ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ»

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ «МОЙОФИС ЗАЩИЩЁННОЕ ОБЛАКО»

РУКОВОДСТВО АДМИНИСТРАТОРА ПО БЕЗОПАСНОСТИ

RU.2914487.501490.001 99

На 28 листах

Москва 2022

АННОТАЦИЯ

Данный документ является руководством администратора по безопасности «МойОфис Защищённое Облако» (далее по тексту — изделие, МойОфис) и предназначено для предоставления рекомендаций по безопасной конфигурации окружения и продукта МойОфис Защищенное Облако.

Следующие общие принципы должны учитываться при внесении изменений в конфигурацию продукта:

- Применение средств защиты. Средства защиты, в том числе сторонние, должны применяться и корректно конфигурироваться.
- Криптографическая защита данных. Данные пересылаемые по сети во внутреннем и внешнем периметре, используя проводное и беспроводное соединение, подвержены угрозе нарушения конфиденциальности. В случае наличия практически применимых средств данные должны быть зашифрованы.
- Минимизация ПО. Наиболее простой путь для ограничения векторов атаки на программное обеспечение - это применение только необходимых программ.
- Разграничение доступа для сетевых сервисов. Необходимо выполнять конфигурацию средств защиты от несанкционированного доступа или иным образом разграничивать независимые сетевые службы, там где это возможно, для того чтобы ограничить количество служб, которые могут быть скомпрометированы в случае успешной атаки на одну из служб.
- Назначение наименьших привилегий. Необходимо назначать наименьшие привилегии для пользователей и программного обеспечения.
- Проверка в тестовом окружении. Применимость руководства на продуктивных окружениях должна предварительно проверяться в тестовой среде.

СОДЕРЖАНИЕ

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

Перечень сокращений, терминов и определений приведен в таблице 1.

Таблица 1 – Сокращения и расшифровки

Сокращение	Расшифровка
ОС	Операционная система
ПО МойОфис	Программное обеспечение МойОфис Защищенное Облако
МойОфис Администрирование	Административная панель ПО МойОфис
Тенант	Виртуальный экземпляр продукта в рамках одной инсталляции

1 ОБЩИЕ СВЕДЕНИЯ

1.1 Назначение

ПО «МойОфис Защищенное Облако» – продукт для организации виртуальной рабочей среды в государственных организациях и крупных коммерческих предприятиях. Включает редакторы текста, таблиц, презентаций. В состав продукта входят сервер совместной работы, система хранения данных и информационно–коммуникационный сервис Логос.

МойОфис Документы – это продукт для совместного редактирования текстовых документов, электронных таблиц и графических презентаций. В продукт входят редакторы Текст, Таблица и Презентация, созданные на базе единого программного ядра, а также средство доступа к документам, хранящимся во внешнем облаке или на серверах самой компании. Доступ к документам и совместная работа над ними возможна с любого устройства.

МойОфис Текст обеспечивает удобное и быстрое создание документов с использованием шаблонов, стилей и средств форматирования текста. Функции совместного редактирования обеспечивают эффективную совместную работу сотрудников.

МойОфис Таблица – это приложение для быстрой и удобной работы с электронными таблицами и анализа данных. Продукт поддерживает расширенный набор формул и средств для обработки данных. Совместное редактирование на любой из поддерживаемых платформ обеспечивает быстрый анализ и подготовку документов группой сотрудников.

МойОфис Презентация – приложение с полным набором инструментов для просмотра графических презентаций.

МойОфис Логос – информационно-коммуникационный сервис быстрого обмена сообщениями, документами и файлами внутри организации или предприятия. С его помощью сотрудники могут общаться друг с другом, участвовать в групповых дискуссиях и проводить аудио- и видеоконференции. МойОфис Логос интегрирован с редакторами МойОфис, что позволяет редактировать и обсуждать документы в режиме одного окна.

Административная панель (МойОфис Администрирование) МойОфис Защищенное Облако предназначена для заполнения профайла организации, управления и ведения

пользователями и их группами, настройки используемых доменов, выполнения восстановления пользовательских файлов.

1.2 Системные требования

ПО «МойОфис Защищенное Облако» поставляется в виде набора пакетов. Такой способ распространения выбран для упрощения процедуры установки изделия средствами операционной системы.

1.2.1 Требования к программному обеспечению

Изделие функционирует в среде операционных систем Astra Linux Special Edition 1.6, Astra Linux Special Edition 1.7

1.2.2 Требования к аппаратному обеспечению

Штатная работа серверных компонентов изделия возможна на оборудовании, удовлетворяющем требованиям, которые приведены в таблице 2. В качестве АРМ пользователя с web-клиентом может использоваться любой персональный компьютер, соответствующий требованиям операционной системы. Пример расчета параметров сервера приведен ниже. Для точного расчета необходимых ресурсов рекомендуется обращение в службу технической поддержки ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ».

Таблица 2 - Минимальные требования к оборудованию для работы серверных компонентов

Параметр	**Характеристики**
Процессор	четырёхядерный, x86_64-совместимый, с частотой не менее 2 ГГц
Оперативная память	не менее 16 Гбайт
Жёсткий диск	50 Гбайт для установки компонентов; дополнительно по 2 Гбайт на каждого пользователя системы
Видеоадаптер	любой с поддержкой современных графических режимов
Сетевой адаптер	Gigabit Ethernet (1000BASE-T)
Дополнительно	клавиатура, манипулятор типа «мышь», устройство для чтения DVD

В таблице 3 приведен расчет установки МойОфис Защищенное Облако на 100 пользователей для операционной системы Astra Linux Special Edition версий 1.6 и 1.7.

Таблица 3 - Расчет установки МойОфис Защищенное Облако на 100 пользователей

Роль	**Количество ядер, шт**	**Оперативная память, Гб**	**Жесткий диск основной, Гб**	**Жесткий диск дополнительный, Гб**
PGS	40	20	100	1000 (Опционально)
CO	40	40	100	-
Logos	20	20	200	-

Процессы установки и настройки ПО МойОфис Защищенное облако приведены в документе «ПО МойОфис Защищенное облако. Руководство администратора» RU.2914487.501490.001 98.

2 ИДЕНТИФИКАЦИЯ И АУТЕНТИФИКАЦИЯ

2.1 Идентификация и аутентификация пользователей

Идентификация и аутентификация пользователей и администраторов реализована на страницах авторизации https://auth.DOMAIN/auth и https://admin.DOMAIN/ соответственно.

При успешной аутентификации, осуществляется авторизация пользователя, отображается страница (https://auth.DOMAIN/landing) со списком доступных приложений «Мой Офис»: «Документы» и «Логос». В ”Логос” может осуществляться отдельный вход: если осуществлена авторизация в auth, то отдельной авторизации в Логос не требуется. Из любого приложения можно перейти в Логос без повторной организации. Если осуществлен отдельный вход в ”Логос” (не через auth), то переход из ”Логос” в другие приложения невозможен без авторизации в auth).

Пользователям, имеющим права на администрирование помимо основных приложений доступно средство администрирования.

После авторизации на главной странице, пользователь может войти в любое из приложений, не проходя повторную аутентификацию (реализована технология single signon).

Отдельно осуществляется вход в средство администрирования: при переходе по ссылке на вход в средство администрирования открывается страница входа административного интерфейса https://admin.DOMAIN/, где необходимо ввести логин и пароль.

Требования: ИАФ.1

2.2 Управление идентификаторами

Идентификатор пользователя — представляет собой значение, однозначно определяющее пользователя в системе.

Исключение повторного использование идентификатора пользователя контролируется администратором безопасности.

Требования: ИАФ.3, ИАФ.3-У1б

2.2.1 Неиспользуемые идентификаторы

Рекомендуется блокировать идентификаторы пользователей через период времени неиспользования. Эта операция осуществляется скриптом `/etc/cron.daily/block_inactive_users`, который автоматически запускается каждый день и устанавливает статус `enabled: false` учетным записям, неактивным в течение 45 дней.

Требования: ИАФ.3-У2б

2.3 Управление средствами аутентификации

Настройка парольной политики позволяет обеспечить использование надёжных, устойчивых ко взлому паролей. Парольные политики применяются ко всем новым создаваемым пользователям, а также во время смены пароля пользователя.

Требования: ИАФ.4

2.3.1 Время блокировки

Параметр задается в интерфейсе Keycloak во вкладке Realm Settings/Security Defenses/Brute Force Detection. Время блокировки рассчитывается как произведение значений параметров Wait Increment * Max Login Failures, но результат не превышает параметр Max Wait.

2.3.2 Максимальное количество неуспешных попыток

Параметр Max Login Failures задается в интерфейсе Keycloak во вкладке Realm Settings/Security Defenses/Brute Force Detection.

2.3.3 Минимальная длина пароля

Минимальная рекомендуемая длина пароля составляет восемь символов. Для настройки этого параметра необходимо на сервере с ролью PGS выполнить команды:

```
etcdctl set /pgs/password_settings/min_length 8
/opt/co/co_starter.sh stop PGS
/opt/co/co_starter.sh start PGS
```

2.3.4 Минимальная размерность алфавита пароля

Мощность алфавита пароля или, другими словами, количество доступных для составления пароля символов достаточно сильно влияет на его устойчивость ко взлому при помощи автоматизированных средств подбора. Рекомендуется использовать алфавит размером не менее 70 символов.

Применяется следующий алфавит: [a–z]U[A–Z]U[0–9]U{! @ # $ % ^& * { }[ ] ( ) / \{} ’ ” ‘ ~, ; : . < > + = – _}U {0x20}.

2.3.5 Срок действия пароля

Администратор безопасности производит сброс паролей пользователей не более чем через 60 дней.

Требования: ИАФ.4-У1г

2.4 Защита обратной связи

Символы в поле ввода пароля маскируются при помощи специальных символов (*).

Маскирование пароля в поле для ввода позволяет предотвратить возможность его ”визуальной компрометации”.

Требования: ИАФ.5

3 УПРАВЛЕНИЕ ДОСТУПОМ

3.1 Управление учетными записями

Механизм разграничения доступа поддерживает следующие типы учетных записей:

- ”владелец компании” – был создан при развертывании;
- ”администратор компании” – в ”Администрировании” пользователю была присвоена роль ”Администратор”;
- ”пользователь” – в ”Администрировании” пользователю была присвоена роль ”Пользователь”.

Изделие предоставляет автоматизированные средства управления учетными записями.

1) Консольный интерфейс администратора

2) Административный web-интерфейс, который доступен по адресу https://admin.DOMAIN.

Блокирование учетной записи осуществляется при блокировании соответствующего идентификатора пользователя.

Требования: УПД.1, УПД.1-У1, УПД.1-У2, УПД.1-У3б

3.2 Правила разграничения доступа

Правила разграничения доступа реализуются на основе механизма реализации списков доступа субъектов к объектам и обеспечивают управление доступом внешних пользователей при входе в информационную систему и разграничение доступа субъектов к объектам, создаваемым прикладным и специальным программным обеспечением.

Владелец объекта - пользователь, которому принадлежит объект. Владелец может предоставить совместный доступ к объекту другим пользователям, предоставляя им те или иные права доступа. Если к папке предоставлен совместный доступ, то он распространяется на все дочерние объекты папки строгим наследованием прав (дочерние объекты расшариваются с правами, идентичными правам родительского объекта). Если владелец объекта поменялся, объект перемещается в дисковое пространство нового пользователя.

Поддерживаются следующие права доступа к объекту:

- Чтение (3);
- Чтение + Комментирование (4);
- Чтение + запись (5);
- Чтение + запись + удаление своих объектов в общей папке (6);
- Чтение + запись + удаление своих объектов в общей папке + предоставление совместного доступа (7);

– Чтение + запись + предоставление совместного доступа + удаление любых объектов (8);

– Полный доступ (только для владельца) (9).

Изделие поддерживает разделение полномочий администраторов и лиц, обеспечивающих функционирование информационной системы на уровне web-инттерфейса.

Всем, кроме ”Пользователей” доступно средство администрирования.

”Пользователям” доступны остальные приложения. В рамках приложений не существует ролевого разграничения доступа, все пользователи имеют одинаковые права.

Заблокированный пользователь имеет статус enabled: false, временно заблокированный - disabled: true. Статусы пользователей ("User enabled" и "User temporary locked" соответственно) отображаются в интерфейсе Keycloak на странице с пользователем.

Требования: УПД.2, УПД.2-У1, УПД.2-У3, УПД.2-У4, УПД.4, УПД4-У1

3.3 Ограничение попыток входа

Если внешний пользователь превысил количество неудачных попыток аутентификации, то он будет временно заблокирован.

Требования: УПД.6

3.3.1 Автоматическое блокирование

Автоматическое блокирование учетной записи пользователя при превышении пользователем ограничения количества неуспешных попыток доступа выполняется посредством функционала Keycloak и устанавливает статус disabled: true для соответствующего пользователя.

3.3.2 Разблокирование администратором

Параметр Permanent Lockout позволяет реализовать возможность разблокирования пользователя только администратором через соответствующий интерфейс, так как статус пользователя меняется на enabled: false. Если параметр выключен, то обеспечиваются временная блокировка с автоматической разблокировкой.

Параметр Permanent Lockout задается в интерфейсе Keycloak во вкладке Realm Settings/Security Defenses/Brute Force Detection.

Требования: УПД.6-У1

3.3.3 Автоматическое разблокирование

Пользователь будет разблокирован по истечении времени, которое определено как произведение значений параметров Wait Increment * Max Login Failures, но не превышающее параметр Max Wait.

3.4 Блокирование сеанса

Блокирование сеанса по запросу для внешнего пользователя осуществляется с помощью web-интерфейса. Пользователь в меню своего профиля может выполнить ”Выход”, в результате чего сессия будет завершена. Для возобновления сеанса потребуется пройти повторно процедуру идентификации и аутентификации.

При выполнении блокировки с возможностью разблокирования только администратором или иным лицом, имеющим соответствующие полномочия (роль) пользователя все сессии последнего завершаются. В результате пользователь не сможет авторизоваться. Если блокировка произошла после того, как пользователь был авторизован, то любое его действие приведёт к принудительной деавторизации.

Требования: УПД.10, УПД.10-У2

3.5 Действия до идентификации и аутентификации

Внешним пользователям разрешается изменение языка интерфейса, написание письма в службу поддержки до процедуры идентификации и аутентификации.

Требования: УПД.11

3.6 Совместимость со средствами управления доступом

Не рекомендуется отключать SELinux. SELinux должен быть установлен в enforcing в файле /etc/selinux/config.

Наличие включенного SELinux позволяет ограничить потенциально опасные процессы.

4 РЕГИСТРАЦИЯ СОБЫТИЙ

4.1 События, подлежащие регистрации

Журнал аудита расположен в /var/log/syslog.

Регистрируются следующие события безопасности (в скобках указан запрос):

– Создание учетной записи (POST /adminapi/tenants/default/users);

– Вход в систему с учетной записью администратора (POST /adminapi/auth);

– Вход в систему с учетной записью пользователя (POST /pgsapi/?cmd=auth);

– Выход из учетной записи (POST /pgsapi/?cmd=logout);

– Изменение учетных данных (PUT /adminapi/tenants/default/users/{user_id});

– Сброс пароля учетной записи (PUT /adminapi/tenants/default/users/{user_id, также в теле запроса указано "reset_password=******");

– Блокирование учетной записи (DELETE /adminapi/tenants/default/users/{user_id});

– Разблокирование учетной записи (PUT /adminapi/tenants/default/users/{user_id}), также в теле запроса указано "enabled");

– Удаление учетной записи (DELETE /adminapi/tenants/default/users/{user_id}), также в теле запроса указано "full_delete");

– Создание группы (POST /adminapi/tenants/default/groups);

– Удаление группы (DELETE /adminapi/tenants/default/groups/{group_id});

– Добавление учетной записи в группу (POST /adminapi/tenants/default/groups/{group_id});

– Удаление учетной записи из группы (DELETE /adminapi/tenants/default /groups/{group_id}/users/{user_id});

– Операции с объектами доступа (POST /pgsapi/?cmd={put_file, share_object, unshare_object, remove_file}).

Требования: РСБ.1, РСБ.1-У1, РСБ.1-У2, РСБ.1-У3, РСБ.1-У4б

4.2 Состав и содержание событий

Для каждого события выполняется регистрация набора переменных web-сервера nginx:

– имя приложения, источника события (name);

– имя хоста, источника события (host);

– дата и время события в соответствии с rfc3339 (timestamp);

– адрес клиента (remote_addr);

– результат события (status);

– длина запроса, включая строку запроса, заголовок и тело запроса (request_length);

– число байт, переданных клиенту (bytes_sent);

– коэффициент сжатия (gzip_ratio);

– время выполнения запроса в миллисекундах (request_time);

– URL источника запроса (http_referer);

– тип и версия браузера и операционной системы клиента (http_user_agent);

– первоначальная строка запроса - позволяет идентифицировать тип события (request);

– параметры запроса с фильтром чувствительной информации - позволяет идентифицировать субъект доступа (request_body).

Требования: РСБ.2, РСБ.2-У1а

4.3 Сбор, запись и хранение событий

Выбор событий безопасности, подлежащих регистрации осуществляется фильтрацией событий в системном журнале.

Управление генерацией событий выполняется в конфигурации web-сервера:

`/etc/co/nginx/conf/nginx.conf`

Время хранения информации устанавливается в соответствии с системными настройками журналирования ОС - logrotate, rsyslog и journald.

Требования: РСБ.3, РСБ.3-У1

5 КОНТРОЛЬ ЗАЩИЩЕННОСТИ

Для запуска программного средства анализа защищенности следует на сервере с ролью (PGS/CO/LOGOS) выполнить действия от имени административного пользователя ОС:

– Открыть консоль

– Перейти в каталог /opt/co

– Выполнить скрипт:

```
./co_starter.sh status {PGS/CO/MSG}
```

, где переменная MSG соответствует серверу с ролью LOGOS.

Результатом работы станет отчет со статусами функционирования сервисов соответствующих роли сервера (PGS/CO/LOGOS).

Рекомендуется внимательно изучить каждую запись отчёта и, в случае наличия отрицательного статуса выполнить действия, направленные на исправление.

Пример результата проверки анализа защищенности сервисов сервера роли PGS:

Пример результата проверки анализа защищенности сервисов сервера роли CO:

Пример результата проверки анализа защищенности сервисов сервера роли MSG:

Изделие позволяет выполнять контроль работоспособности, параметров настройки и правильности функционирования программного обеспечения в процессе эксплуатации

Руководство администратора по безопасности предоставляется в структурированной форме и позволяет применять автоматизированные средства, обеспечивающие контроль правил генерации и смены паролей пользователей, учетных записей пользователей, правил разграничения доступом и полномочий пользователей.

В качестве дополнительных мер по анализу защищенности рекомендуется выполнить проверку операционной системы на предмет наличия известных уязвимостей.

В случаях, где использование сторонних сканеров с открытым исходным кодом не противоречит политике безопасности компании, данное руководство предлагает применять Vulners (лицензия MIT) и Lynis (GNU GPL)

Требования: АН3.3, АН3.3-У1, АН3.5, АН3.5-У1

6 ОБЕСПЕЧЕНИЕ ДОСТУПНОСТИ

6.1 Создание резервных копий баз данных и важных переменных

Резервное копирование выполняется скриптом /opt/myOffice/scripts/backup.sh.. Данный скрипт вызывается на сервере роли PGS. В процессе выполнения скрипта будет будет предложено ввести пароли от машин, если публичные ключи ssh еще не скопированы.

Скрипт принимает следующие параметры:

- --backup-mode - указывает тип резервного копирования, db - резервное копирование баз данных, var - резервное копирование важных переменных, all - резервное копирование баз данных и важных переменных;
- --backup-user - имя пользователя машины для хранения резервных копий;
- --backup-machine - ip адрес машины для хранения резервных копий;
- --backup-path - путь до папки, где будут храниться резервные копии;
- --user-msg - имя пользователя на сервере роли LOGOS;
- --user-co - имя пользователя на сервере роли CO.

Данный скрипт делает резервные копии важных переменных и баз данных.

На сервере роли PGS следующие базы данных: keycloak - БД keycloak. На сервере роли LOGOS БД с именем logos_demo. В результате выполнения резервного копирования баз данных получим 2 файла соответственно: keycloak.sql, logos_demo.sql.

При выполнении резервного копирования важных переменных создаются копии следующих файлов:

1. Сервер роли PGS:
   - /etc/sysconfig/myoffice
   - /etc/sysconfig/myoffice-auth.props
   - /etc/sysconfig/myoffice-tenant-params
2. Сервер роли CO:
   - /etc/sysconfig/myoffice
3. Сервер роли Logos:
   - /etc/sysconfig/myoffice

В результате выполнения резервного копирования важных переменных в результирующей папке получим следующие файлы:

- myoffice_pgs (копия /etc/sysconfig/myoffice сервера роли PGS)
- myoffice_co (копия /etc/sysconfig/myoffice сервера роли CO)
- myoffice_msg (копия /etc/sysconfig/myoffice сервера роли Logos)
- myoffice-auth.props

- /etc/sysconfig/myoffice-tenant-params

Последовательность действий для выполнения резервного копирования:

1. Запустить ssh сервис на сервере роли CO командой:

```
sudo systemctl start ssh
```

2. Запустить ssh сервис на сервере роли Logos командой:

```
sudo systemctl start ssh
```

3. Запустить ssh сервис на машине для резервного копирования командой:

```
sudo systemctl start ssh
```

4. Сделать скрипт /opt/myOffice/scripts/backup.sh исполняемой командой:

```
sudo chmod oug+x backup.sh
```

5. Выполнить запустить скрипт backup.sh:

```
sudo ./backup.sh --backup-mode={db/var/all} --backup-user={user of backup_machine} --backup-machine={backup machine address} --backup-path={path to backup at backup machine} --user-msg={msg machine user} --user-co={co machine user}
```

В результате резервного копирования должны получиться следующие файлы и папки:

- keycloak.sql (резервная копия базы данных keycloak для роли PGS)
- logos_demo.sql (резервная копия базы данных logos_demo для роли LOGOS)
- myoffice-auth.props (резервная копия файла с важными переменными для всех ролей)
- myoffice_co (резервная копия файла с важными переменными для роли CO)
- myoffice_pgs (резервная копия файла с важными переменными для роли PGS)
- myoffice_msg (резервная копия файла с важными переменными для роли LOGOS)
- myoffice-tenant-params (резервная копия файла с важными переменными для всех ролей)
- папка dump/ (резервная копия базы данных arangodb для роли PGS)

Текст скрипта backup.sh:

```
#!/bin/sh
	[ -f ./utils.sh ] && source ./utils.sh || exit 1
	source /etc/sysconfig/myoffice-tenant-params
	source /etc/sysconfig/myoffice

	snct_msg 0 "Starting backup"

	usage() {
	echo "Usage:"
```

```
        echo $0 --backup-mode={db/var/all} --backup-user={user of backup_machine} --backup-machine={backup
machine address} --backup-path={path to backup at backup machine} --user-msg={msg machine user} --user-co={co
machine user}
        exit 1
        }

        param_array=("--backup-mode|BACKUP_MODE"
                  "--backup-user|BACKUP_USER"
                 "--backup-machine|BACKUP_MACHINE"
                 "--backup-path|BACKUP_PATH"
                 "--user-msg|MSG_USER"
                 "--user-co|CO_USER")
        snct_un_read_params "$*" "$param_array" || usage

        if [[ ! -f ~/.ssh/id_rsa ]] && [[ ! -f ~/.ssh/id_rsa.pub ]]
        then
             snct_msg 0 "Keys id_rsa and id_rsa.pub not found in ~/.ssh"
             snct_msg 0 "Generating ssh keys"
             ssh-keygen -f ~/.ssh/id_rsa -N ""
             snct_msg 0 "Keys successfully created"
        else
            snct_msg 0 "Keys exists"
        fi

        snct_exec "ssh-copy-id $BACKUP_USER@$BACKUP_MACHINE -i ~/.ssh/id_rsa.pub -o
\"StrictHostKeyChecking=no\""; snct_msg $? "Copy ssh key to backup machine"
        snct_exec "ssh-copy-id $MSG_USER@$MSG_IP -i ~/.ssh/id_rsa.pub -o \"StrictHostKeyChecking=no\"";
snct_msg $? "Copy ssh key to MSG machine"
        snct_exec "ssh-copy-id $CO_USER@$CO_IP -i ~/.ssh/id_rsa.pub -o #!/bin/sh
        [ -f ./utils.sh ] && source ./utils.sh || exit 1
        source /etc/sysconfig/myoffice-tenant-params
        source /etc/sysconfig/myoffice

        snct_msg 0 "Starting backup"

        usage() {
        echo "Usage:"
        echo $0 --backup-mode={db/var/all} --backup-user={user of backup_machine} --backup-machine={backup
machine address} --backup-path={path to backup at backup machine} --user-msg={msg machine user} --user-co={co
machine user}
        exit 1
        }

        param_array=("--backup-mode|BACKUP_MODE"
                  "--backup-user|BACKUP_USER"
                 "--backup-machine|BACKUP_MACHINE"
                 "--backup-path|BACKUP_PATH"
                 "--user-msg|MSG_USER"
                 "--user-co|CO_USER")
        snct_un_read_params "$*" "$param_array" || usage

        if [[ ! -f ~/.ssh/id_rsa ]] && [[ ! -f ~/.ssh/id_rsa.pub ]]
        then
             snct_msg 0 "Keys id_rsa and id_rsa.pub not found in ~/.ssh"
             snct_msg 0 "Generating ssh keys"
             ssh-keygen -f ~/.ssh/id_rsa -N ""
             snct_msg 0 "Keys successfully created"
        else
            snct_msg 0 "Keys exists"
        fi
```

```
        snct_exec "ssh-copy-id $BACKUP_USER@$BACKUP_MACHINE -i ~/.ssh/id_rsa.pub -o \"StrictHostKeyChecking=no\""; snct_msg $? "Copy ssh key to backup machine"
        snct_exec "ssh-copy-id $MSG_USER@$MSG_IP -i ~/.ssh/id_rsa.pub -o \"StrictHostKeyChecking=no\""; snct_msg $? "Copy ssh key to MSG machine"
        snct_exec "ssh-copy-id $CO_USER@$CO_IP -i ~/.ssh/id_rsa.pub -o \"StrictHostKeyChecking=no\""; snct_msg $? "Copy ssh keys to CO machine"

        backup_databases () {
            sudo -u keycloak pg_dump "host=127.0.0.1 port=5432 dbname=keycloak" | ssh $BACKUP_USER@$BACKUP_MACHINE "cat > $BACKUP_PATH/keycloak.sql"
            snct_msg $? "Backup keycloak database"
            mkdir tmp
            arangodump --server.password $ARANGO_ROOT_PASSWORD --all-databases --output-directory tmp/dump
            chmod oug+x tmp/dump/ -R
            scp -r tmp/dump $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH
            rm -rf tmp/
            snct_msg $? "Backup arangodb"
           ssh $MSG_USER@$MSG_IP "sudo -u msg pg_dump --host=$MSG_IP --port=5432 --dbname=logos_demo" | ssh $BACKUP_USER@$BACKUP_MACHINE "cat > $BACKUP_PATH/logos_demo.sql"
            snct_msg $? "Backup logos_demo database"
        }

        backup_vars () {
            snct_exec "scp /etc/sysconfig/myoffice $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice_pgs"; snct_msg $? "Backup /etc/sysconfig/myoffice PGS"
            snct_exec "scp /etc/sysconfig/myoffice-auth.props $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH"; snct_msg $? "Backup /etc/sysconfig/myoffice-auth.props"
            snct_exec "scp /etc/sysconfig/myoffice-tenant-params $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH"; snct_msg $? "Backup /etc/sysconfig/myoffice-tenant-params"
            mkdir tmp
            snct_exec "scp $MSG_USER@$MSG_IP:/etc/sysconfig/myoffice tmp/myoffice_msg"
            snct_exec "scp tmp/myoffice_msg $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH"; snct_msg $? "Backup /etc/sysconfig/myoffice MSG"
            snct_exec "scp $CO_USER@$CO_IP:/etc/sysconfig/myoffice tmp/myoffice_co"
            snct_exec "scp tmp/myoffice_co $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH"; snct_msg $? "Backup /etc/sysconfig/myoffice CO"
            rm -rf tmp
        }

        case "$BACKUP_MODE" in
            "db" ) backup_databases;;
            "var" ) backup_vars;;
            "all" )
            backup_databases
            backup_vars
            ;;
        esac\"StrictHostKeyChecking=no\""; snct_msg $? "Copy ssh keys to CO machine"

        backup_databases () {
            sudo -u keycloak pg_dump "host=127.0.0.1 port=5432 dbname=keycloak" | ssh $BACKUP_USER@$BACKUP_MACHINE "cat > $BACKUP_PATH/keycloak.sql"
            snct_msg $? "Backup keycloak database"
            mkdir tmp
            arangodump --server.password $ARANGO_ROOT_PASSWORD --all-databases --output-directory tmp/dump
            chmod oug+x tmp/dump/ -R
            scp -r tmp/dump $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH
```

```
            rm -rf tmp/
            snct_msg $? "Backup arangodb"
           ssh $MSG_USER@$MSG_IP "sudo -u msg pg_dump --host=$MSG_IP --port=5432 --dbname=logos_demo"
| ssh $BACKUP_USER@$BACKUP_MACHINE "cat > $BACKUP_PATH/logos_demo.sql"
            snct_msg $? "Backup logos_demo database"
        }

        backup_vars () {
            snct_exec "scp /etc/sysconfig/myoffice
$BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice_pgs"; snct_msg $? "Backup
/etc/sysconfig/myoffice PGS"
            snct_exec "scp /etc/sysconfig/myoffice-auth.props
$BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH"; snct_msg $? "Backup /etc/sysconfig/myoffice-
auth.props"
            snct_exec "scp /etc/sysconfig/myoffice-tenant-params
$BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH"; snct_msg $? "Backup /etc/sysconfig/myoffice-
tenant-params"
            mkdir tmp
            snct_exec "scp $MSG_USER@$MSG_IP:/etc/sysconfig/myoffice tmp/myoffice_msg"
            snct_exec "scp tmp/myoffice_msg $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH";
snct_msg $? "Backup /etc/sysconfig/myoffice MSG"
            snct_exec "scp $CO_USER@$CO_IP:/etc/sysconfig/myoffice tmp/myoffice_co"
            snct_exec "scp tmp/myoffice_co $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH";
snct_msg $? "Backup /etc/sysconfig/myoffice CO"
            rm -rf tmp
        }

        case "$BACKUP_MODE" in
            "db" ) backup_databases;;
            "var" ) backup_vars;;
            "all" )
            backup_databases
            backup_vars
            ;;
Esac
```

6.2 Восстановление из резервной копии

Восстановление из резервных копий выполняется скриптом /opt/myOffice/scripts/backup_recovery.sh. Данный скрипт вызывается на сервере роли PGS. В процессе выполнения скрипта будет будет предложено ввести пароли от машин, если публичные ключи ssh еще не скопированы.

Скрипт принимает следующие параметры:

- --recovery-mode - указывает тип восстановления, db - баз данных, var - важных переменных, all - баз данных и важных переменных;
- --recovery-machine - указывает машину, на которой необходимо выполнить восстановление (PGS/CO/MSG);
- --backup-machine - ip адрес машины для хранения резервных копий;
- --backup-path - путь до папки, где будут храниться резервные копии;
- --user-msg - имя пользователя на сервере роли LOGOS;

- --user-co - имя пользователя на сервере роли CO.

Последовательность действий для выполнения восстановления:

1. Запустить ssh сервис на сервере роли CO командой:

```
sudo systemctl start ssh
```

2. Запустить ssh сервис на сервере роли Logos командой:

```
sudo systemctl start ssh
```

3. Запустить ssh сервис на машине для резервного копирования командой:

```
sudo systemctl start ssh
```

4. Сделать скрипт /opt/myOffice/scripts/backup_recovery.sh исполняемой командой:

```
sudo chmod oug+x backup_recovery.sh
```

5. Выполнить запустить скрипт backup_recovery.sh:

```
sudo ./backup_recovery.sh --recovery-mode={db/var/all} --recovery-machine={PGS/CO/MSG} --backup-user={user of backup_machine} --backup-machine={backup machine address} --backup-path={path to backup at backup machine} --user-msg={msg machine user} --user-co={co machine user}
```

Для полного восстановления необходимо выполнить скрипт для серверов каждой роли (PGS, CO, Logos).

Текст скрипта backup_recovery.sh:

```
#!/bin/sh
[ -f ./utils.sh ] && source ./utils.sh || exit 1

snct_msg 0 "Starting backup recovery"

usage() {
echo "Usage:"
echo $0 --recovery-mode={db/var/all} --recovery-machine={PGS/CO/MSG} --backup-user={user of backup_machine} --backup-machine={backup machine address} --backup-path={path to backup at backup machine} --user-msg={msg machine user} --user-co={co machine user}
exit 1
}

param_array=("--recovery-mode|RECOVERY_MODE"
             "--recovery-machine|RECOVERY_MACHINE"
             "--backup-user|BACKUP_USER"
            "--backup-machine|BACKUP_MACHINE"
            "--backup-path|BACKUP_PATH"
            "--user-msg|MSG_USER"
            "--user-co|CO_USER")
snct_un_read_params "$*" "$param_array" || usage

if [[ -f /etc/sysconfig/myoffice-tenant-params ]]
then
    source /etc/sysconfig/myoffice-tenant-params
else
   scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice-tenant-params /etc/sysconfig/myoffice-tenant-params
    source /etc/sysconfig/myoffice-tenant-params
```

```
fi

if [[ -f /etc/sysconfig/myoffice ]]
then
   source /etc/sysconfig/myoffice
else
   scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice /etc/sysconfig/myoffice
   source /etc/sysconfig/myoffice
fi

if [[ ! -f ~/.ssh/id_rsa ]] && [[ ! -f ~/.ssh/id_rsa.pub ]]
then
    snct_msg 0 "Keys id_rsa and id_rsa.pub not found in ~/.ssh"
    snct_msg 0 "Generating ssh keys"
    ssh-keygen -f ~/.ssh/id_rsa -N ""
    snct_msg 0 "Keys successfully created"
else
    snct_msg 0 "Keys exists"
fi

snct_exec "ssh-copy-id $BACKUP_USER@$BACKUP_MACHINE -i ~/.ssh/id_rsa.pub -o \"StrictHostKeyChecking=no\""; snct_msg $? "Copy ssh key to backup machine"
snct_exec "ssh-copy-id $MSG_USER@$MSG_IP -i ~/.ssh/id_rsa.pub -o \"StrictHostKeyChecking=no\""; snct_msg $? "Copy ssh key to MSG machine"
snct_exec "ssh-copy-id $CO_USER@$CO_IP -i ~/.ssh/id_rsa.pub -o \"StrictHostKeyChecking=no\""; snct_msg $? "Copy ssh key to CO machine"

recovery_pgs () {
    if [[ $1 = "all" ]] || [[ $1 = "db" ]]
    then
            systemctl stop keycloak
            systemctl stop nct-logos-sync
            su - postgres -c 'psql -lqt | cut -d \| -f 1 | grep -qw keycloak'
            if [[ $? = 0 ]]
            then
                    snct_exec "su - postgres -c 'dropdb keycloak' "; snct_msg $? "Drop db keycloak"
            fi
            su - postgres -c 'psql -tAc "SELECT * FROM pg_roles" | cut -d \| -f 1 | grep keycloak'
            if [[ $? != 0 ]]
            then
                    su - postgres -c "createuser -s keycloak"
                    snct_msg $? "Create db user keycloak"
            fi
            snct_exec "su - postgres -c 'createdb keycloak -O keycloak' "; snct_msg $? "Create db keycloak"

            mkdir tmp
            snct_exec "scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/keycloak.sql tmp/"; snct_msg $? "Copy keycloak dump file"

            snct_exec "su - postgres -c 'psql keycloak' < tmp/keycloak.sql"; snct_msg $? "Recovery keycloak"
            systemctl start keycloak
            snct_exec "echo 'db._dropDatabase(\"pgs\");' | OPENSSL_CONF=/etc/ssl/arangodb3/openssl.cnf arangosh --server.password $ARANGO_ROOT_PASSWORD"; snct_msg $? "Drop pgs db"
            snct_exec "echo 'db._createDatabase(\"pgs\");' | OPENSSL_CONF=/etc/ssl/arangodb3/openssl.cnf arangosh --server.password $ARANGO_ROOT_PASSWORD"; snct_msg $? "Create pgs db"
            snct_exec "scp -r $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/dump/ tmp/"; snct_msg $? "Copy arangodb dump files"
```

```
                snct_exec "arangorestore --server.password $ARANGO_ROOT_PASSWORD --all-databases --input-directory ./tmp/dump/"; snct_msg $? "Recovery pgs db"
                rm -rf tmp
                systemctl restart nct-aristoteles
                systemctl start nct-euclid
                sleep 10
                systemctl start nct-logos-sync
        fi
        if [[ $1 = "all" ]] || [[ $1 = "var" ]]
        then
                snct_exec "scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice_pgs /etc/sysconfig/myoffice"; snct_msg $? "Recovery /etc/sysconfig/myoffice"
                snct_exec "scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice-auth.props /etc/sysconfig/myoffice-auth.props"; snct_msg $? "Recovery /etc/sysconfig/myoffice-auth.props"
                snct_exec "scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice-tenant-params /etc/sysconfig/myoffice-tenant-params"; snct_msg $? "Recovery /etc/sysconfig/myoffice-tenant-params"
        fi
}

recovery_msg () {
    if [[ $1 = "all" ]] || [[ $1 = "db" ]]
    then
                ssh $MSG_USER@$MSG_IP "sudo /opt/co/co_starter.sh stop MSG"
                snct_msg $? "Stop MSG"
                systemctl stop nct-logos-sync
                ssh $MSG_USER@$MSG_IP "sudo su - postgres -c 'psql -lqt | cut -d \| -f 1 | grep -qw logos_demo'"
                if [[ $? = 0 ]]
                then
                        snct_exec "ssh $MSG_USER@$MSG_IP \"sudo su - postgres -c 'dropdb logos_demo'\""; snct_msg $? "Drop logos_demo db"
                fi
                ssh $MSG_USER@$MSG_IP "sudo su - postgres -c 'psql -tAc \"SELECT * FROM pg_roles\" | cut -d \| -f 1 | grep msg'"
                if [[ $? != 0 ]]
                then
                        ssh $MSG_USER@$MSG_IP "sudo su - postgres -c \"createuser -s msg\""
                        snct_msg $? "Create db user msg"
                fi
                snct_exec "ssh $MSG_USER@$MSG_IP \"sudo su - postgres -c 'createdb logos_demo -O msg'\""; snct_msg $? "Create db logos_demo"
                ssh $MSG_USER@$MSG_IP "mkdir ~/tmp"
                mkdir tmp
                snct_exec "scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/logos_demo.sql tmp/"; snct_msg $? "Copy logos_demo dump file"
                snct_exec "scp tmp/logos_demo.sql $MSG_USER@$MSG_IP:~/tmp"
                snct_exec "ssh $MSG_USER@$MSG_IP \"sudo su - postgres -c 'psql logos_demo' < ~/tmp/logos_demo.sql\""; snct_msg $? "Recovery logos_demo"
                ssh $MSG_USER@$MSG_IP "rm -rf ~/tmp"
                rm -rf tmp
                sleep 10
                ssh $MSG_USER@$MSG_IP "sudo /opt/co/co_starter.sh start MSG"
                snct_msg $? "Start MSG"
                systemctl start nct-logos-sync
    fi
    if [[ $1 = "all" ]] || [[ $1 = "var" ]]
    then
                mkdir tmp
                snct_exec "scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice_msg tmp"
```

```
            snct_exec "scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice-auth.props tmp"
            snct_exec "scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice-tenant-params tmp"
            snct_exec "ssh $MSG_USER@$MSG_IP \"mkdir ~/tmp/\""
                    snct_exec "scp tmp/myoffice_msg $MSG_USER@$MSG_IP:~/tmp/"
                    snct_exec "scp tmp/myoffice-auth.props $MSG_USER@$MSG_IP:~/tmp/"
                    snct_exec "scp tmp/myoffice-tenant-params $MSG_USER@$MSG_IP:~/tmp/"
                    snct_exec "ssh $MSG_USER@$MSG_IP \"sudo cp ~/tmp/myoffice_msg /etc/sysconfig/myoffice ; sudo cp ~/tmp/myoffice-tenant-params /etc/sysconfig/ ; sudo cp ~/tmp/myoffice-auth.props /etc/sysconfig/ ; rm -rf ~/tmp\""
                    snct_msg 0 "Copy vars"
                    rm -rf tmp
        fi
    }

    recovery_co () {
        mkdir tmp
        snct_exec "scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice_co tmp"
       snct_exec "scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice-auth.props tmp"
       snct_exec "scp $BACKUP_USER@$BACKUP_MACHINE:$BACKUP_PATH/myoffice-tenant-params tmp"
       snct_exec "ssh $CO_USER@$CO_IP \"mkdir ~/tmp/\""
        snct_exec "scp tmp/myoffice_co $CO_USER@$CO_IP:~/tmp/"
        snct_exec "scp tmp/myoffice-auth.props $CO_USER@$CO_IP:~/tmp/"
        snct_exec "scp tmp/myoffice-tenant-params $CO_USER@$CO_IP:~/tmp/"
        snct_exec "ssh $CO_USER@$CO_IP \"sudo cp ~/tmp/myoffice_co /etc/sysconfig/myoffice ; sudo cp ~/tmp/myoffice-tenant-params /etc/sysconfig/ ; sudo cp ~/tmp/myoffice-auth.props /etc/sysconfig/ ; rm -rf ~/tmp\""

       snct_msg 0 "Copy vars"
       rm -rf tmp
    }

    case "$RECOVERY_MACHINE" in
        "PGS" ) recovery_pgs $RECOVERY_MODE;;
        "MSG" ) recovery_msg $RECOVERY_MODE;;
        "CO" ) recovery_co;;
    esac
```

Требования: ОДТ.4, ОДТ.4 -У1, ОДТ.4-У3

6.3 Создание резервных копий на случай полного отказа

На случай полного отказа одного или нескольких серверов установки необходимо скопировать на сервер для резервного копирования папку /opt/myOffice/install_bundle на сервере роли PGS.

6.4 Восстановление системы в случае полного отказа

В случае полного отказа сервера роли PGS необходимо установить роль PGS на сервер в соответствии с п. 2.3.1 руководства администратора. В процессе установки на этапе 6 необходимо скопировать папку install_bundle с сервера для резервного копирования в папку /opt/myOffice вместо генерации нового бандла. Далее выполнять установку согласно

инструкции. После выполнения установки выполнить восстановление баз данных и важных переменных в соответствии с подразделом 6.2 настоящего руководства.

В случае полного отказа сервера роли LOGOS необходимо установить роль LOGOS на сервер в соответствии с п. 2.3.3 руководства администратора. В процессе установки на этапе 6 необходимо скопировать папку install_bundle с сервера для резервного копирования в папку /opt/myOffice. Далее выполнять установку согласно инструкции. После выполнения установки выполнить восстановление баз данных и важных переменных в соответствии с подразделом 6.2 настоящего руководства.

В случае полного отказа сервера роли CO необходимо установить роль CO на сервер в соответствии с п. 2.3.2 руководства администратора. В процессе установки на этапе 6 необходимо скопировать папку install_bundle с сервера для резервного копирования в папку /opt/myOffice. Далее выполнять установку согласно инструкции. После выполнения установки выполнить восстановление важных переменных в соответствии с подразделом 6.2 настоящего руководства.

7 ТЕХНИЧЕСКАЯ ПОДДЕРЖКА

Контактная информация службы технической поддержки ООО «Новые облачные технологии» в случае возникновения вопросов, неописанных в данном руководстве:

Адрес электронной почты: support@service.myoffice.ru

Телефон: 8-800-222-1-888.