otpd.socket	Метрика показывает состояние сервиса ipa-otpd.socket	Служба каталогов	ALD-Pro directory service/ Status of ipa-otpd service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
9	Срок службы LDAP сертификата	Метрика показывает время, оставшееся до истечения срока действия LDAP сертификата ALD-Pro	Служба каталогов	ALD-Pro directory service/ LDAP certificate expiration time	ald_ldap_cert_exp_check Прerequisites: 1. Используется ipahealthcheck-exporter Делается запрос openssl для получения времени истечения сертификата
10	Срок службы WEB сертификата портала ALD-Pro	Метрика показывает время, оставшееся до истечения срока действия WEB сертификата портала ALD-Pro	Служба каталогов	ALD-Pro directory service/ WEB certificate expiration time	ald_web_cert_exp_check Прerequisites: 1. Используется ipahealthcheck-exporter Делается запрос openssl для получения времени истечения сертификата
11	Статус службы isc-dhcp-server	Метрика показывает состояние сервиса isc-dhcp-server	Динамическая настройка узлов	ALD-Pro dynamic node configuration/ Status of isc-dhcp-server service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
12	Статус службы smbд	Метрика показывает состояние сервиса smbд	Общий доступ к файлам	ALD-Pro file sharing/ Status of samba service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
13	Статус службы apache2	Метрика показывает состояние сервиса httpd/apache2	Установка ОС по сети	ALD-Pro installing the OS over the network/ Status of apache2 service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
14	Статус службы rabbitmq-server	Метрика показывает состояние сервиса rabbitmq-server	Установка ОС по сети	ALD-Pro installing the OS over the network/ Status of rabbitmq-server service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
15	Статус службы postgresql	Метрика показывает состояние сервиса postgresql	Установка ОС по сети	ALD-Pro installing the OS over the network/ Status of postgresql service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
16	Статус службы salt-master	Метрика показывает состояние сервиса salt-master	Портал управления	ALD-Pro management portal/ Status of salt-master service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
17	Статус службы apache2	Метрика показывает состояние сервиса httpd/apache2	Портал управления	ALD-Pro management portal/ Status of apache2 service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
18	Статус службы celery	Метрика показывает состояние сервиса celery	Портал управления	ALD-Pro management portal/ Status of celery service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
19	Статус службы rabbitmq-server	Метрика показывает состояние сервиса rabbitmq-server	Портал управления	ALD-Pro management portal/ Status of rabbitmq-server service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
20	Статус службы aldpro-mp-services	Метрика показывает состояние сервиса aldpro-mp-services	Портал управления	ALD-Pro management portal/ Status of aldpro-mp-services service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
21	Доступность WEB портала ALD-Pro	Метрика показывает доступность web портала ALD-Pro	Портал управления	ALD-Pro management portal/ Connect to WEB portal	ald_portal_check Прerequisites: 1. Используется freeipa-exporter 2. В freeipa-exporter.conf прописан URL для проверки портала Измеряется доступность https странички портала
22	Использование ЦПУ службами ALD-Pro	Метрика показывает использование ЦПУ определенными службами	Портал управления	ALD-Pro management portal/ Использование ЦП процессами служб, %	systemd_unit_cpu_usage, node_cpu_seconds_total Прerequisites: 1. Используется systemd exporter 2. Используется node exporter Комментарий: Сервисы, которые выводятся метрикой: dirsrv, salt-master, httpd, apache2, celery, postgresql, rabbitmq-server, salt-master, bind9

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
23	Использование памяти службами ALD-Pro	Метрика показывает использование памяти определенными службами	Портал управления	ALD-Pro management portal/ Использование памяти процессами служб	systemd_unit_memory_usage Прerequisites: 1. Используется systemd exporter Комментарий: Сервисы, которые выводятся метрикой: dirsrv, salt-master, httpd, apache2, celery, postgresql, rabbitmq-server, salt-master, bind9
24	Статус службы zabbix-agent	Метрика показывает состояние сервиса zabbix-agent	Мониторинг	ALD-Pro monitoring/ Status of zabbix-agent service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
25	Статус службы zabbix-server	Метрика показывает состояние сервиса zabbix-server	Мониторинг	ALD-Pro monitoring/ Status of zabbix-server service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
26	Статус службы apache2	Метрика показывает состояние сервиса apache2	Мониторинг	ALD-Pro monitoring/ Status of apache2 service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
27	Статус службы postgresql	Метрика показывает состояние сервиса postgresql	Мониторинг	ALD-Pro monitoring/ Status of postgresql service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
28	Проверка работы ДНС через локальный запрос	Метрика проверяет работу DNS службы, выполняя преобразование заданного имени в ip-адрес (DNS запрос) в соответствии с локальными настройками DNS.	Разрешение имен	ALD-Pro name resolution/ Internal DNS test	<p><code>dns_internal_check</code></p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется freeipa-exporter. 2. В freeipa-exporter.conf прописаны необходимые параметры для DNS запроса: имя (fqdn), ожидаемый ip-адрес (может быть localhost), тип проверки (internal) <p>Экспортер делает запрос к внутреннему DNS-серверу для выяснения ip-адреса, соответствующего имени из запроса. Если ip адрес совпадает с ожидаемым, проверка пройдена</p>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
29	Проверка работы ДНС через запрос к внешнему ДНС	Метрика проверяет работу DNS службы, выполняя преобразование заданного имени в ip-адрес (DNS запрос) с обращением к заданному внешнему серверу DNS	Разрешение имен	ALD-Pro name resolution/ External DNS test	<p>dns_external_check</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется freeipa-exporter. 2. В freeipa-exporter.conf прописаны необходимые параметры для DNS запроса: имя (fqdn), ожидаемый ip-адрес, адрес внешнего DNS-сервера, тип проверки (external) <p>Экспортер делает запрос к указанному в настройках внешнему DNS-серверу для выяснения ip-адреса, соответствующего имени из запроса. Если ip адрес совпадает с ожидаемым, проверка пройдена</p>
30	Статус службы cups	Метрика показывает состояние сервиса cups	Печать	ALD-Pro printing system/ Status of cups service	<p>systemd_unit_state_id</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется systemd exporter
31	Статус службы apache2	Метрика показывает состояние сервиса apache2	Репозитории программного обеспечения	ALD-Pro software repositories/ Status of apache2 service	<p>systemd_unit_state_id</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется systemd exporter

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
32	Статус службы postgresql	Метрика показывает состояние сервиса postgresql	Репозитории программного обеспечения	ALD-Pro software repositories/ Status of postgresql service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
33	Статус службы rabbitmq-server	Метрика показывает состояние сервиса rabbitmq-server	Репозитории программного обеспечения	ALD-Pro software repositories/ Status of rabbitmq-server service	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
34	Проверка синхронизации времени	Метрика проверяет ошибки синхронизации контроллера с вышестоящими серверами времени.	Синхронизация времени	ALD-Pro time synchronization/ NTP Status	ipa_ntp_check Прerequisites: 1. Используется локальная служба chrony Проверяется вывод команды <code>chronyc tracking</code> . Проверка осуществляется по полю: "Leap status". Статус "Normal" считается ОК и возвращается 0. Статус "Insert second" считается NOK и возвращается 1. Статус "Delete second" считается NOK и возвращается 2. Статус "Not synchronized" считается NOK и возвращается 3. Любой другой статус или ошибка получения данных возвращается 4

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
35	Статус службы chrony	Метрика показывает состояние сервиса chrony	Синхронизация времени	ALD-Pro time synchronization/ Status of chronyservice	systemd_unit_state_id Прerequisites: 1. Используется systemd exporter
36	Проверка возможности входа в LDAP тестового пользователя	Метрика отрабатывает локальную попытку входа пользователя в контроллер домена	FreeIPA	ALD-Pro state with history data/Test user connection	ipa_user_connect Развернуть Прerequisites: 1. Используется freeipa-exporter(github.com/ccin2p3/go-freeipa ¹⁵¹) . 2. Создан технический пользователь в FreeIPA 3. Пользователь прописан в ipahealthcheck-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 Специальный технический пользователь локально подключается к LDAP и происходит попытка получить статус этого пользователя. Если LDAP возвращает ошибку, то метрика переходит в NOK и возвращает 0. Если попытка успешная то возвращает 1

151 <http://github.com/ccin2p3/go-freeipa/freeipa>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
37	Кол-во неуспешных попыток авторизации в LDAP	Метрика подсчитывает кол-во неуспешных попыток аутентификации на контроллере домена	FreeIPA	ALD-Pro state with history data/ Failed user login attempts	<p>Развернуть</p> <p>Метрика собирается из логов.</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется сервис vector 2. В /etc/vector/conf.d/freeipa.toml прописан путь к логу /var/log/dirsrv/*/access 3. Там же настроен парсинг этого лога: <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre>(.*(?P<valid_cred>Invalid credentials))</pre> </div> <p>После неуспешной попытки входа в логе появляется запись "Invalid credentials". Vector обрабатывает запись и пишет ее в БД с необходимыми labels</p> <p>Комментарий: Необходимо помнить, что не все системы, которые аутентифицируют пользователей через FreeIPA, возвращают неуспешные попытки контроллеру. Поэтому в логе собирается информация не обо всех неуспешных попытках входа пользователей</p>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
38	Время (длительность) прохождения аутентификации тестовым пользователем	Метрика подсчитывает время, необходимое для проведения аутентификации пользователя	FreeIPA	ALD-Pro state with history data/ User connection time	<p>ipa_time_for_connect</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется freeipa-exporter(github.com/ccin2p3/go-freeipa¹⁵²) 2. Создан технический пользователь в FreeIPA 3. Пользователь прописан в freeipa-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 <p>Специальный технический пользователь локально подключается к LDAP и происходит попытка получить статус этого пользователя. Замеряется время прохождения этого процесса. Если происходит ошибка, то возвращается -1</p>

152 <http://github.com/ccin2p3/go-freeipa/freeipa>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
39	<p>Статус репликации на контроллере домена</p> <p>Было: Кол-во конфликтов репликации (между кд)</p>	<p>Метрика проверяет статус репликации между контроллерами</p>	FreeIPA	<p>ALD-Pro state with history data/ Summary replication for each controller; Replication status; Last replication time (time from now)</p>	<p>ipa_ldap_replication_status; ipa_last_ldap_replication_time</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется freeipa-exporter. 2. Пользователь должен иметь роль с привилегией на чтение Read Replication Agreements¹⁵³ <p>Специальный технический пользователь локально подключается к LDAP, делает запрос на статус репликации. Ответ анализируется на наличие настроенных репликаций к другим хостам, их статус и последнее время репликации между ними</p> <p>Комментарий: Метрика не выдает конкретику по ошибкам репликации</p>

153 <https://ipatest2.astra.dcs.lan/ipa/ui/#/e/permission/details/Read%20Replication%20Agreements>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
40	Кол-во заблокированных пользователей	Метрика подсчитывает кол-во выключенных пользователей	FreeIPA	ALD-Pro state with history data/ Amount of disabled users	<p><code>ipa_disabled_users</code></p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется <code>freeipa-exporter</code> (gopkg.in/ldap.v2¹⁵⁴). 2. Создан технический пользователь в FreeIPA 3. Пользователь прописан в <code>freeipa-exporter.conf</code> 4. Пароль для пользователя должен быть зашифрован Base64 <p>Специальный технический пользователь локально подключается к LDAP и собирает данные обо всех пользователях, у которых параметр <code>"nsAccountLock" = TRUE</code></p> <p>Комментарий: Данные о выключенных пользователях синхронизированы между контроллерами, поэтому кол-во одинаковое на всех контроллерах</p>

154 <http://gopkg.in/ldap.v2>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
41	Проверка соответствия заданному dns имени - ip адресу при запросе через локальные dns настройки	Метрика проверяет работу DNS службы, выполняя преобразование заданного имени в ip-адрес (DNS запрос) в соответствии с локальными настройками DNS.	FreeIPA	ALD-Pro state with history data/DNS check for expected IP address using local dns settings	<p><code>dns_localhost_check</code></p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется freeipa-exporter. 2. В freeipa-exporter.conf прописаны необходимые параметры для DNS запроса: имя (fqdn), ожидаемый ip-адрес (может быть localhost), тип проверки (internal) <p>Экспортер делает запрос к внутреннему DNS-серверу для выяснения ip-адреса, соответствующего имени из запроса. Если ip адрес совпадает с ожидаемым, проверка пройдена</p>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
42	Проверка соответствия заданному dns имени - ip адресу при запросе через локальные dns настройки	Метрика проверяет работу DNS службы, выполняя преобразование заданного имени в ip-адрес (DNS запрос) с обращением к заданному внешнему серверу DNS	FreeIPA	ALD-Pro state with history data/DNS check for expected IP address using external dns server	<p>dns_specific_server</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется freeipa-exporter. 2. В freeipa-exporter.conf прописаны необходимые параметры для DNS запроса: имя (fqdn), ожидаемый ip-адрес, адрес внешнего DNS-сервера, тип проверки (external) <p>Экспортер делает запрос к указанному в настройках внешнему DNS-серверу для выяснения ip-адреса, соответствующего имени из запроса. Если ip адрес совпадает с ожидаемым, проверка пройдена</p>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
43	Статус ntp (ошибок синхронизации с вышестоящими ntp серверами)	Метрика проверяет ошибки синхронизации контроллера с вышестоящими серверами времени	FreeIPA	ALD-Pro state with history data/ NTP Status	<p><code>ipa_ntp_check</code></p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> Используется локальная служба <code>chrony</code> <p>Проверяется вывод команды <code>chronyc tracking</code>. Проверка осуществляется по полю: "Leap status". Статус "Normal" считается ОК и возвращается 0. Статус "Insert second" считается NOK и возвращается 1. Статус "Delete second" считается NOK и возвращается 2. Статус "Not synchronized" считается NOK и возвращается 3. Любой другой статус или ошибка получения данных возвращается 4</p> <p>Комментарий: В документации FreeIPA сказано, что при установке контроллера всегда используется служба <code>chrony</code> для синхронизации времени</p>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
44	Статус служб FreeIPA	Метрика проверяет состояние служб контроллера домена	FreeIPA	ALD-Pro state with history data/ FreeIPA summary services state on all controllers; FreeIPA services on \$hostname	systemd_unit_state_id Развернуть Прerequisites: 1. Используется systemd_exporter. 2. Список сервисов фильтруется в самом запросе.

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
45	Кол-во записей (изменений) в базу LDAP за единицу времени	Метрика собирает количество операций в LDAP в единицу времени по их типам	FreeIPA	ALD-Pro state with history data/ Changes in LDAP	<p>Развернуть</p> <p>Метрика собирается из логов.</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется сервис vector 2. В /etc/vector/conf.d/freeipa.toml прописан путь к логу /var/log/dirsrv/*/access 3. Там же настроен парсинг этого лога: <pre>(.*conn=(? P<connection_id>[0-9]+) op=(? P<operation_id>[0-9]+) (? P<ldap_operation>[A-Z]+)(.* \$))</pre> <p>В логе собирается информация по типам операций ldap_operation. Все операции собираются, им присваивается labels и vector пишет все в БД. Метрика строится по следующим типам операций: ADD, DEL, MOD</p> <p>Комментарий: Данные об изменениях в LDAP синхронизированы между контроллерами, поэтому кол-во одинаковое на всех контроллерах</p>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
46	Кол-во событий уровня error во всех логах системы за единицу времени	Метрика собирает кол-во ошибок уровня error из предоставленных логов	FreeIPA	ALD-Pro state with history data/ Errors in log files	<p>Развернуть</p> <p>Метрика собирается из логов.</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется сервис vector 2. В /etc/vector/conf.d/freeipa.toml прописан путь к логам 3. Работает по всем логам, которые собираются на данном хосту <p>Все события уровня ERROR собираются, помечаются label "Severity" для дальнейшего анализа на стороне grafana</p>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
47	Кол-во Search запросов на контроллере домена	Метрика собирает кол-во Search запросов к контроллеру	FreeIPA	ALD-Pro state with history data/ Search requests in LDAP	<p>Развернуть</p> <p>Метрика собирается из логов.</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется сервис vector 2. В /etc/vector/conf.d/freeipa.toml прописан путь к логу /var/log/dirsrv/*/access 3. Там же настроен парсинг этого лога: <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre>(.*conn=(? P<connection_id>[0-9]+) op=(? P<operation_id>[0-9]+) (? P<ldap_operation>[A-Z]+)(.* \$))</pre> </div> <p>В логе собирается информация по типам операций ldap_operation. Все операции собираются, им присваивается labels и vector пишет все в БД. Метрика строится по следующим типам операций: SRCH</p>

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
48	Нагрузка на процессор на контроллерах	Метрика показывает загрузку процессора на хостах контроллера домена	FreelPA	ALD-Pro state with history data/ Total CPU usage, %; CPU	node_cpu_seconds_total Развернуть Прerequisites: 1. Используется prometheus node exporter. Стандартный способ сбора метрики с хостов основанный на стандартном node exporter. Метрика образовывается путем вычитания текущего времени использования ЦПУ и минуту назад
49	Использование памяти	Метрика показывает использование памяти на хостах контроллера домена	FreelPA	ALD-Pro state with history data/ Total memory usage, %; Memory Stack	node_memory_MemTotal_bytes ; node_memory_Buffers_bytes ; node_memory_MemFree_bytes ; node_memory_Cached_bytes Развернуть Прerequisites: 1. Используется prometheus node exporter. Стандартный способ сбора метрики с хостов основанный на стандартном node exporter. Метрика образовывается путем вычитания из максимального количества ОЗУ всей использованной памяти

	Наименование	Описание	Название подсистемы ALD Pro	Дашборд/ Название графика	Название метрики в БД и Технология сбора
50	Проверка сроков действия паролей пользователей	Метрика показывает конечную дату действия пароля пользователя и количество времени до этой даты.	FreeIPA	ALD-Pro state with history data/ Password expiration	ald_expiration_user_password Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется freeipa-exporter. 2. В freeipa-exporter.conf прописаны имена пользователей для проверки их паролей Экспортер проверяет ответ от ldap по времени окончания действия пароля пользователя

6.19.4 4. Метрики, используемые для мониторинга ПК СВ "Брест"

	Наименование метрики	Описание	Дашборд/ Название графики	Название метрики в БД и Технология сбора
1	Количество запущенных VM на выбранном кластере	Метрика показывает сколько VM находится в запущенном (running) состоянии в выбранном кластере	Brest Cluster info/ Running VMs	one_cluster_runningvms Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос всех запущенных VM на инстансе Бреста.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
2	Всего ЦПУ в кластере	Метрика показывает сколько всего ЦПУ может быть использовано в выбранном кластере	Brest Cluster info/ CPUs total	<p>one_cluster_totalcpu</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос сколько всего ЦПУ на инстансе Бреста.</p>
3	Количество использованных ЦПУ в кластере	Метрика показывает сколько всего ЦПУ используется в данный момент в выбранном кластере	Brest Cluster info/ CPUs used	<p>one_cluster_cpusage</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос сколько всего ЦПУ используется на инстансе Бреста.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
4	Всего памяти в кластере	Метрика показывает сколько всего Memory в выбранном кластере	Brest Cluster info/ Memory total	<p>one_cluster_totalmem</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос сколько всего Memory на инстансе Бреста.</p>
5	Количество используемой памяти в кластере	Метрика показывает сколько всего Memory используется в выбранном кластере	Brest Cluster info/ Memory used	<p>one_cluster_memusage</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос сколько всего Memory используется на инстансе Бреста.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
6	Суммарный статус всех сервисов на хостах выбранного кластера	Метрика показывает суммарный статус всех заранее заданных сервисов по нодам выбранного кластера	Brest Cluster info/ Services on nodes	systemd_unit_state_id Развернуть Прerequisites: 1. Используется systemd exporter Комментарий: Список сервисов: libvirtd.* postgre.* chrony.* sssd.* opennebula.*
7	Суммарный статус всех сервисов на fronts выбранного кластера	Метрика показывает суммарный статус всех заранее заданных сервисов по фронтам выбранного кластера	Brest Cluster info/ Services on fronts	systemd_unit_state_id Развернуть Прerequisites: 1. Используется systemd exporter Комментарий: Список сервисов: libvirtd.* postgre.* chrony.* sssd.* opennebula.*

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
8	Количество использованного места в датасторах	Метрика показывает общий размер датасторов выбранного кластера и сколько места использовано.	Brest Cluster info/ Datastores size	<p>one_ds_usedmb; one_ds_totalmb</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос общего количества места на датасторах и сколько использовано на инстансе Бреста.</p>
9	Распределение датасторов по кластерам	Метрика показывает распределение датасторов по выбранным кластерам	Brest Cluster info/ Datastore uses on cluster	<p>one_ds_available</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос какие датасторы представлены на кластерах инстанса Бреста.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
10	Количество используемого ЦПУ, %	Метрика показывает отношение использованных ЦПУ к всего имеющихся ЦПУ на выбранном кластере	Brest Cluster info/ CPUs usage, %	<p>one_cluster_cpuusage; one_cluster_totalcpu</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>На основе данных API вычисляется отношение использованного ЦПУ к всего имеющемуся.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
11	Количество используемой памяти в кластере, %	Метрика показывает отношение занятой памяти к всего имеющейся памяти на выбранном кластере	Brest Cluster info/ Memory usage, %	<p>one_cluster_memusage; one_cluster_totalmem</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>На основе данных API вычисляется отношение использованной памяти к всего имеющейся памяти.</p>
12	Статус front хостов в выбранной зоне	Метрика показывает какой фронт является лидером, а какие фоловерами.	Brest Management servers/ RAFT status	<p>one_zone_raft</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос статуса фронтов на инстансе Бреста.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
13	Статус проверки доступности API для данной зоны	Метрика показывает доступность API для данного инстанса Бреста	Brest Management servers/ API connection	<p>one_api_connect</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>При первом запросе к API анализируется ответ и делается вывод о доступности или не доступности API.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
14	Статус проверки доступности WEB консоли управления Брестом	Метрика показывает доступность веб консоли управления Брестом	Brest Management servers/ WEB console connection	<p>one_web_connect</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесены специальные переменные для проверки (webuser, webpasswd, webconsole) <p>С хоста, где установлен One-exporter делается попытка обратиться к URL веб консоли управления Бреста</p> <p>Комментарий:</p> <p>Пользователь должен быть создан не как core пользователь.</p> <p>Проверка делается с хоста, где установлен one-exporter.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
15	Время отклика WEB консоли управления Брестом	Метрика показывает время отклика при замере доступности веб консоли управления Брестом	Brest Management servers/ WEB console time connection	<p>one_web_connect_duration</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесены специальные переменные для проверки (webuser, webpasswd, webconsole) <p>С хоста, где установлен One-exporter делается попытка обратиться к URL веб консоли управления Бреста и время ответа замеряется</p> <p>Комментарий:</p> <p>Пользователь должен быть создан не как core пользователь.</p> <p>Проверка делается с хоста, где установлен one-exporter.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
16	Статус сервисов на выбранном front	Метрика показывает статусы определенных сервисов на выбранном фронте	Brest Management servers/ Services on front	systemd_unit_state_id Развернуть Прerequisites: 1. Используется systemd exporter Комментарий: Список сервисов: libvirt.* postgre.* chrony.* sssd.* opennebula.*
17	Статус front хоста	Метрика показывает статус выбранного фронта	Brest Management servers/ Host state	node_exporter_build_info Развернуть Прerequisites: 1. Используется node exporter
18	Использование ЦПУ на выбранном front, %	Метрика показывает распределение нагрузки на ЦПУ по процессорам на фронте	Brest Management servers/ CPUs usage, %	node_cpu_seconds_total Развернуть Прerequisites: 1. Используется node exporter
19	Использование памяти на выбранном front	Метрика показывает распределение памяти на фронте	Brest Management servers/ Memory utilization	node_memory_MemTotal_bytes; node_memory_MemFree_bytes; node_memory_Cached_bytes; node_memory_Buffers_bytes; node_memory_SReclaimable_bytes; node_memory_SwapTotal_bytes; node_memory_SwapFree_bytes Развернуть Прerequisites: 1. Используется node exporter

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
20	Поиск по логам выбранного хоста	Метрика позволяет осуществлять поиск по ключевым словам в логах или просто просматривать логи	Brest Management servers/ Поиск по логам	Таблица logs в БД ClickHouse Развернуть Метрика собирается из логов. Прerequisites: <ol style="list-style-type: none"> 1. Используется сервис vector 2. В /etc/vector/conf.d/ лежит файл с описанием нужных логов 3. Там же настроен парсинг этого лога по необходимым параметрам Комментарий: Описание парсинга логов для Бреста: https://life.astralinux.ru/x/GLS4DQ
21	Список зомби ВМок	Метрика показывает список зомби ВМ по хостам виртуализации в выбранном кластере	Brest VMs info/ List of zombie VMs	one_host_zombie Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста Через API делается запрос на наличие зомби ВМ на хостах виртуализации на инстансе Бреста.

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
22	Список ВМок со статусами.	Метрика показывает все текущие статусы ВМок в данном кластере	Brest VMs info/ List of VMs with status	<p>one_vm_state</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос статусов всех ВМ на хостах виртуализации на инстансе Бреста.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
23	Информация о VM	Метрика показывает название выбранной виртуальной машины, кто ее владелец и какая группа в Брест	Brest VMs info/ VM Info	<p>one_vm_state; one_vm_totalcpu</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос об имени VM, информации о владельце, группе</p> <p>Комментарий: Два графика без названий во вкладке VM Info</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
24	Использование ЦПУ конкретной VM	Метрика показывает использование ЦПУ выбранной виртуальной машиной	Brest VMs info/ CPU usage (API)	<p>one_vm_cpuusage; one_vm_totalcpu</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о использовании ЦПУ конкретной VM</p> <p>Комментарий: Данные предоставлены через API Бреста (не нод экспортер)</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
25	Использование памяти конкретной VM	Метрика показывает использование памяти выбранной виртуальной машиной	Brest VMs info/ Memory usage (API)	<p>one_vm_memusage; one_vm_totalmem</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о использовании памяти конкретной VM</p> <p>Комментарий: Данные предоставлены через API Бреста (не нод экспортер)</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
26	Скорость чтения/записи в секунду	Метрика показывает скорость работы с диском выбранной VM	Brest VMs info/ Disk write/read	<p>one_vm_disk_read_bytes; one_vm_disk_write_bytes</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о скорости чтения/записи на диск конкретной VM</p>
27	Среднее число операций чтения/записи в секунду	Метрика показывает число операций с диском выбранной VM	Brest VMs info/ Disk IOPS	<p>one_vm_disk_read_iops; one_vm_disk_write_iops</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о числе операций с диском конкретной VM</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
28	Сетевой трафик	Метрика показывает объем сетевого трафика у выбранной VM	Brest VMs info/ Network usage	<p>one_vm_network_rx; one_vm_network_tx</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о сетевом трафике у конкретной VM</p>
29	Скорость трафика через сетевые интерфейсы VM	Метрика показывает скорость передачи данных на выбранной VM	Brest VMs info/ Network speed	<p>one_vm_network_rx; one_vm_network_tx</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о сетевом трафике у конкретной VM</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
30	Информация по хосту виртуализации	Метрика выводит данные об IP адресе, имени хоста виртуализации и времени последней загрузки хоста.	Brest Virtualization servers	node_boot_time_seconds Развернуть Прerequisites: 1. Используется node exporter На основе метрики из node exporter собираются данные об IP адресе, имени хоста и времени его последней загрузки
31	Статус хоста	Метрика показывает текущий статус хоста виртуализации	Brest Virtualization servers/ Host state	one_host_state Развернуть Прerequisites: 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста Через API делается запрос о текущем статусе хоста виртуализации

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
32	Статус сервисов на выбранном хосте виртуализации	Метрика показывает статус всех заранее заданных сервисов по выбранному хосту виртуализации	Brest Virtualization servers/ Services on host	systemd_unit_state_id Развернуть Прerequisites: 1. Используется systemd exporter Комментарий: Список сервисов: libvirtd.* postgre.* chrony.* sssd.* opennebula.*
33	Утилизация ЦПУ на хосте	Метрика показывает распределение ЦПУ между процессорами на хосте виртуализации	Brest Virtualization servers/ CPU utilization	node_cpu_seconds_total Развернуть Прerequisites: 1. Используется node exporter На основе метрики из node exporter собираются данные о использовании ЦПУ на хосте виртуализации
34	Утилизация ЦПУ на каждую VM	Метрика показывает распределение ЦПУ между VM на хосте виртуализации	Brest Virtualization servers/ CPU utilization by VMs	libvirt_domain_info_cpu_time_seconds_total Развернуть Прerequisites: 1. Используется libvirt exporter На основе метрики из экспортера собираются данные о использовании ЦПУ для каждой VM на хосте виртуализации

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
35	Утилизация памяти	Метрика показывает распределение памяти на хосте виртуализации	Brest Virtualization servers/ Memory Basic	node_memory_MemTotal_bytes; node_memory_MemFree_bytes; node_memory_Cached_bytes; node_memory_Buffers_bytes; node_memory_SReclaimable_bytes; node_memory_SwapTotal_bytes; node_memory_SwapFree_bytes Развернуть Прerequisites: 1. Используется node exporter На основе метрики из node exporter собираются данные о использовании памяти на хосте виртуализации
36	Утилизация памяти по VM	Метрика показывает распределение памяти между VM на хосте виртуализации	Brest Virtualization servers/ Memory baloon by VMs	node_memory_MemTotal_bytes; libvirt_domain_memory_stats_actual_balloon Развернуть Прerequisites: 1. Используется node exporter 2. Используется libvirt exporter На основе метрик собираются данные о использовании памяти по всем VM на хосте виртуализации
37	Среднее число операций чтения в секунду	Метрика показывает среднее число операций чтения с диска на хосте виртуализации	Brest Virtualization servers/ Disk Read IOPS	node_disk_reads_completed_total Развернуть Прerequisites: 1. Используется node exporter На основе метрики из node exporter собираются данные о числе операций чтения с диска на хосте виртуализации

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
38	Количество запросов операций чтения в секунду от VM	Метрика показывает среднее число операций чтения с диска от каждой VM на хосте виртуализации	Brest Virtualization servers/ Read requests by VMs	libvirt_domain_block_stats_read_requests_total Развернуть Прerequisites: 1. Используется libvirt exporter На основе метрики из экспортера собираются данные об операциях чтения с диска для каждой VM на хосте виртуализации
39	Среднее число операций записи в секунду	Метрика показывает среднее число операций записи на диск на хосте виртуализации	Brest Virtualization servers/ Disk Write IOPS	node_disk_writes_completed_total Развернуть Прerequisites: 1. Используется node exporter На основе метрики из node exporter собираются данные о числе операций записи на диск на хосте виртуализации
40	Количество запросов операций записи в секунду от VM	Метрика показывает среднее число операций записи на диск от каждой VM на хосте виртуализации	Brest Virtualization servers/ Write requests by VMs	libvirt_domain_block_stats_write_requests_total Развернуть Прerequisites: 1. Используется libvirt exporter На основе метрики из экспортера собираются данные об операциях записи на диск для каждой VM на хосте виртуализации

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
41	Сетевой трафик	Метрика показывает скорость передачи сетевого трафика на хосте виртуализации	Brest Virtualization servers/ Network traffic, b/s	node_network_receive_bytes_total; node_network_transmit_bytes_total Развернуть Прerequisites: 1. Используется node exporter На основе метрики из node exporter собираются данные о скорости передачи сетевого трафика на хосте виртуализации
42	Сетевой трафик VM	Метрика показывает скорость передачи сетевого трафика по каждой VM на хосте виртуализации	Brest Virtualization servers/ Network traffic by VMs, b/s	libvirt_domain_interface_stats_receive_bytes_total; libvirt_domain_interface_stats_transmit_bytes_total Развернуть Прerequisites: 1. Используется libvirt exporter На основе метрики из экспортера собираются данные о сетевом трафике от каждой VM на хосте виртуализации

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
43	Количество пакетов в секунду	Метрика показывает скорость передачи сетевых пакетов на хосте виртуализации	Brest Virtualization servers/ Network traffic, packets/s	node_network_receive_packets_total; node_network_transmit_packets_total Развернуть Прerequisites: 1. Используется node exporter На основе метрики из node exporter собираются данные о количестве переданных пакетов в секунду на хосте виртуализации
44	Количество пакетов в секунду VM	Метрика показывает скорость передачи сетевых пакетов по каждой VM на хосте виртуализации	Brest Virtualization servers/ Network traffic by VMs, packets/s	libvirt_domain_interface_stats_receive_packets_total; libvirt_domain_interface_stats_transmit_packets_total Развернуть Прerequisites: 1. Используется libvirt exporter На основе метрики из экспортера собираются данные о количестве переданных пакетов в секунду от каждой VM на хосте виртуализации

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
45	Поиск по логам выбранного хоста	Метрика позволяет осуществлять поиск по ключевым словам в логах или просто просматривать логи	Brest Virtualization servers/ Поиск по логам	Таблица logs в БД ClickHouse Развернуть Метрика собирается из логов. Прerequisites: <ol style="list-style-type: none"> 1. Используется сервис vector 2. В /etc/vector/conf.d/ лежит файл с описанием нужных логов 3. Там же настроен парсинг этого лога по необходимым параметрам Комментарий: Описание парсинга логов для Бреста: https://life.astralinux.ru/x/GLS4DQ
46	Статус front хостов в выбранной зоне	Метрика показывает какой фронт является лидером, а какие фолловерами.	Brest summary/ RAFT status	one_zone_raft Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста Через API делается запрос статуса фронтов на инстансе Бреста.

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
47	Количество смены статуса лидера на fronts	Метрика показывает сколько раз менялся лидер у фронтов.	Brest summary/ RAFT switches	<p>one_zone_raft</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос сколько раз менялся лидер фронтов на инстансе Бреста.</p>
48	Количество активных хостов для зоны	Метрика показывает количество активных хостов виртуализации для выбранной зоны с разбивкой по кластерам.	Brest summary/ Hosts	<p>one_cluster_activehosts</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о количестве активных хостов для зоны.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
49	Количество запущенных VM для зоны	Метрика показывает количество запущенных VM для зоны с разбивкой по кластерам.	Brest summary/ Running VMs	one_cluster_runningvms Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста Через API делается запрос о количестве запущенных VM для зоны.
50	Количество запущенных VM на всех кластерах	Метрика показывает количество запущенных VM для зоны для всех кластеров.	Brest summary/ Running	one_cluster_runningvms Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста Через API делается запрос о количестве запущенных VM для зоны.

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
51	Количество выключенных VM на всех кластерах	Метрика показывает количество выключенных VM для зоны для всех кластеров.	Brest summary/ Powered off	<p>one_vms_states_count</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о количестве выключенных VM для зоны.</p>
52	Количество VM в статусе failed на всех кластерах	Метрика показывает количество VM в статусе failed для зоны для всех кластеров.	Brest summary/ Failed	<p>one_vms_states_count</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о количестве VM в статусе failed для зоны.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
53	Количество VM в статусе pending на всех кластерах	Метрика показывает количество VM в статусе pending для зоны для всех кластеров.	Brest summary/ Pending	<p>one_vms_states_count</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о количестве VM в статусе pending для зоны.</p>
54	Количество VM в статусе hold на всех кластерах	Метрика показывает количество VM в статусе hold для зоны для всех кластеров.	Brest summary/ Hold	<p>one_vms_states_count</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о количестве VM в статусе hold для зоны.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
55	Количество VM в статусе init на всех кластерах	Метрика показывает количество VM в статусе init для зоны для всех кластеров.	Brest summary/ Init	<p>one_vms_states_count</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о количестве VM в статусе init для зоны.</p>
56	Количество VM в статусе cloning на всех кластерах	Метрика показывает количество VM в статусе cloning для зоны для всех кластеров.	Brest summary/ Cloning	<p>one_vms_states_count</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о количестве VM в статусе cloning для зоны.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
57	Количество VM в статусе clone fail на всех кластерах	Метрика показывает количество VM в статусе clone fail для зоны для всех кластеров.	Brest summary/ Clone fail	<p>one_vms_states_count</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о количестве VM в статусе clone fail для зоны.</p>
58	Количество ЦПУ по кластерам	Метрика показывает общее количество ЦПУ для зоны с разбивкой по кластерам.	Brest summary/ CPU total	<p>one_cluster_totalcpu</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о общем количестве ЦПУ для зоны.</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
59	Количество использованных ЦПУ по кластерам	Метрика показывает количество использованных ЦПУ для зоны с разбивкой по кластерам.	Brest summary/ CPU used	one_cluster_cpuusage Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста Через API делается запрос о количестве использованных ЦПУ для зоны.
60	Количество памяти по кластерам	Метрика показывает общее количество памяти для зоны с разбивкой по кластерам.	Brest summary/ Memory total	one_cluster_totalmem Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста Через API делается запрос о общем количестве памяти для зоны.

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
61	Количество используемой памяти по кластерам	Метрика показывает количество использованной памяти для зоны с разбивкой по кластерам.	Brest summary/ Memory used	one_cluster_memusage Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста Через API делается запрос о количестве использованной памяти для зоны.
62	Amount of used CPUs	Метрика показывает отношение использованных ЦПУ к общему количеству ЦПУ для зоны с разбивкой по кластерам.	Brest summary/ CPUs usage, %	one_cluster_cpuusage; one_cluster_totalcpu Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста Через API делается запрос о количестве использованных ЦПУ для зоны.

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
63	Отношение количества использованной памяти к общему количеству памяти	Метрика показывает отношение использованной памяти к общему количеству памяти для зоны с разбивкой по кластерам.	Brest summary/ Memory usage, %	one_cluster_memusage; one_cluster_totalmem Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста Через API делается запрос о количестве использованной памяти для зоны.
64	Размер используемых датасторов	Метрика показывает размер и количество занятого места на датасторах для зоны.	Brest summary/ Datastores size	one_ds_usedmb Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста Через API делается запрос о размерах датасторов для зоны.

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
65	Разбивка датасторов по кластерам	Метрика показывает распределение датасторов по кластерам для зоны.	Brest summary/ Datastore uses on cluster	<p>one_ds_available</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется one exporter 2. Создан технический пользователь в Брест как core пользователь 3. Пользователь прописан в one-exporter.conf 4. Пароль для пользователя должен быть зашифрован Base64 5. В конфиг внесен адрес API Бреста <p>Через API делается запрос о использовании датасторов на кластерах для зоны.</p>
66	Утилизация ЦПУ по хостам в группе	Метрика показывает использование ЦПУ по хостам внутри группы	Brest summary/ CPU utilization by hosts in group	<p>node_cpu_seconds_total</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется node exporter <p>На основе метрики из node exporter собираются данные о распределении ЦПУ по хостам внутри группы</p> <p>Комментарий: Группа - это виртуальное разделение хостов виртуализации, заданное в параметрах дашборда</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
67	Использование памяти на хостах виртуализации в группе	Метрика показывает использование памяти по хостам внутри группы	Brest summary/ Memory Used	node_memory_MemTotal_bytes; node_memory_MemFree_bytes; node_memory_Cached_bytes; node_memory_Buffers_bytes; node_memory_SReclaimable_bytes Развернуть Прerequisites: 1. Используется node exporter На основе метрики из node exporter собираются данные о распределении памяти по хостам внутри группы
68	Скорость входящего сетевого трафика	Метрика показывает скорость входящего сетевого трафика по хостам внутри группы	Brest summary/ Network Receive Traffic	node_network_receive_bytes_total Развернуть Прerequisites: 1. Используется node exporter На основе метрики из node exporter собираются данные о входящем сетевом трафике по хостам внутри группы
69	Скорость исходящего сетевого трафика	Метрика показывает скорость исходящего сетевого трафика по хостам внутри группы	Brest summary/ Network Transmit Traffic	node_network_transmit_bytes_total Развернуть Прerequisites: 1. Используется node exporter На основе метрики из node exporter собираются данные о исходящем сетевом трафике по хостам внутри группы

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
70	Использованное дисковое пространство по хостам	Метрика показывает использованное дисковое пространство на хостах внутри группы	Brest summary/ Disk Space Used Basic	node_filesystem_size_bytes; node_filesystem_avail_bytes Развернуть Пререквизиты: 1. Используется node exporter На основе метрики из node exporter собираются данные о использовании дискового пространства на хостах внутри группы

6.19.5 5. Метрики, используемые для мониторинга RuPost

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
1	Топ 10 пользователей по использованию места на диске для писем	Метрика показывает список пользователей, которые больше всего используют места на диске для хранения своих писем	RuPost PostgreSQL custom metrics/ Top 10 users by quota bytes	rupost_top10_users_by_quota_bytes Развернуть Пререквизиты: 1. Используется sql-exporter 2. Создан пользователь с необходимыми правами к БД 3. Пользователь прописан в строке подключения в файле sql-exporter.yml Метрика получается через запрос в БД

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
2	Топ 10 пользователей по количеству писем в ящике	Метрика показывает список пользователей, у которых больше всего писем	RuPost PostgreSQL custom metrics/ Top 10 users by messages	rupost_top10_users_by_messages Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется sql-exporter 2. Создан пользователь с необходимыми правами к БД 3. Пользователь прописан в строке подключения в файле sql-exporter.yml Метрика получается через запрос в БД
3	Postgresql ratio	Postgresql ratio	RuPost PostgreSQL custom metrics/ Postgresql ratio	rupost_postgresql_ratio Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется sql-exporter 2. Создан пользователь с необходимыми правами к БД 3. Пользователь прописан в строке подключения в файле sql-exporter.yml Метрика получается через запрос в БД

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
4	Unused indexes	Unused indexes	RuPost PostgreSQL custom metrics/ Unused indexes	rupost_postgresql_unused_indexes Развернуть Пререквизиты: 1. Используется sql-exporter 2. Создан пользователь с необходимыми правами к БД 3. Пользователь прописан в строке подключения в файле sql-exporter.yml Метрика получается через запрос в БД
5	Activity of users in DB	Activity of users in DB	RuPost PostgreSQL custom metrics/ Activity of users in DB	rupost_postgresql_activity_users Развернуть Пререквизиты: 1. Используется sql-exporter 2. Создан пользователь с необходимыми правами к БД 3. Пользователь прописан в строке подключения в файле sql-exporter.yml Метрика получается через запрос в БД

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
6	Xact_rollback	Xact_rollback	RuPost PostgreSQL custom metrics/ Xact_rollback	rupost_postgresql_xact_rollback Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется sql-exporter 2. Создан пользователь с необходимыми правами к БД 3. Пользователь прописан в строке подключения в файле sql-exporter.yml Метрика получается через запрос в БД
7	Количество активных подключений к базе	Метрика показывает, сколько в данный момент активных подключений к БД	RuPost PostgreSQL custom metrics/ Active connections to DB	rupost_postgresql_active_connections Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется sql-exporter 2. Создан пользователь с необходимыми правами к БД 3. Пользователь прописан в строке подключения в файле sql-exporter.yml Метрика получается через запрос в БД

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
8	Использование ЦП выбранными службами , %	Метрика показывает потребление ЦПУ выбранными службами	RuPost metrics (row "General metrics")/ Использование ЦП процессами служб, %	systemd_unit_cpu_usage; node_cpu_seconds_total Развернуть Прerequisites: 1. Используется systemd- exporter 2. В конфигурационном файле экспортера указан список служб 3. Используется node- exporter Метрика создается отношением между потребленными ресурсами служб и общим потреблением ЦПУ Комментарий: Метрика отображается для выбранного хоста
9	Использование памяти выбранными службами	Метрика показывает потребление памяти выбранными службами и доступное количество памяти	RuPost metrics (row "General metrics")/ Использование памяти процессами служб	systemd_unit_memory_usage ; node_memory_MemTotal_byt es Развернуть Прerequisites: 1. Используется systemd- exporter 2. В конфигурационном файле экспортера указан список служб 3. Используется node- exporter Метрика создается на основе данных экспортеров Комментарий: Метрика отображается для выбранного хоста

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
10	Количество использованного места по выбранным файловым системам в %	Метрика выводит процент использованного места на файловых системах выбранного хоста	RuPost metrics (row "General metrics")/ Disk Space Used %	fs_total_bytes; fs_avail_bytes Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется rpost-exporter 2. В конфигурационном файле экспортера прописан путь до директории с очередями 3. В конфигурационном файле экспортера задан список очередей Метрика создается подсчетом файлов в выбранных каталогах, где хранятся данные по очередям до их отправки Комментарий: Метрика отображается для выбранного хоста
11	Статус службы dovecot	Метрика показывает состояние службы dovecot	RuPost metrics (row "All node metrics")/ Dovecot	systemd_unit_state_id Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется systemd-exporter 2. В конфигурационном файле экспортера указан список служб Метрика создается на основе данных экспортера Комментарий: График отображает все доступные сервера RuPost независимо от фильтра

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
12	Статус службы haproxy	Метрика показывает состояние службы haproxy	RuPost metrics (row "All node metrics")/ Haproxy	systemd_unit_state_id Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется systemd-exporter 2. В конфигурационном файле экспортера указан список служб Метрика создается на основе данных экспортера Комментарий: График отображает все доступные сервера RuPost независимо от фильтра
13	Статус службы sogo	Метрика показывает состояние службы sogo	RuPost metrics (row "All node metrics")/ Sogo	systemd_unit_state_id Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется systemd-exporter 2. В конфигурационном файле экспортера указан список служб Метрика создается на основе данных экспортера Комментарий: График отображает все доступные сервера RuPost независимо от фильтра

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
14	Статус службы nginx	Метрика показывает состояние службы nginx	RuPost metrics (row "All node metrics")/ Nginx	systemd_unit_state_id Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется systemd-exporter 2. В конфигурационном файле экспортера указан список служб Метрика создается на основе данных экспортера Комментарий: График отображает все доступные сервера RuPost независимо от фильтра
15	Статус службы postfix	Метрика показывает состояние службы postfix	RuPost metrics (row "All node metrics")/ Postfix	systemd_unit_state_id Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется systemd-exporter 2. В конфигурационном файле экспортера указан список служб Метрика создается на основе данных экспортера Комментарий: График отображает все доступные сервера RuPost независимо от фильтра

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
16	Статус службы rupost	Метрика показывает состояние службы rupost	RuPost metrics (row "All node metrics")/ Rupost	systemd_unit_state_id Развернуть Прerequisites: <ol style="list-style-type: none"> Используется systemd-exporter В конфигурационном файле экспортера указан список служб Метрика создается на основе данных экспортера Комментарий: График отображает все доступные сервера RuPost независимо от фильтра
17	Статус синхронизации времени на почтовом сервере	Метрика показывает статус синхронизации времени на почтовом сервере	RuPost metrics (row "All node metrics")/ NTP Status	ntp_check Развернуть Прerequisites: <ol style="list-style-type: none"> Используется rupost-exporter Метрика создается обработкой ответа команды chronyc traking Комментарий: График отображает все доступные сервера RuPost независимо от фильтра

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
18	Количество активных пользователей лицензии RuPost	Метрика показывает сколько пользователей уже используют текущую лицензию	RuPost metrics (row "All node metrics")/ People in license / total people	license_people_using Развернуть Пререквизиты: 1. Используется rupost-exporter Метрика создается обработкой ответа команды chronyc tracking Комментарий: График отображает все доступные сервера RuPost независимо от фильтра
19	Общее количество доступных пользователей на лицензии RuPost	Метрика выводит общее количество доступных пользователей на текущей лицензии RuPost	RuPost metrics (row "All node metrics")/ People in license / total people	license_people_total Развернуть Пререквизиты: 1. Используется rupost-exporter Метрика создается обработкой ответа команды rupost about Комментарий: График отображает все доступные сервера RuPost независимо от фильтра

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
20	Время действия лицензии RuPost	Метрика показывает сколько времени осталось до окончания срока действия лицензии RuPost	RuPost metrics (row "All node metrics")/ Rupost license expiration time	license_expire_date Развернуть Прerequisites: 1. Используется rupost-exporter Метрика создается обработкой ответа команды rupost about Комментарий: График отображает все доступные сервера RuPost независимо от фильтра Метрика не работает на RuPost начиная с версии 2.7.1
21	Статус текущей лицензии RuPost	Метрика выводит статус текущей лицензии сервера RuPost	RuPost metrics (row "All node metrics")/ Status of license	license_status Развернуть Прerequisites: 1. Используется rupost-exporter Метрика создается обработкой ответа команды rupost about Комментарий: График отображает все доступные сервера RuPost независимо от фильтра

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
22	Срок действия SSL сертификатов Рупост	Метрика показывает сколько времени осталось до окончания срока действия SSL сертификатов RuPost	RuPost metrics (row "All node metrics")/ Rupost SSL certificate expiration time	cert_expire_date Развернуть Прerequisites: 1. Используется rupost-exporter Метрика создается обработкой ответа команды rupost about Комментарий: График отображает все доступные сервера RuPost независимо от фильтра
23	Количество активных пользователей по каждой ноде по времени	Метрика выводит количество пользователей, распределенных по всем нодам инстанса RuPost	RuPost metrics (row "All node metrics")/ Number of active people on node by time	users_on_node Развернуть Прerequisites: 1. Используется rupost-exporter Метрика создается обработкой ответа команды rupost about Комментарий: График отображает все доступные сервера RuPost независимо от фильтра Метрика убрана из экспортера с версии 1.0.5

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
24	Длина выбранных очередей	Метрика показывает длину очередей на всех хостах по времени	RuPost metrics (row "All hosts metrics")/ Queue length by time	<p>queue_length</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется ruPost-exporter 2. В конфигурационном файле экспортера прописан путь до директории с очередями 3. В конфигурационном файле экспортера задан список очередей <p>Метрика создается подсчетом файлов в выбранных каталогах, где хранятся данные по очередям до их отправки</p> <p>Комментарий: График отображает все доступные сервера RuPost независимо от фильтра</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
25	Количество haproxy backend сессий	Метрика выводит количество текущих haproxy сессий в зависимости от backend	RuPost metrics (row "HAProxy metrics")/ Back - Number of sessions	haproxy_backend_sessions_total; haproxy_backend_current_sessions Развернуть Прerequisites: <ol style="list-style-type: none"> Используется встроенный экспортер компонента haproxy Настройки haproxy в конфиг файле выполнены в соответствии с инструкцией Метрика строится на основе данных от компонента haproxy Комментарий: Метрика отображается для выбранного хоста
26	Количество ошибок на HAProxy	Метрика показывает количество ошибок по HTTP кодам	RuPost metrics (row "HAProxy metrics")/ Server - HTTP responses code	haproxy_server_http_responses_total Развернуть Прerequisites: <ol style="list-style-type: none"> Используется встроенный экспортер компонента haproxy Настройки haproxy в конфиг файле выполнены в соответствии с инструкцией Метрика строится на основе данных от компонента haproxy Комментарий: Метрика отображается для выбранного хоста

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
27	Количество доставленных сообщений компонентом dovecot	Метрика показывает, сколько сообщений доставляет компонент dovecot по каждому хосту	RuPost metrics (row "Dovecot metrics")/ Dovecot MAIL delivery counts by 1m	dovecot_mail_delivery_total Развернуть Пререквизиты: 1. Используется встроенный экспортер компонента dovecot 2. Настройки dovecot в конфиг файле выполнены в соответствии с инструкцией Метрика строится на основе данных от компонента dovecot Комментарий: График отображает все доступные сервера RuPost независимо от фильтра
28	Количество SMTP команд в минуту	Метрика показывает, сколько SMTP запросов выполняет компонент dovecot в минуту по каждому хосту	RuPost metrics (row "Dovecot metrics")/ Dovecot SMTP-command counts	dovecot_smtp_command_count Развернуть Пререквизиты: 1. Используется встроенный экспортер компонента dovecot 2. Настройки dovecot в конфиг файле выполнены в соответствии с инструкцией Метрика строится на основе данных от компонента dovecot Комментарий: График отображает все доступные сервера RuPost независимо от фильтра

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
29	Количество ошибок SMTP протокола	Метрика показывает, сколько было ошибок при SMTP запросах в зависимости от конкретных кодов ответа	RuPost metrics (row "Dovecot metrics")/ SMTP Failures	dovecot_smtp_command_total Развернуть Пререквизиты: 1. Используется встроенный экспортер компонента dovecot 2. Настройки dovecot в конфиг файле выполнены в соответствии с инструкцией Метрика строится на основе данных от компонента dovecot Комментарий: График отображает все доступные сервера RuPost независимо от фильтра
30	Количество ошибок аутентификации в минуту	Метрика показывает количество ошибок аутентификации через компонент dovecot в минуту по каждому хосту	RuPost metrics (row "All hosts metrics")/ Dovecot Failures	dovecot_auth_failures_total Развернуть Пререквизиты: 1. Используется встроенный экспортер компонента dovecot 2. Настройки dovecot в конфиг файле выполнены в соответствии с инструкцией Метрика строится на основе данных от компонента dovecot Комментарий: График отображает все доступные сервера RuPost независимо от фильтра

6.19.6 6. Метрики, используемые для мониторинга RuBackup

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
1	Использование ЦП выбранным и службами, %	Метрика показывает потребление ЦПУ выбранными службами	RuBackup metrics/ Использование ЦП процессами служб, %	systemd_unit_cpu_usage; node_cpu_seconds_total Развернуть Прerequisites: 1. Используется systemd-exporter 2. В конфигурационном файле экспортера указан список служб 3. Используется node-exporter Метрика создается отношением между потребленными ресурсами служб и общим потреблением ЦПУ
2	Использование памяти выбранным и службами	Метрика показывает потребление памяти выбранными службами и доступное количество памяти	RuBackup metrics/ Использование памяти процессами служб	systemd_unit_memory_usage; node_memory_MemTotal_bytes Развернуть Прerequisites: 1. Используется systemd-exporter 2. В конфигурационном файле экспортера указан список служб 3. Используется node-exporter Метрика создается на основе данных экспортеров

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
3	Статус службы rubackup_server	Метрика показывает состояние службы rubackup_server	RuBackup metrics/ Status of rubackup_server	systemd_unit_state_id Развернуть Прerequisites: 1. Используется systemd-exporter 2. В конфигурационном файле экспортера указан список служб Метрика создается на основе данных экспортера
4	Статус службы rubackup_client	Метрика показывает состояние службы rubackup_client	RuBackup metrics/ Status of rubackup_client	systemd_unit_state_id Развернуть Прerequisites: 1. Используется systemd-exporter 2. В конфигурационном файле экспортера указан список служб Метрика создается на основе данных экспортера
5	Статус службы postgresql@11-main	Метрика показывает состояние службы postgresql@11-main	RuBackup metrics/ Status of postgresql@11-main service	systemd_unit_state_id Развернуть Прerequisites: 1. Используется systemd-exporter 2. В конфигурационном файле экспортера указан список служб Метрика создается на основе данных экспортера

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
6	Количество бэкапных тасков в статусе Done	Метрика выводит количество бэкапных тасков в статусе Done за выбранный период времени	RuBackup metrics/ Done task	<p>rubackup_tasks_with_status</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется rubackup-exporter 2. Установлен API RuBackup модуль 3. В конфигурационном файле экспортера указаны необходимые данные для подключения к API <p>Метрика создается на основе данных экспортера</p>
7	Количество бэкапных тасков в статусе Broken	Метрика выводит количество бэкапных тасков в статусе Broken за выбранный период времени	RuBackup metrics/ Broken task	<p>rubackup_tasks_with_status</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется rubackup-exporter 2. Установлен API RuBackup модуль 3. В конфигурационном файле экспортера указаны необходимые данные для подключения к API <p>Метрика создается на основе данных экспортера</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
8	Количество бэкапных тасков в статусе Suspended	Метрика выводит количество бэкапных тасков в статусе Suspended за выбранный период времени	RuBackup metrics/ Suspended task	<p>rubackup_tasks_with_status</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется rubackup-exporter 2. Установлен API RuBackup модуль 3. В конфигурационном файле экспортера указаны необходимые данные для подключения к API <p>Метрика создается на основе данных экспортера</p>
9	Количество бэкапных тасков в статусе Error	Метрика выводит количество бэкапных тасков в статусе Error за выбранный период времени	RuBackup metrics/ Error task	<p>rubackup_tasks_with_status</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется rubackup-exporter 2. Установлен API RuBackup модуль 3. В конфигурационном файле экспортера указаны необходимые данные для подключения к API <p>Метрика создается на основе данных экспортера</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
10	Количество бэкапных тасков в статусе On pause	Метрика выводит количество бэкапных тасков в статусе On pause за выбранный период времени	RuBackup metrics/ On pause task	<p>rubackup_tasks_with_status</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется rubackup-exporter 2. Установлен API RuBackup модуль 3. В конфигурационном файле экспортера указаны необходимые данные для подключения к API <p>Метрика создается на основе данных экспортера</p>
11	Количество бэкапных тасков в статусе Killed	Метрика выводит количество бэкапных тасков в статусе Killed за выбранный период времени	RuBackup metrics/ Killed task	<p>rubackup_tasks_with_status</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется rubackup-exporter 2. Установлен API RuBackup модуль 3. В конфигурационном файле экспортера указаны необходимые данные для подключения к API <p>Метрика создается на основе данных экспортера</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
12	Список бакапных тасков	Метрика выводит список бакапных тасков за выбранный период с возможностью фильтрации	RuBackup metrics/ Backup tasks by last status	<p>rubackup_tasks_with_status</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется rubackup-exporter 2. Установлен API RuBackup модуль 3. В конфигурационном файле экспортера указаны необходимые данные для подключения к API <p>Метрика создается на основе данных экспортера</p>

6.19.7 7. Метрики, используемые для мониторинга Termidesk

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
1	Количество доменов аутентификации	Метрика показывает сколько доменов аутентификации и подключено в системе	Termidesk metrics (row "General metrics")/ Auth domain count	<p>count_auth_domains</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) <p>Метрика создается обращением к API по пути /api/auth/{version}/authenticators</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
2	Количество активных подключений	Метрика показывает количество активных подключений к системе	Termidesk metrics (row "General metrics")/ Active sessions	count_active_sessions Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисCOVER (переменная discover_api) Метрика создается обращением к API по пути /api/webui/{version}/spsessions

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
3	Количество пользователей в разрезе доменов аутентификации	Метрика показывает сколько пользователей создано в каждом домене аутентификации	Termidesk metrics (row "General metrics")/ Users in auth domain	count_users_in_auth_domain Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисCOVER (переменная discover_api) Метрика создается обращением к API по пути /api/webui/{version}/authenticators

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
4	Количество конкурентных соединений	Метрика показывает сколько конкурентных соединений используется (параметр лицензирования)	Termidesk metrics (row "General metrics")/ Competitive connections	<p>count_competitive_connections; free_competitive_connections</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисCOVER (переменная discover_api) <p>Метрика создается обращением к API по пути /api/webui/{version}/dashboard</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
5	Общий статус компонента VDI	Метрика показывает общий статус компонента VDI (HealthCheck)	Termidesk metrics (row "Health Checks")/ Common VDI HealthCheck	health_vdi_common Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан токен (параметр health_token) Метрика создается обращением к API по пути / api/health/
6	Статус Celery в компоненте VDI	Метрика показывает статус службы Celery в рамках компонента VDI (HealthCheck)	Termidesk metrics (row "Health Checks")/ Celery VDI HealthCheck	health_vdi_celery Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан токен (параметр health_token) Метрика создается обращением к API по пути / api/health/

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
7	Статус Database в компоненте VDI	Метрика показывает статус службы Database в рамках компонента VDI (HealthCheck)	Termidesk metrics (row "Health Checks")/ Database VDI HealthCheck	health_vdi_db Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан токен (параметр health_token) Метрика создается обращением к API по пути / api/health/
8	Статус Message broker в компоненте VDI	Метрика показывает статус службы Message broker в рамках компонента VDI (HealthCheck)	Termidesk metrics (row "Health Checks")/ Message broker VDI HealthCheck	health_vdi_messages_broker Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан токен (параметр health_token) Метрика создается обращением к API по пути / api/health/

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
9	Общий статус компонента Taskman	Метрика показывает общий статус компонента Taskman (HealthCheck)	Termidesk metrics (row "Health Checks")/ Common Taskman HealthCheck	health_taskman_common Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан токен (параметр health_token) Метрика создается обращением к API по пути / api/health/
10	Статус Database в компоненте Taskman	Метрика показывает статус службы Database в рамках компонента Taskman (HealthCheck)	Termidesk metrics (row "Health Checks")/ Database Taskman HealthCheck	health_taskman_db Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан токен (параметр health_token) Метрика создается обращением к API по пути / api/health/

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
11	Общий статус компонента WSPроxy	Метрика показывает общий статус компонента WSPроxy (HealthCheck)	Termidesk metrics (row "Health Checks")/ Common WSPроxy HealthCheck	health_wsproxy_common Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан токен (параметр health_token) Метрика создается обращением к API по пути / api/health/
12	Статус Websockify в компоненте WSPроxy	Метрика показывает статус службы Celery в рамках Websockify WSPроxy (HealthCheck)	Termidesk metrics (row "Health Checks")/ Websockify WSPроxy HealthCheck	health_wsproxy_websockify Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан токен (параметр health_token) Метрика создается обращением к API по пути / api/health/

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
13	Статус проверки порта web console	Метрика показывает статус проверки порта работы web console Termidesk (HealthCheck)	Termidesk metrics (row "Health Checks")/ Web service ping	web_service_ping Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан токен (параметр health_token) Метрика создается обращением к API по пути / api/health/
14	Статус проверки health port для компонента Taskman	Метрика показывает статус порта для проверки HealthCheck компонента Taskman	Termidesk metrics (row "Health Checks")/ Taskman health port ping	taskman_health_port_ping Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан токен (параметр health_token) 4. В конфигурационном файле указан порт для проверки (параметр taskman_port) Метрика создается обращением к API по пути / api/health/

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
15	Статус проверки health port для компонента WSPроxy	Метрика показывает статус порта для проверки HealthCheck компонента WSPроxy	Termidesk metrics (row "Health Checks")/ WSPроxy health port ping	wsproxу_health_port_ping Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан токен (параметр health_token) 4. В конфигурационном файле указан порт для проверки (параметр wsproxу_port) Метрика создается обращением к API по пути /api/health/

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
16	Количество назначенных рабочих мест со статусом "Действительный"	Метрика показывает сколько рабочих назначенных мест имеет статус "Действительный"	Termidesk metrics (row "ServicesPools") Статусы назначенных рабочих мест пользователя / "Действительный" статус рабочего места	servicespool_workplaces_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути / api/webui/{version}/ servicespools/{uuid}/ services

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
17	Количество назначенных рабочих мест со статусом "Подготовка"	Метрика показывает сколько рабочих назначенных мест имеет статус "Подготовка"	Termidesk metrics (row "ServicesPools") Статусы назначенных рабочих мест пользователя / "Подготовка" статус рабочего места	servicespool_workplaces_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисCOVER (переменная discover_api) Метрика создается обращением к API по пути / api/webui/{version}/ servicespools/{uuid}/ services

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
18	Количество назначенных рабочих мест со статусом "Отменяется"	Метрика показывает сколько рабочих назначенных мест имеет статус "Отменяется"	Termidesk metrics (row "ServicesPools") Статусы назначенных рабочих мест пользователя / "Отменяется" статус рабочего места	servicespool_workplaces_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути / api/webui/{version}/ servicespools/{uuid}/ services

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
19	Количество назначенных рабочих мест со статусом "Отменено"	Метрика показывает сколько рабочих назначенных мест имеет статус "Отменено"	Termidesk metrics (row "ServicesPools") Статусы назначенных рабочих мест пользователя / "Отменено" статус рабочего места	servicespool_workplaces_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковой (переменная discover_api) Метрика создается обращением к API по пути / api/webui/{version}/ servicespools/{uuid}/ services

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
20	Количество назначенных рабочих мест со статусом "Удаление"	Метрика показывает сколько рабочих назначенных мест имеет статус "Удаление"	Termidesk metrics (row "ServicesPools") Статусы назначенных рабочих мест пользователя / "Удаление" статус рабочего места	servicespool_workplaces_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути / api/webui/{version}/ servicespools/{uuid}/ services

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
21	Количество назначенных рабочих мест со статусом "Удаляется"	Метрика показывает сколько рабочих назначенных мест имеет статус "Удаляется"	Termidesk metrics (row "ServicesPools") Статусы назначенных рабочих мест пользователя / "Удаляется" статус рабочего места	servicespool_workplaces_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковой (переменная discover_api) Метрика создается обращением к API по пути / api/webui/{version}/ servicespools/{uuid}/ services

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
22	Количество назначенных рабочих мест со статусом "Удален"	Метрика показывает сколько рабочих назначенных мест имеет статус "Удален"	Termidesk metrics (row "ServicesPools") Статусы назначенных рабочих мест пользователя / "Удален" статус рабочего места	servicespool_workplaces_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути / api/webui/{version}/ servicespools/{uuid}/ services

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
23	Количество назначенных рабочих мест со статусом "Ошибка"	Метрика показывает сколько рабочих назначенных мест имеет статус "Ошибка"	Termidesk metrics (row "ServicesPools") Статусы назначенных рабочих мест пользователя / "Ошибка" статус рабочего места	servicespool_workplaces_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути / api/webui/{version}/ servicespools/{uuid}/ services

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
24	Количество кэшированных рабочих мест со статусом "Действительный"	Метрика показывает сколько рабочих кэшированных мест имеет статус "Действительный"	Termidesk metrics (row "ServicesPools") Статусы кэшированных рабочих мест пользователя/ "Действительный" статус рабочего места	servicespool_workplaces_caches_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковой (переменная discover_api) Метрика создается обращением к API по пути api/webui/{version}/servicespools/{uuid}/cache

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
25	Количество кэшированных рабочих мест со статусом "Подготовка"	Метрика показывает сколько рабочих кэшированных мест имеет статус "Подготовка"	Termidesk metrics (row "ServicesPools") Статусы кэшированных рабочих мест пользователя / "Подготовка" статус рабочего места	servicespool_workplaces_caches_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути api/webui/{version}/servicespools/{uuid}/cache

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
26	Количество кэшированных рабочих мест со статусом "Отменяется"	Метрика показывает сколько рабочих кэшированных мест имеет статус "Отменяется"	Termidesk metrics (row "ServicesPools") Статусы кэшированных рабочих мест пользователя/ "Отменяется" статус рабочего места	servicespool_workplaces_caches_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути api/webui/{version}/servicespools/{uuid}/cache

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
27	Количество кэшированных рабочих мест со статусом "Отменено"	Метрика показывает сколько рабочих кэшированных мест имеет статус "Отменено"	Termidesk metrics (row "ServicesPools") Статусы кэшированных рабочих мест пользователя / "Отменено" статус рабочего места	servicespool_workplaces_caches_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути api/webui/{version}/servicespools/{uuid}/cache

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
28	Количество кэшированных рабочих мест со статусом "Удаление"	Метрика показывает сколько рабочих кэшированных мест имеет статус "Удаление"	Termidesk metrics (row "ServicesPools") Статусы кэшированных рабочих мест пользователя / "Удаление" статус рабочего места	servicespool_workplaces_caches_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути api/webui/{version}/servicespools/{uuid}/cache

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
29	Количество кэшированных рабочих мест со статусом "Удаляется"	Метрика показывает сколько рабочих кэшированных мест имеет статус "Удаляется"	Termidesk metrics (row "ServicesPools") Статусы кэшированных рабочих мест пользователя/ "Удаляется" статус рабочего мест "Ошибка" статус рабочего места	servicespool_workplaces_caches_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковой (переменная discover_api) Метрика создается обращением к API по пути api/webui/{version}/servicespools/{uuid}/cache

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
30	Количество кэшированных рабочих мест со статусом "Удален"	Метрика показывает сколько рабочих кэшированных мест имеет статус "Удален"	Termidesk metrics (row "ServicesPools") Статусы кэшированных рабочих мест пользователя / "Удален" статус рабочего места	servicespool_workplaces_caches_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути api/webui/{version}/servicespools/{uuid}/cache

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
31	Количество кэшированных рабочих мест со статусом "Ошибка"	Метрика показывает сколько рабочих кэшированных мест имеет статус "Ошибка"	Termidesk metrics (row "ServicesPools") Статусы кэшированных рабочих мест пользователя / "Ошибка" статус рабочего места	servicespool_workplaces_caches_status Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути api/webui/{version}/servicespools/{uuid}/cache

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
32	Количество рабочих мест	Метрика показывает количество рабочих мест с разбивкой по фондам	Termidesk metrics (row "ServicesPools")/ User services count	count_servicespools_user_services Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковой (переменная discover_api) Метрика создается обращением к API по пути /api/webui/{version}/servicespools

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
33	Статус фондов	Метрика показывает текущий статус фондов	Termidesk metrics (row "ServicesPools")/ Servicespools statuses	<p>servicespools_statuses</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковой (переменная discover_api) <p>Метрика создается обращением к API по пути /api/webui/{version}/servicespools</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
34	Количество фондов	Метрика показывает текущее количество фондов	Termidesk metrics (row "ServicesPools")/ Servicespools count	count_servicespools Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути /api/webui/{version}/servicespools

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
35	Количество темплейтов провайдеров	Метрика отображает количество темплейтов на каждом провайдере	Termidesk metrics (row "Providers")/ Providers templates count	count_providers_templates Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковери (переменная discover_api) Метрика создается обращением к API по пути /api/webui/{version}/providers

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
36	Статус провайдеров	Метрика показывает в каком статусе находятся провайдеры	Termidesk metrics (row "Providers")/ Providers status	<p>providers_statuses</p> <p>Развернуть</p> <p>Прerequisites:</p> <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисковой (переменная discover_api) <p>Метрика создается обращением к API по пути /api/webui/{version}/providers</p>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
37	Количество провайдеров	Метрика показывает сколько провайдеров создано в системе	Termidesk metrics (row "Providers")/ Providers count	count_providers Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется termidesk-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. В конфигурационном файле указан пользователь для доступа к API (переменная api_user) 4. В конфигурационном файле указан пароль для доступа к API (переменная auth_name) 5. В конфигурационном файле указан путь к дисCOVER (переменная discover_api) Метрика создается обращением к API по пути /api/webui/{version}/providers

6.19.8 8. Метрики, используемые для мониторинга Billmanager

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
1	Статус обработчиков	Метрика показывает статус всех обработчиков для приложения	BillManager Main/ Статус обработчиков	billm_processing_status Развернуть Пререквизиты: <ol style="list-style-type: none"> 1. Используется billm-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. Создан пользователь для мониторинга 4. В конфигурационном файле перечислены провайдеры Метрика создается обращением к API по пути <code>func=processing.edit&sok=ok</code>
2	Количество проблем	Метрика показывает наличие проблем при использовании приложения	BillManager Main/ Количество проблем	billm_problems_count Развернуть Пререквизиты: <ol style="list-style-type: none"> 1. Используется billm-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. Создан пользователь для мониторинга Метрика создается обращением к API по пути <code>func=problems</code>

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
3	Проблемы по типам	Метрика выводит список существующих проблем по их типам	BillManager Main/ Проблемы по типам	billm_problems_count_by_key Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется billm-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. Создан пользователь для мониторинга Метрика создается обращением к API по пути func=problems
4	Текущее количество запущенных операций	Метрика выводит число операций пользователей, запущенных в приложении	BillManager Main/ Количество запущенных операций	billm_running_operation_count Развернуть Прerequisites: <ol style="list-style-type: none"> 1. Используется billm-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. Создан пользователь для мониторинга Метрика создается обращением к API по пути func=runningoperation&sok=ok

	Наименование метрики	Описание	Дашборд/ Название графика	Название метрики в БД и Технология сбора
5	Операции, закончившиеся ошибкой	Метрика выводит список операций, которые закончились ошибкой	BillManager Main/ Операции с ошибкой	billm_error_running_operation Развернуть Пререквизиты: <ol style="list-style-type: none"> 1. Используется billm-exporter 2. В конфигурационном файле указан путь до API (переменная path) 3. Создан пользователь для мониторинга Метрика создается обращением к API по пути func=runningoperation&sok=ok

7 Пользовательский интерфейс

7.1 Пользовательское окно "Проблемы"

7.1.1 Общая информация

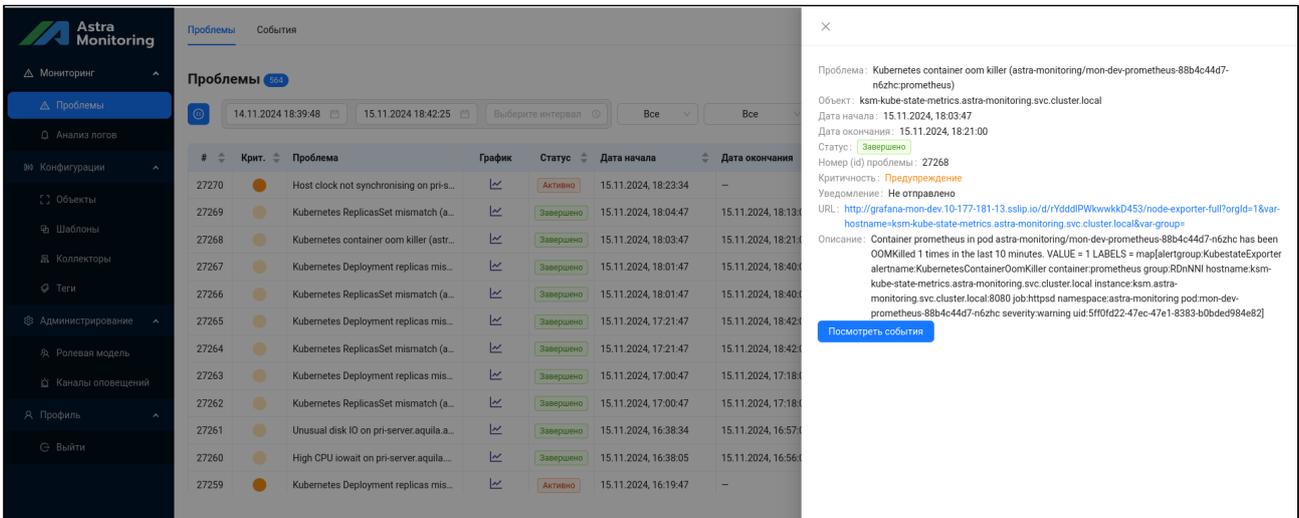
Окно "Проблемы" содержит таблицу, которая отображает список проблем и соответствующие данные. Таблица включает интерактивные элементы для сортировки записей.

Над таблицей расположены элементы управления, которые расширяют возможности поиска и фильтрации, отображают количество найденных проблем с учетом примененных фильтров.

Дополнительно имеется кнопка для управления автоматическим обновлением данных.

#	Крит.	Проблема	График	Статус	Дата начала	Дата окончания	Уведомление	Объект	Описание проблемы
27270	●	Host clock not synchronising on pri-s...	📈	Активно	15.11.2024, 18:23:34	–	–	pri-server.aqu...	Clock not synchronising. Ensure NTP is configured on this host. V
27269	●	Kubernetes ReplicaSet mismatch (a...	📈	Завершено	15.11.2024, 18:04:47	15.11.2024, 18:13:00	–	ksm-kube-sta...	ReplicaSet astra-monitoring/mon-example-admin-ui-d8459bd4 r
27268	●	Kubernetes container oom killer (astr...	📈	Завершено	15.11.2024, 18:03:47	15.11.2024, 18:21:00	–	ksm-kube-sta...	Container prometheus in pod astra-monitoring/mon-dev-promethy
27267	●	Kubernetes Deployment replicas mis...	📈	Завершено	15.11.2024, 18:01:47	15.11.2024, 18:40:00	–	ksm-kube-sta...	Deployment astra-monitoring/mon-mon-1655-config-api replicas i
27266	●	Kubernetes ReplicaSet mismatch (a...	📈	Завершено	15.11.2024, 18:01:47	15.11.2024, 18:40:00	–	ksm-kube-sta...	ReplicaSet astra-monitoring/mon-mon-1655-config-api-68d4f54cl
27265	●	Kubernetes Deployment replicas mis...	📈	Завершено	15.11.2024, 17:21:47	15.11.2024, 18:42:00	–	ksm-kube-sta...	Deployment astra-monitoring/mon-mon-1592-clickhouse-1 replici
27264	●	Kubernetes ReplicaSet mismatch (a...	📈	Завершено	15.11.2024, 17:21:47	15.11.2024, 18:42:00	–	ksm-kube-sta...	ReplicaSet astra-monitoring/mon-mon-1592-clickhouse-1-b7c658
27263	●	Kubernetes Deployment replicas mis...	📈	Завершено	15.11.2024, 17:00:47	15.11.2024, 17:18:00	–	ksm-kube-sta...	Deployment astra-monitoring/mon-mon-1592-clickhouse-1 replici
27262	●	Kubernetes ReplicaSet mismatch (a...	📈	Завершено	15.11.2024, 17:00:47	15.11.2024, 17:18:00	–	ksm-kube-sta...	ReplicaSet astra-monitoring/mon-mon-1592-clickhouse-1-b7c658
27261	●	Unusual disk IO on pri-server.aquila a...	📈	Завершено	15.11.2024, 16:38:34	15.11.2024, 16:57:00	–	pri-server.aqu...	Time spent in IO is too high on pri-server.aquila.astralinux.ru. Che
27260	●	High CPU iowait on pri-server.aquila...	📈	Завершено	15.11.2024, 16:38:05	15.11.2024, 16:56:00	–	pri-server.aqu...	CPU iowait > 10%. A high iowait means that you are disk or netwo
27259	●	Kubernetes Deployment replicas mis...	📈	Активно	15.11.2024, 16:19:47	–	–	ksm-kube-sta...	Deployment astra-monitoring/mon-tul-1-config-api replicas mism

По щелчку левой кнопкой мыши (ЛКМ) на событии в правой части экрана открывается боковая панель с подробной информацией о событии.



В правом нижнем углу под таблицей находится функция навигации для перехода между страницами.



7.1.2 Основная таблица с проблемами

Таблица содержит следующие поля:

	Поле	Описание
1	#	Уникальный идентификатор проблемы
2	Крит.	Уровень критичности проблемы, выделенный цветом (активные события обозначаются ярким цветом, завершённые - тусклым): <ul style="list-style-type: none"> • Critical - красный круг • Warning - желтый круг • Info - синий круг • Неизвестное значение - серый круг со знаком вопроса
3	Проблема	Краткая информация о проблеме с контекстной гиперссылкой на соответствующий дашборд в Grafana

	Поле	Описание
4	Статус	Состояние проблемы, указывающее, активна она или уже завершена
5	Дата начала	Дата и время начала проблемы в формате в соответствии с настройками локализации браузера
6	Дата окончания	Дата и время окончания проблемы в формате в соответствии с настройками локализации браузера. Поле может быть пустым, если проблема еще активна
7	Уведомление	Статус уведомления о проблеме, обозначенный иконкой конверта, если уведомление отправлено или прочерком, если уведомление не отправлено
8	Объект	Название объекта мониторинга, к которому относится проблема
9	Описание проблемы	Краткое описание проблемы

7.1.3 Правая боковая панель

При щелчке ЛКМ на проблеме активируется боковая панель справа. Панель содержит полную информацию о данной проблеме.

Чтобы скрыть боковую панель, пользователь может либо нажать на значок "крестик" в верхнем левом углу, либо щелкнуть ЛКМ в области вне боковой панели.

The screenshot displays the Astra Monitoring web interface. On the left is a navigation sidebar with options like 'Мониторинг', 'Проблемы', 'Анализ логов', 'Конфигурация', 'Объекты', 'Шаблоны', 'Коллекторы', 'Теги', 'Администрирование', 'Ролевая модель', 'Каналы оповещений', 'Профиль', and 'Выйти'. The main area shows a 'Проблемы' (Problems) section with a table of active and resolved issues. A modal window is open on the right, providing details for a specific problem: 'Kubernetes container oom killer (astra-monitoring/mon-dev-prometheus-88b4c44d7-n6zhc:prometheus)'. The modal includes fields for object name, start/end dates, status (Completed), severity (Warning), and a detailed description of the OOMKilled event. A 'Посмотреть события' (View Events) button is visible at the bottom of the modal.

#	Крит.	Проблема	График	Статус	Дата начала	Дата окончания
27270	●	Host clock not synchronising on pri-s...	📊	Активно	15.11.2024, 18:23:34	–
27269	●	Kubernetes ReplicasSet mismatch (a...	📊	Завершено	15.11.2024, 18:04:47	15.11.2024, 18:13:...
27268	●	Kubernetes container oom killer (astr...	📊	Завершено	15.11.2024, 18:03:47	15.11.2024, 18:21:...
27267	●	Kubernetes Deployment replicas mis...	📊	Завершено	15.11.2024, 18:01:47	15.11.2024, 18:40:...
27266	●	Kubernetes ReplicasSet mismatch (a...	📊	Завершено	15.11.2024, 18:01:47	15.11.2024, 18:40:...
27265	●	Kubernetes Deployment replicas mis...	📊	Завершено	15.11.2024, 17:21:47	15.11.2024, 18:42:...
27264	●	Kubernetes ReplicasSet mismatch (a...	📊	Завершено	15.11.2024, 17:21:47	15.11.2024, 18:42:...
27263	●	Kubernetes Deployment replicas mis...	📊	Завершено	15.11.2024, 17:00:47	15.11.2024, 17:18:...
27262	●	Kubernetes ReplicasSet mismatch (a...	📊	Завершено	15.11.2024, 17:00:47	15.11.2024, 17:18:...
27261	●	Unusual disk IO on pri-server.aquila.a...	📊	Завершено	15.11.2024, 16:38:34	15.11.2024, 16:57:...
27260	●	High CPU iowait on pri-server.aquila...	📊	Завершено	15.11.2024, 16:38:05	15.11.2024, 16:56:...
27259	●	Kubernetes Deployment replicas mis...	📊	Активно	15.11.2024, 16:19:47	–

	Поле	Описание
1	Проблема	Краткая информация о проблеме
2	Объект	Название объекта мониторинга, к которому относится проблема
3	Дата начала	Дата и время начала проблемы в формате в соответствии с настройками локализации браузера
4	Дата окончания	Дата и время окончания проблемы в формате в соответствии с настройками локализации браузера. Поле может быть пустым, если проблема еще активна
5	Статус	Состояние проблемы, указывающее, активна она или уже завершена
6	Номер (ID) проблемы	Уникальный идентификатор проблемы
7	Критичность	Уровень критичности проблемы
8	Уведомление	Статус уведомления о проблеме, обозначенный иконкой конверта, если уведомление отправлено или прочерком, если уведомление не отправлено
9	URL	Гиперссылка на соответствующий дашборд в Grafana
10	Описание проблемы	Описание проблемы

	Поле	Описание
1 1	Посмотреть события	Кнопка для контекстного перехода на страницу с событиями (подробнее в "Пользовательское окно "События"")

7.1.4 Контекстный переход на страницу "События"

Каждая проблема является результатом обработки одного или нескольких событий. Чтобы просмотреть исходные события, нажмите кнопку "Просмотреть события" на боковой панели проблемы.

Это откроет контекстный переход в новом окне браузера с активным фильтром по соответствующей проблеме (подробнее в "Пользовательское окно "События"").

7.1.5 Дополнительные элементы

Над таблицей находятся интерактивные элементы, которые расширяют функциональность и возможности окна "Проблемы":

Проблемы 564

14.11.2024 18:39:48 15.11.2024 18:42:25 Выберите интервал Все Все (0)

#	Крит.	Проблема	График	Статус	Дата начала	Дата окончания	Уведомление	Объект	Описание проблемы
27270	●	Host clock not synchronising on pri-s...	📈	Активно	15.11.2024, 18:23:34	–	–	pri-server.aqu...	Clock not synchronising. Ensure NTP is configured on this host. V
27269	●	Kubernetes ReplicaSet mismatch (a...	📈	Завершено	15.11.2024, 18:04:47	15.11.2024, 18:13:00	–	ksm-kube-sta...	ReplicaSet astra-monitoring/mon-example-admin-ui-d8459bdd4 r

- задание даты и времени - границы диапазона из календаря

Проблемы 564

14.11.2024 18:39:48 15.11.2024 18:42:25 Выберите интервал Все Все (0)

Ноя 2024 18:39:48

#	Статус	Дата начала	Дата окончания	Уведомление	Объект
27270	Активно	15.11.2024, 18:23:34	–	–	pri-server.aqu...
27269	Завершено	15.11.2024, 18:04:47	15.11.2024, 18:13:00	–	ksm-kube-sta...
27268	Завершено	15.11.2024, 18:03:47	15.11.2024, 18:21:00	–	ksm-kube-sta...
27267	Завершено	15.11.2024, 18:01:47	15.11.2024, 18:40:00	–	ksm-kube-sta...
27266	Завершено	15.11.2024, 18:01:47	15.11.2024, 18:40:00	–	ksm-kube-sta...
27265	Завершено	15.11.2024, 17:21:47	15.11.2024, 18:42:00	–	ksm-kube-sta...
27264	Завершено	15.11.2024, 17:21:47	15.11.2024, 18:42:00	–	ksm-kube-sta...
27263	Завершено	15.11.2024, 17:00:47	15.11.2024, 17:18:00	–	ksm-kube-sta...

- задание предустановленного интервала

Проблемы 564

14.11.2024 18:39:48 15.11.2024 18:42:25 Выберите интервал Все Все (0)

#	Крит.	Проблема	График	Статус	Дата начала	Дата окончания	Уведомление
27270	●	Host clock not synchronising on pri-s...	📈	Активно	15.11.2024, 18:23:34	–	–
27269	●	Kubernetes ReplicasSet mismatch (a...	📈	Завершено	15.11.2024, 18:04:47	15.11.2024, 18:13:00	–
27268	●	Kubernetes container oom killer (astr...	📈	Завершено	15.11.2024, 18:03:47	15.11.2024, 18:21:00	–
27267	●	Kubernetes Deployment replicas mis...	📈	Завершено	15.11.2024, 18:01:47	15.11.2024, 18:40:00	–
27266	●	Kubernetes ReplicasSet mismatch (a...	📈	Завершено	15.11.2024, 18:01:47	15.11.2024, 18:40:00	–
27265	●	Kubernetes Deployment replicas mis...	📈	Завершено	15.11.2024, 17:21:47	15.11.2024, 18:42:00	–

- задание фильтра по признаку завершения/активности проблемы

Проблемы 564

14.11.2024 18:39:48 15.11.2024 18:42:25 Выберите интервал Все Все (0)

#	Крит.	Проблема	График	Статус	Дата начала	Дата окончания	Уведомление
27270	●	Host clock not synchronising on pri-s...	📈	Активно	15.11.2024, 18:23:34	–	–
27269	●	Kubernetes ReplicasSet mismatch (a...	📈	Завершено	15.11.2024, 18:04:47	15.11.2024, 18:13:00	–

- задание фильтра по критичности

Проблемы 564

14.11.2024 18:39:48 15.11.2024 18:42:25 Выберите интервал Все Все (0)

#	Крит.	Проблема	График	Статус	Дата начала	Дата окончания	Уведомление
27270	●	Host clock not synchronising on pri-s...	📈	Активно	15.11.2024, 18:23:34	–	–
27269	●	Kubernetes ReplicasSet mismatch (a...	📈	Завершено	15.11.2024, 18:04:47	15.11.2024, 18:13:00	–
27268	●	Kubernetes container oom killer (astr...	📈	Завершено	15.11.2024, 18:03:47	15.11.2024, 18:21:00	–

- открытие конструктора дополнительных фильтров

Проблемы 564

14.11.2024 18:39:48 15.11.2024 21:01:20 Выберите интервал Все Все (0)

Дополнительные фильтры

Сохранить Сохранить как

Добавить группу

Применить Сбросить

#	Крит.	Статус	Дата начала	Дата окончания	Длительность	Имя проблемы
27270	●	Активна				
27269	●	Активна				
27268	●	Активна				
27267	●	Активна				
27266	●	Активна				
27265	●	Активна				
27264	●	Завершено	15.11.2024, 17:21:47	15.11.2024, 18:42:00		Kubernetes ReplicasSet mismatch (a...

	Элемент интерфейса	Описание
1	Кнопка автообновления	Эта кнопка позволяет пользователю управлять автоматическим обновлением данных на странице. По умолчанию данные обновляются каждые 60 секунд. Пользователь может приостановить автоматическое обновление, нажав на кнопку, и возобновить его, нажав на нее еще раз
2	Поиск по заданной дате начала проблемы	Этот поисковый инструмент позволяет быстро находить проблемы, которые были созданы в определенном диапазоне дат (параметр Дата начала). Пользователь может указать начальную и конечную даты и времена для поиска. В этом случае отобразятся только проблемы, попадающие в заданный интервал. Поиск по дате исключает поиск по динамическому временному интервалу
3	Поиск по заданному временному интервалу	Этот поисковый инструмент позволяет быстро находить проблемы, созданные в определенный период времени. Пользователь может выбрать один из предустановленных временных интервалов: последние 10 минут, 1 час, 3 часа, 12 часов и 24 часа. Поиск по динамическому временному интервалу исключает поиск по дате начала проблемы
4	Поиск по статусу	Этот переключатель позволяет фильтровать и отображать только активные или завершенные проблемы в таблице (параметр "Статус"). По умолчанию он включен на "Все" для исключения дефолтной фильтрации по этому признаку
5	Активные фильтры по критичности	Этот поисковый инструмент позволяет отфильтровать проблемы по критичности (параметр "Крит."). По умолчанию он включен на "Все" для исключения дефолтной фильтрации по этому признаку

	Элемент интерфейса	Описание
6	Кнопка дополнительных фильтров	Эта кнопка позволяет пользователю раскрыть/скрыть секцию для задания дополнительных фильтров
7	Поиск по названию объекта мониторинга или описанию	С помощью этой функции можно осуществлять поиск проблем на основе названия объектов мониторинга и описания проблем. Пользователь может ввести ключевые слова или фразы, характеризующие объекты или описание. В этом случае отобразятся только проблемы, соответствующие заданным поисковым критериям

7.2 Пользовательское окно "События"

7.2.1 Общая информация

Окно "События" содержит таблицу, которая отображает список событий и соответствующие данные. Таблица включает интерактивные элементы для сортировки записей.

Над таблицей расположены элементы управления, которые расширяют возможности поиска и фильтрации, отображают количество найденных событий с учетом примененных фильтров.

Дополнительно есть кнопка для управления автоматическим обновлением данных.

Крит.	Объект	Проблема	Дата начала	Дата окончания	Кол-во.	Описание
●	ksm-kube-state-metrics.astra-monitoring...	Kubernetes Deploymen...	15.11.2024, 20:51:47	15.11.2024, 21:14:00	15	Deployment astra-monitoring/mon-mon-1592-clickhouse-1 replicas mismatch VAL...
●	ksm-kube-state-metrics.astra-monitoring...	Kubernetes ReplicasSet...	15.11.2024, 20:51:47	15.11.2024, 21:14:00	15	ReplicaSet astra-monitoring/mon-mon-1592-clickhouse-1-b7c65885c replicas mis...
●	ksm-kube-state-metrics.astra-monitoring...	Kubernetes Deploymen...	15.11.2024, 20:09:47	15.11.2024, 20:48:00	31	Deployment astra-monitoring/mon-mon-1592-clickhouse-1 replicas mismatch VAL...
●	ksm-kube-state-metrics.astra-monitoring...	Kubernetes ReplicasSet...	15.11.2024, 20:09:47	15.11.2024, 20:48:00	31	ReplicaSet astra-monitoring/mon-mon-1592-clickhouse-1-b7c65885c replicas mis...
●	ksm-kube-state-metrics.astra-monitoring...	Kubernetes Deploymen...	15.11.2024, 19:44:47	–	104	Deployment astra-monitoring/mon-mon-1655-config-api replicas mismatch VALUE ...
●	ksm-kube-state-metrics.astra-monitoring...	Kubernetes ReplicasSet...	15.11.2024, 19:44:47	–	104	ReplicaSet astra-monitoring/mon-mon-1655-config-api-68d4f54cb4 replicas mism...
●	ksm-kube-state-metrics.astra-monitoring...	Kubernetes Deploymen...	15.11.2024, 19:43:47	15.11.2024, 20:06:00	15	Deployment astra-monitoring/mon-mon-1592-clickhouse-1 replicas mismatch VAL...
●	ksm-kube-state-metrics.astra-monitoring...	Kubernetes ReplicasSet...	15.11.2024, 19:43:47	15.11.2024, 20:06:00	15	ReplicaSet astra-monitoring/mon-mon-1592-clickhouse-1-b7c65885c replicas mis...
●	pri-server.aquila.astralinux.ru	Unusual disk IO on pri-s...	15.11.2024, 19:20:34	15.11.2024, 19:39:00	11	Time spent in IO is too high on pri-server.aquila.astralinux.ru. Check storage for iss...
●	pri-server.aquila.astralinux.ru	High CPU iowait on pri-...	15.11.2024, 19:20:05	15.11.2024, 19:39:00	11	CPU iowait > 10%. A high iowait means that you are disk or network bound. VALUE ...
●	ksm-kube-state-metrics.astra-monitoring...	Kubernetes Deploymen...	15.11.2024, 18:45:47	15.11.2024, 19:40:00	47	Deployment astra-monitoring/mon-mon-1592-clickhouse-1 replicas mismatch VAL...
●	ksm-kube-state-metrics.astra-monitoring...	Kubernetes ReplicasSet...	15.11.2024, 18:45:47	15.11.2024, 19:40:00	47	ReplicaSet astra-monitoring/mon-mon-1592-clickhouse-1-b7c65885c replicas mis...

При щелчке левой кнопкой мыши (ЛКМ) на событии в правой части экрана открывается боковая панель с подробной информацией о нем.

The screenshot shows the Astra Monitoring interface. On the left is a navigation sidebar with options like 'Мониторинг', 'Проблемы', 'Анализ логов', 'Конфигурации', 'Объекты', 'Шаблоны', 'Коллекторы', 'Теги', 'Администрирование', 'Ролевая модель', 'Каналы оповещений', and 'Профиль'. The main area displays a table of events under the 'События' tab. The table has columns for 'Крит.' (Criticality), 'Объект' (Object), 'Проблема' (Problem), 'Дата начала' (Start Date), 'Дата окончания' (End Date), 'Кол-во' (Count), and 'Описание' (Description). A detailed view of an event is shown on the right, including fields for 'Объект', 'Дата начала', 'Дата окончания', 'Статус', 'ID события', 'Критичность', 'Группа событий', 'Ключ', 'TTL', 'Количество событий', 'Имя аддитера', 'Event processor flag', and 'Описание'.

В правом нижнем углу под таблицей находится функция навигации для перехода между страницами.

The screenshot shows a pagination control at the bottom right of the event table. It includes a dropdown menu for selecting the number of items per page, with options: 10 / стр., 20 / стр., 50 / стр., and 100 / стр. The current page is 1, and there are 2 pages in total. The search bar shows '20 / стр.'.

7.2.2 Основная таблица с событиями

Таблица содержит следующие поля по каждому событию:

	Поле	Описание
1	Крит.	Уровень критичности события, выделенный цветом (активные события обозначаются ярким цветом, завершённые - тусклым): <ul style="list-style-type: none"> • Critical - красный круг • Warning - желтый круг • Info - синий круг • Неизвестное значение - серый круг со знаком вопроса
2	Объект	Объект мониторинга, к которому относится событие
3	Проблема	Проблема, которая инициирована или связана с данным событием

	Поле	Описание
4	Дата начала	Дата и время первого прихода события в формате в соответствии с настройками локализации браузера
5	Дата окончания	Дата и время результирующего события в формате в соответствии с настройками локализации браузера. Поле может быть пустым, если событие еще активно
6	Кол-во	Количество таких событий, как результат подсчёта в процессе дедупликации
7	Описание	Краткое описание события

7.2.3 Правая боковая панель

При щелчке ЛКМ на событии активируется боковая панель справа. Панель содержит полную информацию о данном событии.

Чтобы скрыть боковую панель, пользователь может либо нажать на значок "крестик" в верхнем левом углу, либо щелкнуть ЛКМ в области вне боковой панели.

The screenshot displays the Astra Monitoring web interface. On the left is a navigation sidebar with options like 'Мониторинг', 'Проблемы', 'Анализ логов', 'Конфигурации', 'Объекты', 'Шаблоны', 'Коллекторы', 'Теги', 'Администрирование', 'Ролевая модель', 'Каналы оповещений', 'Профиль', and 'Выйти'. The main area is titled 'События' (Events) and shows a table of events with columns for 'Крит.' (Criticality), 'Объект' (Object), 'Проблема' (Problem), 'Дата начала' (Start Date), 'Дата окончания' (End Date), 'Кол-во' (Count), and 'Описание' (Description). A detailed modal window is open on the right, showing information for a specific event: 'Kubernetes DeploymentReplicasMismatch' for the object 'kubernetes-metrics-astra-monitoring.svc.cluster.local'. The modal includes fields for 'Дата начала' (15.11.2024, 19:44:47), 'Дата окончания' (15.11.2024, 21:31:53), 'Статус' (Активно), 'ID события' (27280), 'Критичность' (Предупреждение), 'Группа событий' (KubestateExporter_KubernetesDeploymentReplicasMismatch), 'Ключ' (astra-monitoring_mon-mon-1655-config-api), 'TTL' (—), 'Количество событий' (108), 'Имя адаптера' (0), 'Event processor flag' (Нет), and 'Проблема' (Kubernetes Deployment replicas mismatch (astra-monitoring/mon-mon-1655-config-api)). The description details a 'Deployment mismatch VALUE = 1 LABELS = map[_name=_kube_deployment_spec_replicas alertgroup:KubestateExporter alertname:KubernetesDeploymentReplicasMismatch deployment:mon-mon-1655-config-api group:RDnNNI hostname:kubernetes-metrics-astra-monitoring.svc.cluster.local instance:kubernetes-metrics-astra-monitoring.svc.cluster.local:8080 job:https namespace:astra-monitoring severity:warning]'. A 'История изменений' (Change History) button is visible at the bottom of the modal.

	Поле	Описание
1	Объект	Объект мониторинга, к которому относится событие
2	Дата начала	Дата и время первого прихода события в формате в соответствии с настройками локализации браузера
3	Дата окончания	Дата и время результирующего события в формате в соответствии с настройками локализации браузера
4	Последнее обновление	Дата и время последнего прихода этого же события в формате в соответствии с настройками локализации браузера
5	Статус	Состояние события, указывающее, активно оно или уже завершено
6	ID события	Уникальный идентификатор события
7	Критичность	Уровень критичности события, выделенный цветом (активные события обозначаются ярким цветом, завершённые - тусклым)
8	Группа событий	Приходит от Event Adapter'a и по сути представляет собой краткое описание
9	Ключ	Ключ
10	TTL	Время до автоматического закрытия (для событий, по которым не приходит Resolution)
11	Количество событий	Количество таких событий, как результат подсчёта в процессе дедупликации

	Поле	Описание
12	Имя адаптера	Имя адаптера
13	Event processor flag	Указывает обрабатывалось ли событие eventProcessor'ом
14	Проблема	Проблема, к которой относится событие
15	Описание	Более подробная информация о событии
16	История изменений	Кнопка для просмотра Истории изменения событий

7.2.4 История изменения событий

Чтобы посмотреть историю изменения конкретного события, нажмите кнопку "История изменений" на боковой панели этого события.

Это откроет дополнительную боковую панель с историей изменения данного события.

Чтобы скрыть боковую панель, пользователь может либо нажать на значок "крестик" в верхнем левом углу, либо щелкнуть ЛКМ в области вне боковой панели.

The screenshot displays the Astra Monitoring web interface. On the left is a navigation sidebar with options like 'Мониторинг', 'Проблемы', 'Анализ логов', 'Конфигурация', 'Объекты', 'Шаблоны', 'Коллекторы', 'Телы', 'Администрирование', 'Ролевая модель', 'Каналы оповещения', 'Профиль', and 'Выход'. The main area shows a 'События' (Events) table with columns for severity, object, problem, start date, and end date. A table of events is visible below the header.

Крит.	Объект	Проблема	Дата начала	Дата окончания
●	k8m-kube-state-metrics.astra-monitoring...	Kubernetes Deploymen...	15.11.2024, 22:57:47	–
●	k8m-kube-state-metrics.astra-monitoring...	Kubernetes ReplicasSet...	15.11.2024, 22:57:47	–
●	k8m-kube-state-metrics.astra-monitoring...	Kubernetes Deploymen...	15.11.2024, 22:36:47	15.11.2024, 22:54:00
●	k8m-kube-state-metrics.astra-monitoring...	Kubernetes ReplicasSet...	15.11.2024, 22:36:47	15.11.2024, 22:54:00
●	k8m-kube-state-metrics.astra-monitoring...	Kubernetes Deploymen...	15.11.2024, 22:24:47	–
●	k8m-kube-state-metrics.astra-monitoring...	Kubernetes ReplicasSet...	15.11.2024, 22:24:47	–
●	pri-server.aquila.astralinux.ru	Unusual disk IO on pri-s...	15.11.2024, 22:03:34	15.11.2024, 22:22:00
●	pri-server.aquila.astralinux.ru	High CPU iowait on pri...	15.11.2024, 22:03:05	15.11.2024, 22:22:00
●	k8m-kube-state-metrics.astra-monitoring...	Kubernetes Deploymen...	15.11.2024, 21:54:47	15.11.2024, 22:33:00
●	k8m-kube-state-metrics.astra-monitoring...	Kubernetes ReplicasSet...	15.11.2024, 21:54:47	15.11.2024, 22:33:00
●	k8m-kube-state-metrics.astra-monitoring...	Kubernetes Deploymen...	15.11.2024, 21:33:47	15.11.2024, 21:51:00
●	k8m-kube-state-metrics.astra-monitoring...	Kubernetes ReplicasSet...	15.11.2024, 21:33:47	15.11.2024, 21:51:00

On the right, a detailed view of an event is shown, including its object, dates, status, and a description of the problem. A blue button labeled 'История изменений' (Change History) is visible at the bottom of the event details panel.

7.2.5 Дополнительные элементы

Над таблицей находятся интерактивные элементы, которые расширяют функциональность и возможности окна "События" и аналогичны таким же элементам на странице "Проблемы":

Крит.	Объект	Проблема	Дата начала	Дата окончания	Кол-во	Описание
●	kubernetes-metrics.astra-monitoring...	Kubernetes Deploymen...	15.11.2024, 22:57:47	–	13	Deployment astra-monitoring/mon-mon-1592-clickhouse-1 replicas mismatch VAL...
●	kubernetes-metrics.astra-monitoring...	Kubernetes ReplicasSet...	15.11.2024, 22:57:47	–	13	ReplicaSet astra-monitoring/mon-mon-1592-clickhouse-1-b7c65885c replicas mis...

	Элемент интерфейса	Описание
1	Кнопка автообновления	Эта кнопка позволяет пользователю управлять автоматическим обновлением данных на странице. По умолчанию данные обновляются каждые 60 секунд. Пользователь может приостановить автоматическое обновление, нажав на кнопку, и возобновить его, нажав на нее еще раз
2	Поиск по заданной дате начала проблемы	Этот поисковый инструмент позволяет быстро находить события, которые были созданы в определенном диапазоне дат (параметр Дата начала). Пользователь может указать начальную и конечную даты и времена для поиска. В этом случае отобразятся только события, попадающие в заданный интервал. Поиск по дате исключает поиск по динамическому временному интервалу
3	Поиск по заданному временному интервалу	Этот поисковый инструмент позволяет быстро находить события, созданные в определенный период времени. Пользователь может выбрать один из предустановленных временных интервалов: последние 10 минут, 1 час, 3 часа, 12 часов и 24 часа. Поиск по динамическому временному интервалу исключает поиск по дате начала проблемы
5	Активные фильтры по критичности	Этот поисковый инструмент позволяет отфильтровать события по критичности (параметр "Крит."). По умолчанию он включен на "Все" для исключения дефолтной фильтрации по этому признаку
6	Кнопка дополнительных фильтров	Эта кнопка позволяет пользователю раскрыть/скрыть секцию для задания дополнительных фильтров

	Элемент интерфейса	Описание
7	Поиск по названию объекта мониторинга (Monitoring object) и описанию (Description)	С помощью этой функции можно осуществлять поиск событий на основе названия объектов мониторинга и описания событий. Пользователь может ввести ключевые слова или фразы, характеризующие объекты или описание. В этом случае отобразятся только события, соответствующие заданным поисковым критериям

7.3 Пользовательское окно "Объекты"

7.3.1 Общая информация

Окно "Объекты" содержит таблицу, которая отображает список объектов и соответствующие данные. Таблица включает интерактивные элементы для фильтрации записей.

Для каждого объекта предусмотрены интерактивные элементы для клонирования, редактирования и удаления объекта (для активных объектов), восстановление и безвозвратное удаление (для удаленных объектов).

Над таблицей расположены элементы управления, которые расширяют возможности поиска и фильтрации, отображают количество найденных объектов с учётом примененных фильтров.

Дополнительно есть кнопка для создания объектов.

FQDN/Ip-адрес	Имя объекта	Активирован	Коллектор	Действия
object_active_false	object_active_false	<input checked="" type="checkbox"/>	dc02.example.loc	
object_active_true	object_active_true	<input checked="" type="checkbox"/>	dc02.example.loc	
101b842ad182	Объект NEW с агентом на 101b842ad182	<input checked="" type="checkbox"/>	101b842ad182-name	
32423	23423	<input checked="" type="checkbox"/>	dc02.example.loc	
fwefewf	afwefewfewfwefewfewfwef	<input checked="" type="checkbox"/>	040d83a998c1	
djkdkd	bvz	<input checked="" type="checkbox"/>	dc02.example.loc	
345	54235	<input checked="" type="checkbox"/>	dc02.example.loc	
555	555	<input checked="" type="checkbox"/>	dc02.example.loc	
dc01.ald.pro	testSK01	<input checked="" type="checkbox"/>	dc02.example.loc	
g	gg	<input checked="" type="checkbox"/>	dc02.example.loc	

В правом нижнем углу под таблицей находится функция навигации для перехода между страницами.

345	54235		dc02.example.loc	
555	555		dc02.example.loc	
dc01.aikd.pro	testSK01		dc02.example.loc	
g	99		dc02.example.loc	

< 1 2 3 > 10 / стр.

- 10 / стр.
- 20 / стр.
- 50 / стр.
- 100 / стр.

7.3.2 Основная таблица с объектами

Таблица содержит следующие поля по каждому объекту мониторинга:

	Поле	Описание
1	FQDN/Ip-адрес	Адрес объекта мониторинга
2	Имя объекта	Имя объекта мониторинга в текстовом виде
3	Признак активации	Показывает, включен сбор данных от объекта или нет
4	Коллектор	Имя коллектора, на который интерфейсы объекта передают диагностические данные
5	Действия	Интерактивные элементы для клонирования, редактирования и удаления объекта (для активных объектов), восстановление и безвозвратное удаление (для удаленных объектов)

7.3.3 Дополнительные элементы

Над таблицей находятся интерактивные элементы, которые расширяют функциональность и возможности окна "Объекты":

Объекты 25

Активные

FQDN/Ip-адрес	Имя объекта	Активирован	Коллектор	Действия
object_active_false	object_active_false		dc02.example.loc	
object_active_true	object_active_true		dc02.example.loc	

	Элемент интерфейса	Описание
1	Переключатель статуса	Этот переключатель позволяет фильтровать и отображать только активные или удалённые объекты
2	Кнопка создания объекта	С помощью этой кнопки можно создать новый объект мониторинга. По клику вызывается соответствующее модальное окно
3	Поиск по полям "FQDN/IP-адрес", "Имя объекта"	С помощью этой функции можно осуществлять поиск объектов по вхождению введенного в поле контента в FQDN/IP-адрес объекта или Имя объекта

7.3.4 Форма для создания/редактирования/клонирования объекта

Кликом ЛКМ по кнопке создания объекта или элементу клонирования/редактирования в таблице открывается форма для соответствующих действий с объектом. Дефолтно на форме открыта вкладка "Объект".

Редактировать объект ✕

Объект Интерфейсы

* **Имя объекта**

* **IP/FQDN адрес**

* **Коллектор** ▾

Описание

Включить сбор данных:

Форма содержит следующие элементы:

	Элемент интерфейса	Описание
1	Имя объекта	Имя объекта мониторинга в текстовом виде
2	FQDN/Ip-адрес	Адрес объекта мониторинга
3	Коллектор	Имя и адрес коллектора, на который интерфейсы объекта передают диагностические данные
4	Описание	Описание объекта
5	Включить сбор данных	Показывает, включен сбор данных от объекта или нет
6	Отменить	Кнопка, для отмены изменений данных на форме
7	Сохранить	Кнопка для сохранения данных формы

Для настройки интерфейсов объекта используется вкладка "Интерфейсы".

а. Настройка Агента

! Конфигурирование Агента доступно, только если он установлен на хосте объекта и зарегистрирован на Платформе. Иначе, субвкладка "Агент" скрыта.

Редактировать объект ✕

Объект **Интерфейсы**

Агент Exporter

Имя объекта:

Источники данных/экспортеры

<input type="text" value="one"/>	
<input type="text" value="freeipa"/>	

Добавить экспортер

Сторонние экспортеры

Добавить экспортер

Редактировать объект [X]

Объект Интерфейсы

Агент Exporter

Имя объекта:

Источники данных/экспортёры

🗑️

Сторонние экспортёры

Форма содержит следующие элементы:

	Элемент интерфейса	Описание
1	Имя объекта	Имя объекта мониторинга в текстовом виде
2	Внутренние экспортёры	Список внутренних экспортёров, которыми управляет Агент. Кликом ЛКМ по имени экспортёра открывается форма редактирования его настроек
3	Добавить внутренний экспортёр	Кнопка добавления внутреннего экспортёра в список. По кнопке открывается форма задания настроек внутреннего экспортёра
4	Удалить внутренний экспортёр	Кнопка удаления соответствующего внутреннего экспортёра из списка
5	Сторонние экспортёры	Список сторонних экспортёров, которыми управляет Агент. Кликом ЛКМ по имени экспортёра открывается форма редактирования его настроек

	Элемент интерфейса	Описание
6	Добавить сторонний экспортер	Кнопка добавления стороннего экспортера в список (аналогично внутренним экспортерам). По кнопке открывается форма задания настроек стороннего экспортера
7	Удалить сторонний экспортер	Кнопка удаления соответствующего стороннего экспортера из списка (аналогично внутренним экспортерам)
8	Отменить	Кнопка, для отмены изменений данных на форме
9	Сохранить	Кнопка для сохранения данных формы

- Форма редактирования настроек внутреннего экспортера

Редактировать агент ✕

* Имя экспортера:

Labels:

:

:

Config:

:

Интервал, сек: go_metrics:

Редактировать агент ✕

* Имя экспортера: ▼

Labels:

Config:

Интервал, сек: go_metrics:

Форма содержит следующие элементы:

	Элемент интерфейса	Описание
1	Имя экспортера	Имя внутреннего экспортера, выбирается из списка доступных экспортеров
2	Labels	Лейблы для собираемых метрик. Формат Ключ:Значение
3	Config	Параметры конфигурационного файла экспортера. Формат Ключ:Значение
4	Интервал, сек	Интервал сбора метрик в секундах
5	go_metrics	Включает дефолтные метрики, собираемые пакетом Prometheus на go, в которых информация о ресурсах, потребляемых экспортером
6	Отменить	Кнопка, для отмены изменений данных на форме
7	Сохранить	Кнопка для сохранения данных формы

- Форма задания настроек нового стороннего экспортера

Добавить экспортер ✕

* Имя стороннего экспортера:

* Адрес:порт сервера метрик: * URL путь метрик:

* Интервал сбора метрик, сек:

Лейблы для собираемых метрик:

Форма содержит следующие элементы:

	Элемент интерфейса	Описание
1	Имя стороннего экспортера	Имя стороннего экспортера, вводится в поле руками
2	Адрес сервера метрик	В формате <ip-адрес>:<порт>
3	URL путь метрик	Начинается с "/"
4	Интервал сбора метрик, сек	Целое неотрицательное число
5	Лейблы для собираемых метрик	Формат Ключ:Значение

	Элемент интерфейса	Описание
6	Отменить	Кнопка, для отмены изменений данных на форме
7	Сохранить	Кнопка для сохранения данных формы

b. Настройка сторонних экспортеров, собирающих данные с объекта и предоставляющих их на коллектор, минуя Агента

Их можно сконфигурировать на субвкладке "Exporter" аналогично тому, как это описано выше для сторонних экспортеров, управляемых Агентом.

⚠ Если Агента у объекта нет, то субвкладка "Exporter" не отображается.

Редактировать объект ✕

Объект **Интерфейсы**

Имя объекта:

Источники данных/сторонние экспортеры

🗑

🗑

+ Добавить экспортер

Редактировать экспортер ✕

* Имя стороннего экспортера:

* Адрес:порт сервера метрик: * URL путь метрик:

* Интервал сбора метрик, сек:

Лейблы для собираемых метрик:

7.4 Пользовательское окно "Интерфейсы объектов"

7.4.1 Общая информация

Окно "Интерфейсы объектов" содержит таблицу, которая отображает список интерфейсов и соответствующие данные. Окно предоставляет пользователю сквозной доступ к интерфейсам объектов и позволяет контролировать их активацию и статусы.

Интерфейсы объектов 12

FQDN/IP адрес	Имя объекта	Элемент объекта	Тип интерфейса	Активирован	Статус
1a5b4c715eeb	Объект с агентом на 1a5b4c715eeb	01x0b7a08cc65524	agent	<input checked="" type="checkbox"/>	—
dc02.example.loc	Объект с агентом на dc02.example.loc	01x0b880b79da102	agent	<input checked="" type="checkbox"/>	—
fa8770ff503b	Объект с агентом на fa8770ff503b	01x08367c13cefdd	agent	<input checked="" type="checkbox"/>	—
fa8770ff503b	Объект с агентом на fa8770ff503b	01x08367c13cefdd	exporter	<input checked="" type="checkbox"/>	—
dc03	dc03	01x09b744238ee5b	exporter	<input checked="" type="checkbox"/>	—
test1	test1	01x09898d4fd8dbd	exporter	<input checked="" type="checkbox"/>	—
040d83a998c9	Объект с агентом на 040d83a998c9	01x0c85227e14b51	agent	<input checked="" type="checkbox"/>	—
test1	test	01x0d33b95917653	exporter	<input checked="" type="checkbox"/>	—
localhost	NEW	01x07028ae9249ba	exporter	<input checked="" type="checkbox"/>	—
101b842ad182	Объект NEW с агентом на 101b842ad182	01x06685dc1ce7a4	agent	<input checked="" type="checkbox"/>	—

> | 10 / стр. ▾

В правом нижнем углу под таблицей находится функция навигации для перехода между страницами.

7.4.2 Основная таблица с интерфейсами

Таблица содержит следующие поля по каждому интерфейсу:

	Поле	Описание
1	FQDN/Ip-адрес	Адрес объекта, мониторинга, к которому относится интерфейс
2	Имя объекта	Имя объекта мониторинга, к которому относится интерфейс
3	Элемент объекта	Элемент объекта мониторинга, данные с которого собирает интерфейс
4	Тип интерфейса	Может быть один из: <ul style="list-style-type: none"> • agent, • exporter
5	Признак активации	Указывает на включение интерфейса для сбора данных
6	Статус	Состояние интерфейса

7.4.3 Дополнительные элементы

Над таблицей находятся интерактивные элементы, которые расширяют функциональность и возможности окна "Интерфейсы объектов":

Интерфейсы объектов Поиск...

FQDN/IP адрес	Имя объекта	Элемент объекта	Тип интерфейса	Активирован	Статус
1a5b4c715eeb	Объект с агентом на 1a5b4c715eeb	01x0b7a08cc65524	agent	<input checked="" type="checkbox"/>	—

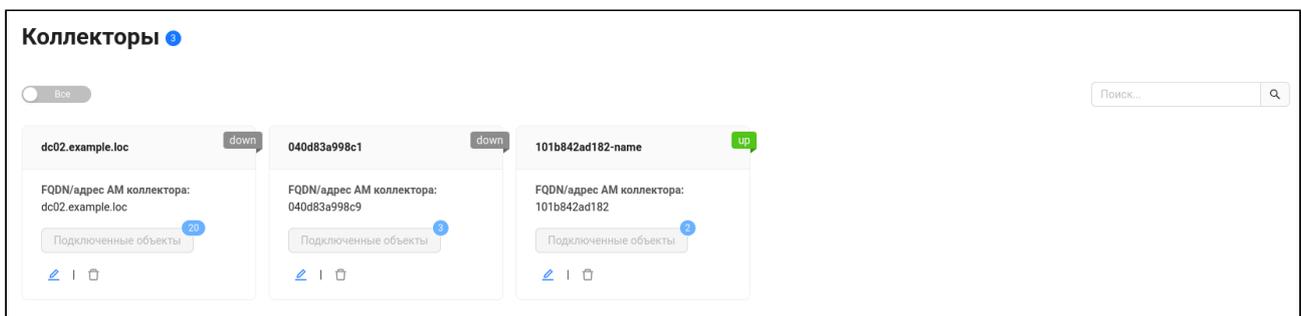
	Элемент интерфейса	Описание
1	Поиск по полям "FQDN/IP-адрес", "Имя объекта", "Элемент объекта"	С помощью этой функции можно осуществлять поиск интерфейсов объектов по вхождению введенного в поле контента в FQDN/IP-адрес объекта или Имя объекта или Элемент объекта

7.5 Пользовательское окно "Коллекторы"

7.5.1 Общая информация

Окно "Коллекторы" содержит список карточек объектов с соответствующими данными.

Над списком расположены элементы управления, которые расширяют возможности поиска и фильтрации, отображают количество найденных коллекторов с учётом примененных фильтров.

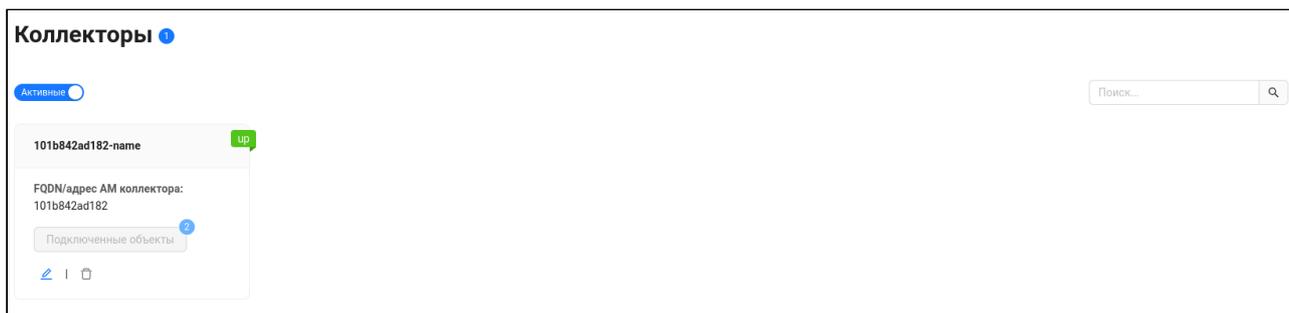


В правом нижнем углу страницы находится функция навигации для перехода между страницами.

7.5.2 Основной список карточек коллекторов

Карточка коллектора автоматически появляется в списке после успешной регистрации коллектора на Платформе.

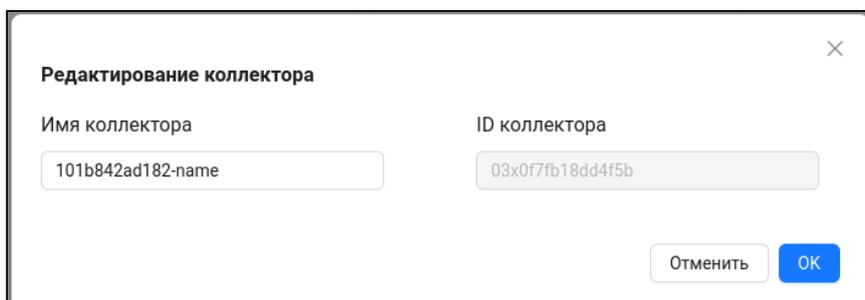
Каждая карточка содержит следующие данные:



	Поле	Описание
1	Имя коллектора	Имя коллектора
2	FQDN/Ip-адрес коллектора	Адрес коллектора, определяется автоматически при регистрации коллектора
3	Подключенные объекты	Количество объектов, интерфейсы которых передают данные в коллектор
4	Действия	Интерактивные элементы для редактирования и удаления коллектора

⚠️ Активные коллекторы или с подключенными объектами удалить нельзя.

При редактировании коллектора к изменению доступно только его имя.



7.5.3 Дополнительные элементы

Над списком карточек находятся интерактивные элементы, которые расширяют функциональность и возможности окна "Коллекторы":



	Элемент интерфейса	Описание
1	Переключатель статуса	Этот переключатель позволяет фильтровать и отображать только активные или удалённые коллекторы
2	Поиск по полям "FQDN/IP-адрес AM коллектора", "Имя коллектора"	С помощью этой функции можно осуществлять поиск коллекторов по вхождению введенного в поле контента в FQDN/IP-адрес коллектора или Имя коллектора

7.6 Пользовательское окно "Теги"

7.6.1 Общая информация

Окно "Теги" содержит таблицу, которая отображает список тегов, сгруппированных по имени. Таблица включает интерактивные элементы для сортировки записей.

Над таблицей расположены элементы управления, которые расширяют возможности поиска, отображают количество найденных тегов с учетом примененных фильтров.

Дополнительно имеются кнопки для создания тегов и принудительного обновления страницы.

Теги 311

Поиск...

Тег	Действия
+ Za	
+ zfsVcYeD_aOWwwXmQueEBVMrcnhmvsCX5_AM_AodWa50QjX1fZE	
+ ZGUgCGbviVvsyoeFZfmBwJtgrkjQrL_eL7UenMvRA9gLaerli	
+ ZmgFcthQGHGlyVaAraPIUgnZb10f2_luLagUoOqeUj8Dq6Nysi	
+ ZvNjkwzhqBsnsgjWISLgIAFnjYWHV1EFKcaHVncHHDULVdAqT	
- Тер-тест1	
Тер-тест1 : значение тега-теста24	🟢 🔍 🗑️
Тер-тест1 : значение тега-теста19	🟢 🔍 🗑️
Тер-тест1 : значение тега-теста6	🟢 🔍 🗑️
Тер-тест1 : значение тега-теста3	🟢 🔍 🗑️
Тер-тест1 : значение тега-теста5	🟢 🔍 🗑️
Тер-тест1 : значение тега-теста15	🟢 🔍 🗑️
Тер-тест1 : значение тега-теста25	🟢 🔍 🗑️
Тер-тест1 : значение тега-теста14	🟢 🔍 🗑️
Тер-тест1 : значение тега-теста28	🟢 🔍 🗑️
Тер-тест1 : значение тега-теста16	🟢 🔍 🗑️
Тер-тест1 : значение тега-теста18	🟢 🔍 🗑️

В правом нижнем углу под таблицей находится функция навигации для перехода между страницами.

- afsD5pbBtuRQNQTrZvSPkcQM4tpdAHPc2_UldR3udGvNhDzj		
afsD5pbBtuRQNQTrZvSPkcQM4tpdAHPc2_UldR3udGvNhDzj : jLScYcuMjMjYMaipcDavARF-vpEEM141nKermtlb0kTWOl1L_YeU	🟢 🔍 🗑️	
- AgzWvmClefTRdbmuxejkmERtVihHSWq4HLXMMsX7rSf6JoHYI		
AgzWvmClefTRdbmuxejkmERtVihHSWq4HLXMMsX7rSf6JoHYI : VZnHnOHg8TVYOSrUjigwgGysjNvP49jX4JKMIPNISTfoRFw	🟢 🔍 🗑️	
- AICmLLGDpdaDOHCdOxBotwiYabaHyjp_SwV9TQV4_wQqidmMzY		
AICmLLGDpdaDOHCdOxBotwiYabaHyjp_SwV9TQV4_wQqidmMzY : zwPKVnpbiDEcBSTLclFKTOaJ8nI64skpk0Uc4Ryr3eeen9B_IB	🟢 🔍 🗑️	
AICmLLGDpdaDOHCdOxBotwiYabaHyjp_SwV9TQV4_wQqidmMzY : DWwWcXUIMfiaoQkzLvgnzGfzOzAylypNmZnrUGfdFOIXVo	🟢 🔍 🗑️	

10 / стр.
20 / стр.
50 / стр.
100 / стр.

< 1 2 3 4 5 ... 11 > 30 / стр. 🔍

7.6.2 Основная таблица с тегами

Таблица содержит следующий данные по каждому тегу:

	Объект интерфейса	Значение
1	Тег	Полное название тега, сформированное на основе имени и значения ("имя : значение"). В случае, если тег уже используется в объектах будет дополнительно отображаться иконка красного закрытого замка, а часть действий с тегом будет недоступна (кнопки с недоступными действиями будут неактивны)

	Объект интерфейса	Значение
2	Действия	Интерактивные элементы для клонирования, редактирования и удаления объекта (для неиспользуемых тегов), клонирования (для используемых тегов)

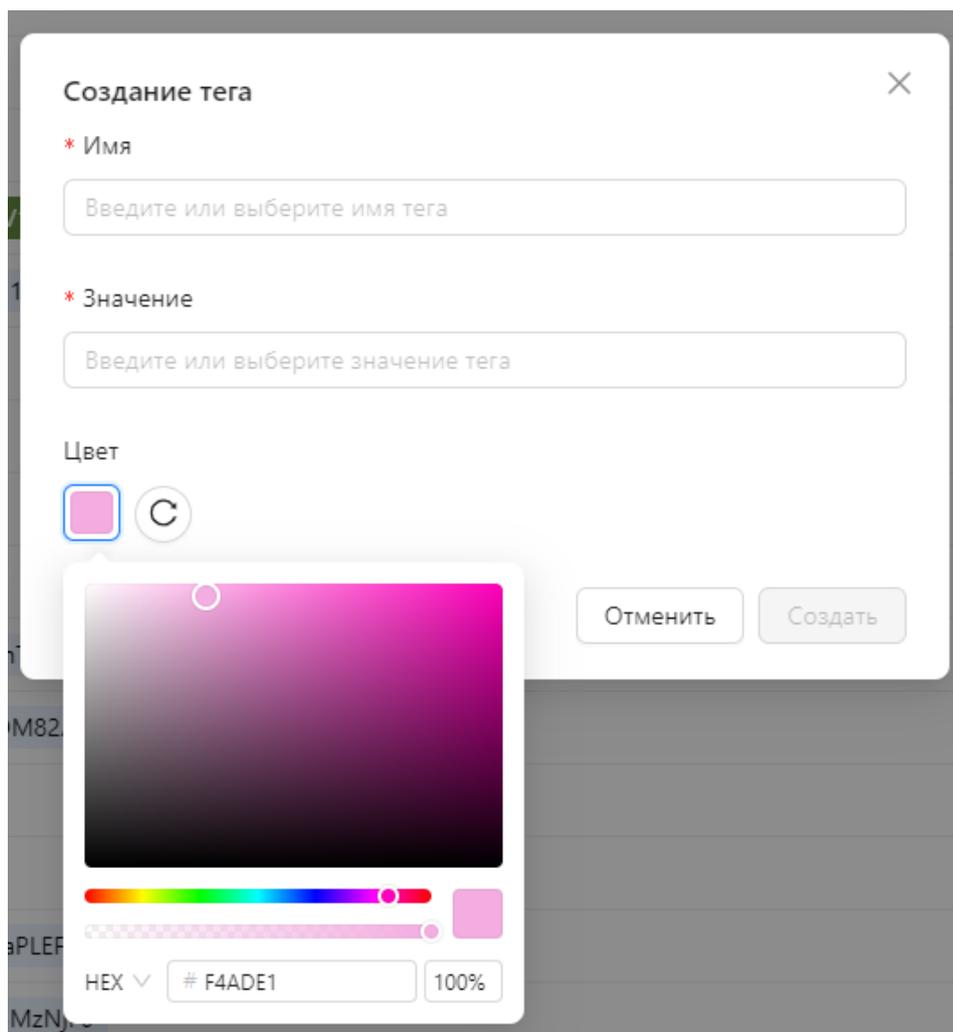
7.6.3 Дополнительные элементы

Над таблицей находятся интерактивные элементы, которые расширяют функциональность и возможности окна "Теги":



	Объект интерфейса	Значение
1	Кнопка создания тега	С помощью этой кнопки можно создать новый тег. По клику вызывается соответствующее модальное окно
2	Кнопка обновления данных	С помощью этой кнопки можно получить обновленные данные
3	Поиск по имени или значению тега	С помощью этой функции можно осуществлять поиск тегов на основе имени и значения тега. Пользователь может ввести ключевое слово, характеризующее имя или значение тега. В этом случае отобразятся только теги, соответствующие заданным поисковым критериям

Для каждого тега при создании, редактировании и клонировании можно указать название и выбрать произвольный цвет:



⚠ Ограничения

	Поле	Ограничение
1	Имя	Допустимое количество символов - 50. Имя может содержать только латинские буквы и цифры или символ _
2	Значение	Допустимое количество символов - 50. Значение может содержать только латинские буквы и цифры или символ _

	Поле	Ограничение
3	Цвет	нет ограничений