Руководство по установке (серверная часть) компонента Объединенный мониторинг Unimon (код компонента: MONA) продукта Platform V Monitor (код продукта: OPM)

ОГЛАВЛЕНИЕ

Руководство по установке серверной части

Здесь и далее поддерживаемой системой приложений-контейнеров является Kubernetes (использование OSE – опционально). В переменных, именах и параметрах системы могут встречаться названия систем контейнеризации, которые применимы для различных сред контейнеризации, указанных в Системных требованиях.

Термины и определения

Общие термины и определения, используемые в данном документе, представлены в общей документации продукта Platform V Monitor (OPM) в документе «Общее описание продукта Platform V Monitor (OPM)».

Состав дистрибутива

1. Дистрибутив с бинарными файлами

Компонент дистрибутива	**Описание**
./package/bh/	содержит бинарные файлы приложения
./package/conf/k8s/base/unimon-server/	содержит файлы с настройками для контейнеров Unimon-server
./package/conf/k8s/base/unimon-metadata/	содержит файлы с настройками для контейнеров Unimon-metadata
./package/conf/k8s/base/unimon-filter/	содержит файлы с настройками для контейнеров Unimon-filter

2. Дистрибутив с конфигурационными файлами:

Компонент дистрибутива	**Описание**
./package/conf/config/parameters/	содержит файлы с параметрами конфигурации с рекомендуемыми значениями
./package/conf/k8s/base/unimon-server/	содержит файлы для развертывания Unimon-server
./package/conf/k8s/overrides/openshift/unimon-server/	содержит файлы для развертывания Unimon-server в среде контейнеризации

Компонент дистрибутива	Описание
./package/conf/k8s/base/unimon-metadata/	содержит файлы для развертывания Unimon-metadata
./package/conf/k8s/overrides/openshift/unimon-metadata/	содержит файлы для развертывания Unimon-metadata в среде контейнеризации
./package/conf/k8s/base/unimon-filter/	содержит файлы для развертывания Unimon-filter
./package/conf/k8s/overrides/openshift/unimon-filter/	содержит файлы для развертывания Unimon-filter в среде контейнеризации
./package/conf/k8s/base/istio/	содержит файлы для развертывания и конфигурации Istio
./package/conf/k8s/overrides/openshift/istio/	содержит файлы для развертывания и конфигурации Istio
./package/conf/k8s/base/secrets/	содержит файлы с шаблонами для формирования секретов в среде контейнеризации

3. Дистрибутив с документацией:

Компонент дистрибутива	Описание
./documentation/documents	содержит файлы с документацией
./documentation/apis	содержит файлы с описанием API сервисов

Системные требования

Внешние сервисы и окружение

Сервис	Версия	Обязательность	Комментарий
OIDC провайдер / IAM Proxy/ LDAP	версии 4.2.0 и выше	Опционально	Требуется для выполнения аутентификации пользователей.

Platform V Аудит SE (AUD)	версия от 4.0 и выше	Опционально	Platform V Аудит SE (AUD) в prod-like окружениях является обязательным. Существует режим с записью событий аудита в логи.
Компонент Abyss (LGDB) в составе продукта Platform V Monitor (OPM)	версии 4.1 и выше	Опционально	Рекомендуемое хранилище телеметрических данных - Abyss. В случае отсутствия Abyss - сервис может записывать данные в Apache Kafka или в лог.
Компонент Indicator (INDA) в составе продукта Platform V Monitor (OPM)	версии 4.1 и выше	Опционально	Реализация UI компонента. Также сервис используется для отображения дашбородов системного и бизнес-мониторинга.
Компонент Журналирование (LOGA) в составе продукта Platform V Monitor (OPM)	версии 4.1 и выше	Опционально	Набор сервисов и инструментов для сбора и отображения в Indicator логов и цепочек вызовов.
Apache Kafka (рекомендуется Platform V Corax)	Версии 2.0 и выше	Обязательно	Для внутренних коммуникаций сервис использует Platform V Corax. ZooKeeper отдельно не указан, т.к. Platform V Corax без него невозможна.
СУБД PostrgeSQL (рекомендуется Platform V Pangolin SE (PSQ))	версии 4.4.x и выше	Обязательно	Для внутренних хранения метаинформации о процессах сбора данных и их конфигурации.
Среда контейнеризации Kubernetes / RedHat Open Shift	Kubernetes v 1.21; RedHat Open Shift 1.21.6+b4b4813 (Kubernetes version)	Обязательно	Сервис устанавливается в облачную инфраструктуру. Требуется наличие ISTIO.
Platform V One-Time-Token (OTT)	версии 4.0.1 и выше	Опционально	Сервис используется для выполнения различных проверок с аутентификацией OTT. Опционально в том случае, если такие проверки не требуются

Platform V Synapse Service Mesh (SSM) (ISTIO)	версии 2.8 и выше	Опционально	Сервис интеграции и оркестрации микросервисов в облаке
Компонент PACMAN (CFGA) в составе продукта Platform V Configuration (CFG)	версия 1.3	Опционально	Централизованный инструмент управления параметрами и конфигурациями
Java Virtual Machine JVM (Open JDK)	версия 11.x	Обязательно	
ОС Linux	версия от 8.2 и выше	Обязательно	Операционная система используемая, как базовый образ для контейнеров приложений и серверов. Рекомендуем ОС «Альт 8 СП»
Репозиторий	15.0	Опционально	VCS для хранения конфигураций, рекомендуем GitLab CE
Platform V DevOps Tools (DOT)	версии 1.1 и выше	Опционально	Инструмент автоматической установки

Пре-реквизиты и требования к окружению

1. Должны быть созданы сущности для хранения метрик в Abyss (проект, топик Kafka, задача для индексации).
2. Подготовить окружение - проверить наличие или создать:
   - Namespace с подключенным и настроенным проектом ISTIO.
3. Проверить в файле installer/system/efs/config/parameters/ssl.conf корректность заполнения параметров для сертификатов Istio.
4. Создать схему БД Postgres SQL или БД Platform V Pangolin SE (PSQ) для Unimon. В данном релизе имеется ограничение, при подключении к БД PostgresSQL / БД Platform V Pangolin SE (PSQ) для прогона liquibase-скриптов ssl не используется.
5. Сервис Unimon поддерживает гео-балансировку. Подробнее о настройке можно посмотреть в документации SberInfra, раздел Гео-балансировка.
6. Для применения конфигураций заданных через Pacman в namespace должен присутствовать Reloader.
7. Наличие Platform V IAM SE (IAM) для авторизации типа IAM или Abyss для авторизации типа PVM.

Одновременная установка клиентской и серверной части на текущий момент невозможна, необходимо выполнить отдельно установку серверной части, затем клиентской.

Установка

Установка дистрибутива производится с использованием одного из вариантов:

- **Ручная установка без использования скриптов** развертывания.
- **Ручная установка с использованием инструмента установки** Platform V Monitor (OPM), расположенного в дистрибутиве по пути package/bh/installer/deploy-pvm.zip (инструкция по его использованию находится внутри архива)

- **Автоматическая установка (опционально)** компонентом Deploy tools (CDJE) в составе Platform V DevOps Tools (DOT) версии не ниже release/D-01.039.049-886

Ручная установка без использования скриптов развертывания

Для клиентов, которые по каким-либо причинам не могут воспользоваться Pipeline для установки Unimon, доступна установка сервиса вручную:

1. Получить дистрибутив Unimon.
2. Выполнить скрипты для БД из дистрибутива.
3. Создать следующие служебные подключения для самомониторинга Unimon (с помощью UI Unimon для подключения или API Unimon):
   - Unimon-server;
   - Unimon-metadata;
   - Unimon-filter.
4. Создать секреты:
   - secret-opm-unimon-server;
   - unimon-db-certs;
   - secret-opm-unimon;
   - unimon-logger-kafka-certs - для интеграции с сервисом Журналирование;
   - unimon-ott-certs - если подразумевается взаимодействие с Platform V One-Time-Token (OTT). При установке вручную в именовании необходимо проставить версию .${distrib.release.version}. Пример: unimon-agent-config.r4.
5. Во всех файлах .yaml, расположенных в директориях Unimon-server, Unimon-metadata, Unimon-filter заполнить значения переменных.
   Список обязательных параметров указан ниже в пункте Установка дистрибутива централизованным Pipeline CD. Описание всех параметров собрано в Справочник конфигов.
6. Если namespace подключен к Istio и требуется выводить трафик через Deployment Egress необходимо:
   - создать секреты egressgateway-ca-certs и egressgateway-certs;
   - заполнить значения переменных для всех файлов .yaml из директории Istio.
7. Далее необходимо установить все заполненные файлы .yaml в среду контейнеризации с помощью команды

```
oc apply -f <имя_файла.yaml>
```

Автоматическая установка (опционально) компонентом Deploy tools (CDJE)

1. Выполнить миграцию конфигов:

- Выбрать SUBSYSTEM: OPM_UNIMON. Если выполняется одновременная установка Unimon с прикладным/технологическим сервисом, то в SUBSYSTEM нужно выбрать прикладной/технологический сервис.
- Выбрать COMPONENTS: Если выполняется одновременная установка Unimon с прикладным/технологическим сервисом, выбрать прикладной/технологический сервис и UNIMON (необходимы дополнительные настройки SUBSYSTEM, для отображения в списке второго компонента).
- Выбрать DISTRIB_VERSION: <последняя версия>.
- Выбрать кластер в OSE_CLUSTERS, в который необходимо установить Unimon.

- Выбрать версию платформы: branch R1.
- Выбрать плейбук MIGRATION_FP_CONF.
- Нажать кнопку «Собрать».

2. В репозитории конфигов установить корректные значения для параметров:

в файле conf/config/parameters/opm_unimon.all.conf должны быть обязательно заполнены параметры:

Параметр	Описание	Пример заполнения
unimon.custom.ose.deployment.spec.template.spec.containers.image.registry	Адрес docker registry	registry.sigma.sbrf.ru
unimon.ingress.endpoint.hostname	Хост ingress namespace, в котором установлен Unimon-server	unimon.ci02707148d43c3.apps.dev-gen.sigma.sbrf.ru
unimon.ingress.url	url ingress серверного namespace	http://unimon.ci02707148d43c3.apps.dev-gen.sigma.sbrf.ru
logger.labels.standid	ID стенда журналирования	DEV
audit.proxy.url	Адрес для подключения к Аудит	http://$:{}/
audit.proxy.host	Хост Аудит	ift.audit2-http-proxy-ott.apps.dev-gen.ca.sbrf.ru
audit.proxy.endpoint	Путь до endpoint Аудит	
audit.proxy.port	Порт Аудит	443
audit.enabled	Включение/выключение аудита событий	true

Следующие параметры opm_unimon.all.conf должны быть обязательно заполнены, если UNIMON_OTT_OSE_DEPLOY=true (включена установка Platform V One-Time-Token (OTT)), иначе оставить пустым:

Параметр	Описание	Пример заполнения
EGRESS_OTT_HOSTS	Адреса хостов Platform V One-Time-Token (OTT)	[XX.XX.XX.XXX, XX.XX.XX.XXX]
EGRESS_OTT_ENDPOINTS	Адреса хостов Platform V One-Time-Token (OTT)	[address: XX.XX.XX.XXX, address: XX.XX.XX.XXX]

unimon.ose.ott.module.id	Идентификатор модуля для сервиса Platform V One-Time-Token (OTT)	unimon_dev

Следующие параметры opm_unimon.all.conf должны быть обязательно заполнены, если UNIMON_LOGGER_PPRB_ENABLE=true (включена установка Журналирование (LOGA)), иначе оставить пустым:

Параметр	Описание	Пример заполнения
logger.pprb.host	Хост Журналирование	logger-endpoint-demo-http-ci00641491-idevgen-logger-dev.apps.dev-gen.ca.sbrf.ru
logger.pprb.endpoint	Endpoint Журналирование	/logger-endpoint/v1/events
logger.pprb.port	Порт Журналирование	80
logger.pprb.vs.port	Порт, на который выходит трафик из egress	00

Следующие параметры opm_unimon.all.conf должны быть обязательно заполнены, если UNIMON_LOGGER_EFS_ENABLE: true (включена установка Журналирование (LOGA)), иначе оставить пустым:

Параметр	Описание	Пример заполнения
logger.efs.kafka.bootstrap.servers	Список bootstrap - серверов Kafka для отправки логов	tkles-pprb00075.vm.esrt.cloud.sbrf.ru:9093,tkles-pprb00076.vm.esrt.cloud.sbrf.ru:9093
logger.efs.kafka.topic.name	Топик, в который будет осуществляется отправка логов	undefined

Следующие параметры opm_unimon.all.conf должны быть обязательно заполнены, если требуется геобалансировка, иначе оставить пустыми:

Параметр	Описание	Пример заполнения
istio.geo.route.balancer.host	FQDN геобалансировщика в unimon-geo-route.yaml(OSE)	unimon.dev-apps.ocp-geo.delta.sbrf.ru
istio.geo.ingress.balancer.host	FQDN геобалансировщика в ingress-geo.yaml(k8s)	unimon.dev-apps.ocp-geo.delta.sbrf.ru

в файле conf/config/parameters/opm_unimon.server.all.conf должны быть обязательно заполнены параметры: Параметры настройки авторизации, неотмеченные, как pvm, одинаковы для всех возможных сервисов авторизации.

Параметр	Описание	Пример заполнения
unimon.db.pool.hosts	Адреса пар хост+порт базы данных. Подробное описание ниже в п. Настройка соединения с БД PostreSQL или Platform V Pangolin SE (PSQ) (п.2)	psql-1.ca.345.ru:8101,psql-2.ca.345.ru:8102
unimon.db.name	Название базы данных	monaadm
unimon.db.schema	Схема базы данных	monadev2
unimon.db.ssl.enable	Включить SSL для подключения к базе данных	true
unimon.db.ssl.mode	Режим SSL при подключении к базе данных	verify-full
auth.iss-urls	Адреса ISS аутентификации	https://XX.XX.XXX.XX:XXXX/auth/realms/PlatformAuth
auth.jwks-url	Адрес получения jwk токена	http://$:{}/$
auth.jwks.port	Порт для адрес получения jwk токена	443
auth.jwks.host	Хост адреса получения jwk токена	all-sh-sudiwsa21u.ca.sbrf.ru
auth.jwks.endpoint	Путь до endpoint получения jwk токена	auth/realms/PlatformAuth/protocol/openid-connect/certs
auth.pvm.url	Адрес сервиса авторизации (только при auth.type=pvm)	http://$:{}/$
auth.pvm.port	Порт сервиса авторизации (только при auth.type=pvm)	443
auth.pvm.host	Хост адреса сервиса авторизации (только при auth.type=pvm)	tkles-pprb00075.vm.esrt.cloud.sbrf.ru

auth.pvm.endpoint	Путь до endpoint сервиса авторизации (только при auth.type=pvm)	v1

в файле conf/config/parameters/opm_unimon.unimon-filter.conf:

unimon-filter.unimonId=unimon-filter - параметр для указания ID подключения для метрик Unimon-filter

в файле conf/config/parameters/opm_unimon.unimon-metadata.conf:

unimon-metadata.unimonId=unimon-metadata - параметр для указания ID подключения для метрик Unimon-metadata

в файле conf/custom_property.conf.yml включить нужны параметры: Если требуется включить установку Platform V One-Time-Token (OTT), Журналирование (LOGA) (опциональная возможность включить один из сервисов журналирования) или сбор инфраструктурных метрик или установку открытого route, то необходимо в соответствующих параметрах указать значение true.. По умолчанию эта функциональность отключена.

UNIMON_OTT_OSE_DEPLOY: false # Включить установку Platform V One-Time-Token (OTT), если необходимо.
UNIMON_LOGGER_PPRB_ENABLE: false # Включить установку Журналирование LOGA при необходимости.
UNIMON_LOGGER_EFS_ENABLE: false # Включить установку Журналирование при необходимости.
UNIMON_FEDERATE_METRICS_ENABLE: false # Включить сбор инфраструктурных метрик(true), по умолчанию сбор инфраструктурных метрик отключен(false).
UNIMON_SENDER_HTTP_ROUTE_ENABLED: false # Включить установку открытого route для Unimon-sender, по умолчанию установка route отключена.
UNIMON_OSE_GEO_BALANCING: false # Включить установку Route в среде контейнеризации для геобалансировки
UNIMON_K8S_GEO_BALANCING: false # Включить установку Kubernetes Ingress для геобалансировки

в файле conf/config/parameters/opm_unimon.unimon-server.conf должны быть обязательно заполнены параметры:

Параметр	Описание	Пример заполнения
unimon-server.abyss.host	Адрес хоста Abyss	tkles-pprb00078.vm.esrt.cloud.sbrf.ru
unimon-server.abyss.port	Порт Abyss	443
abyss.endpoint	Путь координатора Abyss, прибавляемый к хосту	coordinator/api/gateway/v1

unimon-server.unimonId	Параметр для указания ID подключения для метрик Unimon-server	unimon-server

3. Добавить global в common репозиторий (значения задать в соответствии с текущим стендом):

_passwords.conf:

- пароли к хранилищам jks для ingress/egress:

```
ssl.ose.istio.keyStore.ingress.password=
ssl.ose.istio.keyStore.egress.password=
```

- пароли к остальным хранилищам jks, в особенности от jks для Kafka:

```
ssl.ose.keyStore.mq.password=
```

- пароль от ключа для клиентского сертификата для Kafka

```
rdkafka.ssl.key.password=
```

- добавить параметры для авторизации в Abyss:

```
unimon-server.abyss.password=<пользователь_abyss> unimon-
server.abyss.user=<пароль_abyss>
```

- добавить параметры для БД:

```
jdbc.unimon_postgres_liquibase.user=<пользователь_бд_с_правами_для_раскатки_liquibase_с
криптов>
jdbc.unimon_postgres_liquibase.password=<пароль_для_пользователя_бд_с_правами_для_рас
катки_liquibase_скриптов>
jdbc.unimon_postgres.user=<пользователь_бд_с_правами_для_работы_с_ac>
jdbc.unimon_postgres.password=<пароль_для_пользователя_бд_с_правами_для_работы_с_ac
> unimon.db.ssl.key.password=<пароль_для_приватного_ключа_бд>
```

- добавить параметры для авторизации в PVM сервисе авторизации:

```
unimon.auth.pvm.user=<пользователь_pvm_авторизации>
unimon.auth.pvm.password=<пароль_pvm_авторизации>
```

common.conf.yml:

- добавить параметры для подключения к БД:

```
UNIMON_POSTGRES_DB_URL:
"jdbc:postgresql://<хост_бд>:<порт_бд>/<пользователь_бд>" если используется pgbouncer
значение должно иметь вид
"jdbc:postgresql://<хост_бд>:<порт_бд>/<пользователь_бд>?prepareThreshold=0"
UNIMON_POSTGRES_DB_SCHEMA: "<схема_бд>"
```

ssl.conf:

- добавить параметры для Istio:

```
ssl.ose.istio.keyStore.ingress.CertAlias=
ssl.ose.istio.keyStore.ingress.KeyStoreFromFile=ansible/files/ssl/ingress.pacman-tst-keystore.jks
ssl.ose.istio.keyStore.egress.CertAlias=
ssl.ose.istio.keyStore.egress.KeyStoreFromFile=ansible/files/ssl/egress.pacman-tst-keystore.jks
ssl.ose.istio.keyStore.RootCertAlias=root
ssl.ose.istio.keyStore.ingress.password=ssl.ose.istio.keyStore.ingress.password
ssl.ose.istio.keyStore.egress.password=ssl.ose.istio.keyStore.egress.password
```

- добавить пароль от хранилища jks для сертификатов Kafka (а так же для остальных сертификатов, используемых в среде контейнеризации, кроме Istio и OTT):

```
ssl.ose.keyStore.mq.password=ssl.ose.keyStore.mq.password - данный пароль должен быть
заполнен в _passwords.conf среды
```

- добавить месторасположение хранилища jks для сертификатов Kafka (а так же для остальных сертификатов, используемых в среде контейнеризации, кроме Istio и OTT), а так же alias сетрификата для Kafka:

```
ssl.ose.keyStore.mq.keyStoreFromFile=ansible/files/ssl/kafka.jks ssl.ose.keyStore.mq.CertAlias=
```

- добавить месторасположение сертификатов для БД PostgreSQL или БД Platform V Pangolin SE (PSQ) :

```
ssl.ose.db.postrgesql.cert.location=ansible/files/ssl/pg-cert.pem
ssl.ose.db.postrgesql.cacert.location=ansible/files/ssl/pg-root.pem
ssl.ose.db.postrgesql.private.key.location=ansible/files/ssl/pg-key.pk8
```

openShift.conf:

- Параметры для Platform V One-Time-Token (OTT) (если используется, иначе оставить пустым)

```
global.ott.service.url=https://stub-host:stub-port/ott-service/rest/token
global.ott.grpc.port=/mnt/ott-uds-socket/ott.socket
```

```
global.ott.service.hosts="XX.XX.X.XX:XXXX,XX.XXX.XXX.XXX:XXXX"
global.ott.store.type=JKS
```

- Параметры для Logger

```
global.platform.logger.kafka.bootstrap.servers=${global.unimon-sender.kafka.bootstrap_servers}
global.platform.logger.kafka.topic=unimon.sender
global.platform.logger.kafka.security.protocol=SSL
```

- Параметры для Kafka

```
global.unimon-sender.kafka.bootstrap_servers=tkles-pprb00075.vm.esrt.cloud.sbrf.ru:9093,tkles-pprb00076.vm.esrt.cloud.sbrf.ru:9093,tkles-pprb00077.vm.esrt.cloud.sbrf.ru:9093
global.platform.ose.kafka.ports=9093 (указать только одно значение, перечисление недопустимо) global.unimon-sender.kafka.default.topic=unimon.Tengri global.unimon-sender.kafka.ssl.enabled=true
```

- Параметр для указания адреса Kubernetes

```
global.unimon-agent.kubernetes.address=XXX.XX.X.X
```

- Параметр для задания процента минимально доступных подов для конфигурации PodDistruptionBudget

```
global.common.poddisruptionbudget.minAvailable=50%
```

4. Выполнить раскатку скриптов базы данных и Ingress/egress:
- Выбрать SUBSYSTEM: OPM_UNIMON
- Выбрать DISTRIB_VERSION: <последняя версия>
- Выбрать кластер в OSE_CLUSTERS, в который необходимо установить Unimon
- Выбрать версию платформы: branch R1
- Выбрать плейбуки DB_UPDATE, OPENSHIFT_INGRESS_EGRESS_DEPLOY
- Нажать кнопку "Build"
5. Создать следующие служебные подключения для самомониторинга Unimon (с помощью UI Unimon для подключения или API Unimon):
- Unimon-server;
- Unimon-metadata;
- Unimon-filter.

6. Выполнить установке дистрибутива Unimon в среде контейнеризации:
- Выбрать SUBSYSTEM: OPM_UNIMON
- Выбрать DISTRIB_VERSION: <последняя версия>
- Выбрать кластер в OSE_CLUSTERS, в который необходимо установить Unimon
- Выбрать версию платформы: branch R1
- Выбрать плейбуки OPENSHIFT_DEPLOY

- Нажать кнопку "Build"

После установки Unimon, необходимо проверить наличие всех необходимых ServiceEntry. При отсутствии какого-либо - добавить вручную - egress-se-http-logger.yaml - egress-se-tcp-abyss.yaml - egress-se-tcp-kafka.yaml - egress-se-tcp-ott.yaml - egress-se-tcp-postgresql.yaml - egress-se-http-audit.yaml - egress-se-http-auth.yaml - egress-se-http-authorization.yaml

Примечание: если сервис платформенного журналирования (Журналирование) не используются, egress-se-http-logger.yaml вручную устанавливать не следует

В состав дистрибутива входят конфигурационные файлы с рекомендуемыми значениями не стендозависимых параметров для настройки продукта, их изменение может нарушить безопасность продукта.

Обновление

Для Unimon доступно бесшовное обновление, то есть без остановки сервиса. Для обновления необходимо установить Unimon в тот же namespace, где установлена предыдущая версия (не ниже R3.2.1).

Если версия, которую вы собираетесь обновить ниже R3.2.1 необходимо предварительно выполнить удаление ресурсов Unimon (с помощью DELETE на каждом ресурсе или командой в терминале: oc delete all,se,vs,dr -l CHANNEL=unimon):

Deployment
Service
DestinationRule
EnvoyFilter
ServiceEntry
VirtualService
Gateway
Route

Для обновления Unimon с версии R3.2 на 4.0 необходимо сначала удалить Istio манифесты вручную, затем выполнить обновление. При последующих обновлениях - дополнительных действий не требуется. Необходимые настройки сервиса будут осуществлены при установке дистрибутива через Jenkins.

Информация об изменении конфигурационных параметров с предыдущей версии
opm_unimon.server.all.conf

удалены: rn.default= изменены значения: unimon.db.pool.hosts=none unimon.db.name=none unimon.db.schema=none auth.jwks.host=undefined auth.jwks.port=undefined auth.jwks.endpoint=none auth.iss-urls=none auth.pvm.port=none auth.pvm.endpoint=none auth.pvm.host=none

opm_unimon.unimon-filter.conf

новые параметры: unimon-filter.ose.poddisruptionbudget.spec.minAvailable=${global.common.poddisruptionbudget.minAvailable}

opm_unimon.unimon-metadata.conf

новые параметры: unimon-metadata.ose.poddisruptionbudget.spec.minAvailable=${global.common.poddisruptionbudget.minAvailable}

opm_unimon.unimon-server.conf

изменены значения: unimon-server.ose.poddisruptionbudget.spec.minAvailable=${global.common.poddisruptionbudget.minAvailable}

opm_unimon.all.conf

изменены значения: logger.pprb.host=none logger.pprb.endpoint=none logger.pprb.port=none EGRESS_OTT_HOSTS=none EGRESS_OTT_ENDPOINTS=none новые параметры: unimon.ose.istio.envoy-image=${global.registry.url}/${global.envoy.image.registry.path}/proxyv2-rhel8@sha256:093f3b36b3977225fe11ba3e0cb30316243c86be1ae0b05cc51b849e8dd1e74e fluent-bit-sidecar.ose.deployment.spec.template.spec.containers.image.registry=${global.registry.url}/${fluent-bit-sidecar.ose.deployment.spec.template.spec.containers.image.registryPath}/fluent-bit@sha256:04da83ed2af92600f7e0b4055d707c038c6e549ed14dba6372b0a2a50ddae32d fluent-bit-sidecar.ose.deployment.spec.template.spec.containers.image.registryPath=efs/ci01976100/ci02698091_ulogger ott-sidecar.ose.deployment.spec.template.spec.containers.ott-sidecar.image=${global.registry.url}/${global.ott.agent.image.registry.path}/ott/ott-client-api-v2@sha256:05b3d2530abdb9e27b6862793a939c87c3a72d09dde6656c4f192f6d408d7088 #флаг поставки в Гостех, предполагаемое значение: unimon.common.solution=gostech unimon.common.solution= удалено: istio-envoy-image=${global.ufs.synapse.envoy.image} logger.fluent-bit.docker.image.path=efs/ci01976100/ci02698091_ulogger/fluent-bit@sha256:04da83ed2af92600f7e0b4055d707c038c6e549ed14dba6372b0a2a50ddae32d ott.docker.image.path=pprb/ci00641491/ci01125613_ott/ott-client-api-v2@sha256:05b3d2530abdb9e27b6862793a939c87c3a72d09dde6656c4f192f6d408d7088

Использование балансировщика для HTTP сервисов

Если два или более внешних http сервиса находятся за одним балансировщиком, например: Abyss и сервис авторизации находятся за балансировщиком. В этом случае необходимо использовать настройку ресурсов Istio в режиме http балансировщика, чтобы при деплое корректно пройти валидатор Platform V Synapse Service Mesh (SSM). Для этого необходимо в репозитории конфигов АС в файле custom_property.conf.yml описать конфигурацию http сервисов находящихся за балансировщиком:

1. Активируем режим создания DSR/SE/VS ресурсов Istio под балансировщик.

Для этого должна быть заполнена переменная UNIMON_USE_SERVICE_TO_BALANCER в custom_property.conf.yml

```
UNIMON_USE_SERVICE_TO_BALANCER: [{service: "abyss", gateway_port: 10081, internal_egress_port: 7072, destination_port: 443},{service: "auth_pvm", gateway_port: 10084, internal_egress_port: 7079, destination_port: 8080}]
```

Пример для ситуации, когда Abyss и сервис авторизации находятся за балансировщиком. Для сервиса Abyss на балансировщике открыт порт 443, для авторизации на балансировщике порт 8080. В переменную передаем словарь, в каждой секции которого имя сервиса находящегося за балансировщиком, порт сервиса для маркировки на Istio Platform V Synapse Service Mesh (SSM), порт на egress, порт на балансировщике.

Например для: [{service: "abyss", gateway_port: 10081, internal_egress_port: 7072, destination_port: 443}]

```
service: "abyss"  Описание сервиса (произвольное имя для понимания какой сервис пробрасываем, можно: ABYSS и т.д. в латинице)
```

```
gateway_port: 10081  Порт сервиса для маркировки трафика в VirtualService.  Следует использовать значения из переменных в custom_property.conf.yml: UNIMON_ABYSS_GATEWAY_PORT: 10081 UNIMON_AUDIT_GATEWAY_PORT: 10082 UNIMON_AUTH_JWKS_GATEWAY_PORT: 10083 UNIMON_AUTH_PVM_GATEWAY_PORT: 10084 Рекомендация не менять без необходимости данные параметры, если есть необходимость сменить порт то придерживаться правила: порт не должен пересекаться с уже активными портами на egress.
```

```
internal_egress_port: 7072 Порт который необходим открыть на egress gateway, необходимо чтобы трафик гарантированно прошел через egress gateway.  Использовать только следующие порты: internal_egress_port: 7072 для сервиса ABYSS internal_egress_port: 7073 для сервиса аудита internal_egress_port: 7077 для сервиса аутентификации internal_egress_port: 7079 для сервиса авторизации
```

```
destination_port: 443 Порт на балансировщике за которым будет доступен пробрасываемый сервис.
```

Перечисленные параметры в словаре описывают движение трафика из приложения: приложение обращается в Platform V Synapse Service Mesh (SSM) (ISTIO) на gateway_port: 10081, далее трафик проходит egress gateway (internal_egress_port: 7072) и уходит на балансировщик (destination_port: 443).

2. Hostname балансировщика

Обязательно указываем hostname балансировщика в custom_property.conf.yml, например:

```
UNIMON_BALANCER_HOSTNAME: 00021.xx.yyyy.cloud.sbrf.ru
```

3. Вариативность настройки.

На балансировщике используются 4 разные порта, всегда на каждый сервис свой порт, пример для всех сервисов (Abyss, Аудит, аутентификация, авторизация):

```
UNIMON_USE_SERVICE_TO_BALANCER: [{service: "abyss", gateway_port: 10081,
internal_egress_port: 7072, destination_port: 443},{service: "audit", gateway_port: 10082,
internal_egress_port: 7073, destination_port: 3443},{service: "auth_jwks", gateway_port: 10083,
internal_egress_port: 7077, destination_port: 4443},{service: "auth_pvm", gateway_port: 10084,
internal_egress_port: 7079, destination_port: 8080}]
```

Возможна ситуация, когда за балансировщиком только 2-3 сервиса из возможных 4-х и на каждый сервис свой порт, в этом случае из примера просто удалите лишний сервис

На балансировщиике используется один порт для всех сервисов. За балансировщиком 4 сервиса например: Abyss, аудит, аутентификация, авторизация:

```
UNIMON_USE_SERVICE_TO_BALANCER: [{service: "abyss", gateway_port: 10081,
internal_egress_port: 7072, destination_port: 443},{service: "audit", gateway_port: 10082,
internal_egress_port: 7073, destination_port: 443},{service: "auth_jwks", gateway_port: 10083,
internal_egress_port: 7077, destination_port: 443},{service: "auth_pvm", gateway_port: 10084,
internal_egress_port: 7079, destination_port: 443}]
```

Возможна ситуация, когда за балансировщиком только 2-3 сервиса, в этом случае удаляем лишние сервисы из примера.

На балансировщике используется вариация из двух предыдущих пунктов, например Abyss, Аудит, аутентификация используют порт 443, а на сервисе авторизации порт 8080:

```
UNIMON_USE_SERVICE_TO_BALANCER: [{service: "abyss", gateway_port: 10081,
internal_egress_port: 7072, destination_port: 443},{service: "audit", gateway_port: 10082,
internal_egress_port: 7073, destination_port: 443},{service: "auth_jwks", gateway_port: 10083,
internal_egress_port: 7077, destination_port: 443},{service: "auth_pvm", gateway_port: 10084,
internal_egress_port: 7079, destination_port: 8080}]
```

Правила составления словаря все те же.

4. Переход на конфигурацию с использованием балансировщика.

4.1. Обязательно перед обновлением удалить из namespace конфиги DestinationRule/ServiceEntry/VirtualSevice для сервисов которые нужно перенастроить в режим, когда сервис находится за балансировщиком.

4.2. Обязательно перед обновлением выполнить миграцию конфигов АС, затем проверить что после миграции в конфигурации АС присутствуют параметры:

```
unimon-server.abyss.url=http://${unimon-server.abyss.host}:{{
UNIMON_ABYSS_GATEWAY_PORT }}/${abyss.endpoint}
audit.proxy.url=http://${audit.proxy.host}:{{ UNIMON_AUDIT_GATEWAY_PORT
}}/${audit.proxy.endpoint} auth.iss-urls=http://${auth.jwks.host}:{{
UNIMON_AUTH_JWKS_GATEWAY_PORT }} auth.jwks-url=http://${auth.jwks.host}:{{
UNIMON_AUTH_JWKS_GATEWAY_PORT }}/${auth.jwks.endpoint}
auth.pvm.url=http://${auth.pvm.host}:{{ UNIMON_AUTH_PVM_GATEWAY_PORT
}}/${auth.pvm.endpoint}
```

Если переменные отсутствуют, либо отличаются - привести параметры к указанному виду.

Данные переменные должны формироваться динамически на этапе параметризации конфигов при запуске деплоя, потому крайне нежелательны изменения в логике формирования переменной (приведет к не очевидным ошибкам работы)

Возможно придется заполнить, проверить, дополнить значения в переменных:

```
abyss.endpoint auth.jwks.endpoint auth.pvm.endpoint
```

Также не забываем, что должны быть обязательно заполнены:

```
unimon-server.abyss.host unimon-server.abyss.port auth.jwks.host auth.jwks.port auth.pvm.host
auth.pvm.port
```

И если используется аудит:

```
audit.proxy.host audit.proxy.port
```

Настройка соединения с БД PostgreSQL или БД Platform V Pangolin SE (PSQ)

Для того чтобы обеспечить работу сервиса как с одной, так и с несколькими базами данных, необходима некоторая настройка на стороне клиента. О настройке сертификатов для взаимодействия серверной части Unimon с БД PostgreSQL подробно описано в Руководстве по безопасности. За управление включения/выключения пула коннектов отвечает параметр unimon.datasource.pool.enabled.

- при unimon.datasource.pool.enabled=true - включить клиентский пул соединений Hikari, используется только в DEV-режиме, ssl при этом выключен, pgbouncer не используется
- при unimon.datasource.pool.enabled=false - выключить клиентский пул соединений Hikari, включить использование pgbouncer, есть возможность настройки SSL

Далее требуется внимательная настройка путей прохождения трафика для корректного создания Pipeline сущностей Istio (VirtualSevice, ServiceEntry, GateWay, DestinationRule, Service).

1. Создание словарей с хостами БД

Задается значение параметра UNIMON_DB_CLUSTER в custom_property.conf.yml - список серверов в кластере БД, минимальное значение: один сервер. Для каждого сервера в кластере БД описываем путь прохождения трафика, где:

- host - hostname сервера,
- gateway_port - порт для всех запросов из pod на сервер БД
- internal_egress_port - внутренний порт на Egress,
- destination_port - порт на сервере БД

gateway_port и internal_egress_port должны быть уникальны и не пересекаться с уже используемыми портами на egress.

Пример:

```
UNIMON_DB_CLUSTER: [{host: "psql-1.ca.345.ru", gateway_port: 8101, internal_egress_port: 7101, destination_port: 6543},{host: "psql-2.ca.345.ru", gateway_port: 8102, internal_egress_port: 7102, destination_port: 6545},{host: "psql-3.ca.345.ru", gateway_port: 8103, internal_egress_port: 7103, destination_port: 6543}.{host: "psql-4.ca.345.ru", gateway_port: 8104, internal_egress_port: 7104, destination_port: 6543}]
```

В соответсвии с тем, что указано в словаре, будут созданы/обновлены сущности Istio для настройки прохождения трафика в базу данных.

2. Заполнение значений конфигурации БД

1. unimon.db.pool.hosts

Заполнить значениями из словаря UNIMON_DB_CLUSTER в custom_property.conf.yml в формате host:gateway_port. Например, описано три хоста postgresql в разных регионах:

```
UNIMON_DB_CLUSTER: [ {host: "psql-1.ca.345.ru", gateway_port: 8101, internal_egress_port: 7101, destination_port: 6543}, {host: "psql-2.ca.345.ru", gateway_port: 8102, internal_egress_port: 7102, destination_port: 6545}, {host: "psql-3.ca.345.ru", gateway_port: 8103, internal_egress_port: 7103, destination_port: 6543}]
```

Соответственно, строка подключения будет содержать 3 хоста и каждому хосту соответствует свой gateway_port (порт для маркировки трафика из пода на БД)

```
unimon.db.hosts=psql-1.ca.345.ru:8101,psql-2.ca.345.ru:8102,psql-3.ca.345.ru:8103
```

2. unimon.db.url

Подключение к БД не изменять, так как используется динамическое формирование из переменных unimon.db.pool.hosts и unimon.db.name)

```
unimon.db.url=jdbc:postgresql://${unimon.db.pool.hosts}/${unimon.db.name}?targetServerType=master&prepareThreshold=0
```

3. unimon.db.name, unimon.db.schema - заполнять в соответствии с настройками стенда.

3. Создание словарей с хостами IAM

Задается значение параметра UNIMON_IAM_BACKENDS_BEHIND_PROXY в custom_property.conf.yml - список серверов IAM, минимальное значение: один сервер. Для каждого сервера в кластере БД описываем путь прохождения трафика, где:

- host - hostname сервера,
- gateway_port - порт для всех запросов из pod на сервер,
- internal_egress_port - внутренний порт на Egress,
- destination_port - целевой порт на сервере.

gateway_port и internal_egress_port должны быть уникальны и не пересекаться с уже используемыми портами на egress.

Пример:

```
UNIMON_IAM_BACKENDS_BEHIND_PROXY: [ {host: 'iam-backend-1.domain.sbrf', gateway_port: 11001, internal_egress_port: 7771, destination_port: 443 }, {host: 'iam-backend-2.domain.sbrf', gateway_port: 11002, internal_egress_port: 7772, destination_port: 443 },  {host: 'iam-backend-3.domain.sbrf', gateway_port: 11003, internal_egress_port: 7773, destination_port: 443 }  ]
```

Настройка взаимодействия с Indicator (если требуется UI Unimon)

- в Indicator прописать URL Ingress Unimon-server (подробнее в документации Indicator - Настройка Proxy Unimon);
- в Indicator создать datasource с доступом к endpoint Unimon-server (подробнее в документации Indicator - Настройка Proxy Unimon);
- в Nginx Abyss прописать endpoint Unimon-server (подробнее в документации по установке Abyss)

Проверка работоспособности

1. Pod Unimon-server, Unimon-metadata, Unimon-filter работают успешно.
2. Отсутствуют сообщения об ошибках в логах Unimon-server, Unimon-metadata, Unimon-filter (примеры ошибок указаны в Руководстве по системному администрированию).
3. Выполнить проверку взаимодействия Unimon-server и Unimon-sender (проверить логи на наличие ошибок о подключении к Unimon-server).

Откат

1. Откат к предыдущей версии Unimon представляет собой установку последней стабильной версии и удаление предыдущей.
2. Для отката необходимо выполнить установку предыдущей стабильной версии в соответствии с инструкцией по установке Unimon.

Часто встречающиеся проблемы и пути их устранения

1. Ошибка выгрузки образа из registry. Данная ошибка возникает при отсутствии или некорректном секрете для выгрузки образа из regitry, для устранения необходимо исправить или пересоздать секрет.
2. Ошибка исчерпания ресурсов namespace. Данная ошибка возникает при отсутствии необходимого количества ресурсов для развертывания сервиса, для устранения необходимо удалить лишние сущности из namespace, либо расширить квоты.
3. Ошибка отправки логов в сервис Журналирование. Данная ошибка возникает при отсутствии, либо некорректном содержимом DestinationRule, ServiceEntry, VirtualService с

именем egressgateway-logger-monitoring-r*, для устранения необходимо исправить или пересоздать.

4. Ошибка при запуске Unimon-server - не стартует application context. Данная ошибка может возникать из-за отсутствия SE, либо невозможно подключиться к БД . Для устранения необходимо создать SE, либо проверить корректность подключения к БД.
5. Ошибки, связанные с SSL на шаге DB_UPDATE при запуске pipline установки. Данная ошибка возникает, когда север Postgres может работать только с SSL. Накат скриптов происходит только по незашифрованному соединению (см. Системные требования п.5). Необходимо исключить использование SSL для наката скриптов.
6. Ошибка при подключении к БД "ERROR: unsupported startup parameter: search_path". Для устранения необходимо в настройке pgbouncer, в параметр ignore_startup_parameters добавить search_path, пример: ignore_startup_parameters = extra_float_digits*,search_path*

Чек-лист валидации установки

Чтобы убедиться, что Unimon работает корректно:

1. Проверить, что для проекта в среде контейнеризации создались объекты приложений на основании файлов конфигурации: DeploymentConfig, ConfigMap, Service, Route.
2. Проверить, что для проекта в среде контейнеризации создались объекты istio/ingress/egress на основании файлов конфигурации: Deployment, VirtualService, ServiceEntry, DestinationRule, Gateway, ConfigMap, EnvoyFilter.
3. Проверить логи на наличие ошибок.