Yandex Cloud

Стандарт по защите облачной инфраструктуры Yandex Cloud

Введение

Этот документ содержит рекомендации по техническим мерам защиты и помогает выбрать меры обеспечения информационной безопасности (ИБ) при развёртывании информационных систем на облачной платформе Yandex Cloud.

Рекомендации и меры обеспечения безопасности в стандарте сопровождаются ссылками на **инструкции и решения по настройке** безопасных конфигураций ресурсов с помощью штатных средств защиты информации и дополнительных средств защиты, доступных пользователям Yandex Cloud.

Также стандарт описывает способы и средства проверки выполнения рекомендаций, в том числе:

- с помощью интерфейса консоли управления;
- с помощью интерфейса командной строки Yandex Cloud CLI;
- вручную.

Область применения

Рекомендации предназначены для архитекторов, технических специалистов и специалистов по ИБ, которые используют при создании защищённых облачных систем и разработке политик безопасности для работы на облачной платформе следующие сервисы:

- Identity and Access Management (IAM)
- Application Load Balancer
- Audit Trails

- Certificate Manager
- Cloud DNS
- Cloud Logging
- Compute Cloud
- Key Management Service
- Yandex Lockbox
- Managed Service for ClickHouse
- Yandex Managed Service for GitLab
- Managed Service for MongoDB
- Managed Service for MySQL
- Managed Service for PostgreSQL
- Managed Service for Redis
- Network Load Balancer
- Object Storage
- Organization
- Resource Manager
- Virtual Private Cloud
- Managed Service for YDB

Стандарт можно рассматривать как основу для разработки рекомендаций, специфичных для организации. Не все меры обеспечения ИБ и рекомендации из настоящего документа могут быть применимы. Кроме того, могут потребоваться дополнительные меры и рекомендации, не включённые в настоящий стандарт.

Структура стандарта

Стандарт описывает рекомендации для следующих задач обеспечения безопасности:

- Аутентификация и управление доступом.
- Сетевая безопасность.

Требования и подготовка

Для проверок убедитесь, что

- установлен и настроен YC CLI по инструкции;
- вы вошли в консоль управления;
- установлена утилита jq.

Вы можете автоматизировать аудит выполнения всех рекомендаций с помощью доступных решений наших партнёров:
Neocat — продукт для управления безопасностью в облаке от компании Неофлекс. Устанавливается изолированно в периметре облака пользователя без необходимости выдачи административных прав.
Cloud Advisor — безагентская платформа выявляет и приоритезирует риски безопасности, сокращает расходы, обеспечивает соответствие требованиям и упрощает управление вашей облачной инфраструктурой.

Ограничение ответственности

В Yandex Cloud применяется концепция разделения ответственности. Граница разделения ответственности за обеспечение безопасности зависит от сервисов, которые используются системой в облаке, от модели использования этих сервисов (IaaS — инфраструктура как услуга, PaaS — платформа как услуга, SaaS — программное обеспечение как услуга) и имеющихся у облачного провайдера защитных механизмов и политик.

Термины и сокращения

В настоящем документе используются термины и определения, введенные стандартом ISO/IEC 27000:2018 и ISO/IEC 29100:2011, а также термины, используемые в глоссарии Yandex Cloud.

Оглавление

1. Аутентификация и управление доступом 7
1.1 Настроена федерация удостоверений (Single Sign-On, SSO) 7
1.2 Учётные записи Яндекс ID используются только в исключительных случаях 9
1.3 Только необходимые администраторы управляют членством в IAM-группах 10
1.4 Используются сервисные роли вместо примитивных: admin, editor, viewer 11
1.5 Облачные сущности с сервисными аккаунтами учтены и ограничены 13
1.6 В сервисе метаданных ВМ отсутствуют облачные ключи в открытом виде 14
1.7 На ВМ отключено получение токена через AWS IMDSv1 16
1.8 Сервисным аккаунтам назначены минимальные привилегии 17
1.9 Только доверенные администраторы имеют доступ к сервисным аккаунтам 19
1.10 Выполняется периодическая ротация ключей сервисных аккаунтов 21
1.11 Настроена двухфакторная аутентификация для привилегированных аккаунтов 23
1.12 Привилегированные роли назначены только доверенным администраторам 24
1.13 Для локальных пользователей управляемых БД задан стойкий пароль 28
1.14 Доступ для подрядных организаций и третьих лиц контролируется 28
1.15 Используется корректная ресурсная модель 29
1.16 На ресурсах в организации отсутствует «публичный доступ» 29
1.17 Контактные данные ответственного за организацию актуальны 33
1.18 Таймаут жизни cookie в федерации меньше 6 часов 33
1.19 Токен для облачных функций и ВМ выдаётся через сервисный аккаунт 35
2. Сетевая безопасность 36
Введение 36
2.1 Для объектов облака используется межсетевой экран или группы безопасности 36
2.2 Как минимум одна Группа безопасности существует в VPC 39
2.3 В Группах безопасности отсутствует слишком широкое правило доступа 40
2.4 Доступ по управляющим портам открыт только для доверенных IP-адресов 41
2.5 Включена защита от DDoS атак 42
2.6 Используется защищённый удалённый доступ 43
2.7 Исходящий доступ в интернет контролируется 45
2.8 Запросы DNS не передаются в сторонние рекурсивные резолверы 47
3. Безопасная конфигурация виртуальной среды 48
Введение 48
3.1 Использование серийной консоли контролируется либо отсутствует 48
3.2 Используется эталонный образ для развёртывания ВМ 49
3.3 Инструмент Terraform используется в соответствии с лучшими практиками ИБ 51

3.4 Выполняется контроль целостности файлов 52
3.5 Учтены принципы защиты от атак по побочным каналам (side-chanel) 52
Yandex Object Storage 53
3.6 Отсутствует публичный доступ к бакету Object Storage 53
3.7 В Object Storage используются политики доступа (Bucket Policy) 54
3.8 В Object Storage включена функция «Блокировка версии объекта» (object lock) 55
3.9 В Object Storage включён механизм логирования действий с бакетом 57
3.10 В Object Storage настроено управление кросс-доменными запросами (CORS) 57
Managed Services for Databases 58
3.11 На управляемых базах данных назначена Группа безопасности 58
3.12 На управляемых базах данных не назначен публичный IP-адрес 59
3.13 Включена настройка защиты от удаления (deletetion protection) 60
3.14 Выключена настройка доступа из DataLens без необходимости 61
3.15 На управляемых БД выключен доступ из консоли управления 63
Cloud Functions и Yandex API Gateway 64
3.16 Публичные облачные функции применяются только в исключительных случаях 64
3.17 Учтены атаки по побочным каналам в Cloud Functions 65
3.18 Учтены особенности синхронизации времени в Cloud Functions 65
3.19 Учтены особенности управления заголовками в Cloud Functions 66
3.20 Serverless Containers/Cloud Functions использует внутреннюю сеть VPC 66
Managed Service for YDB 67
3.21 Учтены рекомендации по работе с конфиденциальными данными в YDB 67
3.22 Учтены рекомендации по защите от sql- инъекций YDB 68
3.23 Публичный доступ отсутствует для YDB 68
3.24 Учтены рекомендации по резервному копированию YDB 69
Yandex Container Registry 69
3.25 Настроен ACL по IP адресам для Yandex Container Registry 69
Yandex Container Solution 70
3.26 Срок действия сертификата Yandex Certificate Manager как минимум 30 дней 71
Yandex Managed Service for GitLab 73
3.27 Рекомендации по настройке безопасности инстанса GitLab выполняются 73
4. Шифрование данных и управление ключами 74
Введение 74
Шифрование в состоянии покоя (at rest) 74
4.1 В Yandex Object Storage включено шифрование данных at rest с ключом KMS 74
Шифрование в состоянии передачи (in transit) 75
4.2 В Yandex Object Storage включено HTTPS для хостинга статического сайта 76

4.3 В Yandex Application Load Balancer используется HTTPS 77
4.4 В Yandex API Gateway используется HTTPS и собственный домен 78
4.5 В Yandex Cloud CDN используется HTTPS и собственный ssl сертификат 79
4.6 Для критичных ВМ настроено шифрование диска с помощью KMS 80
4.7 Для критичных данных используется шифрование с помощью KMS 81
4.8 Используется шифрование данных на уровне приложения 81
Управление ключами 82
4.9 В KMS используется аппаратный модуль безопасности (HSM) 82
4.10 Права управление ключами в KMS выданы контролируемым пользователям 83
4.11 Для KMS ключей включена ротация 85
4.12 Убедитесь, что для KMS ключей включена защита от удаления 87
Управление секретами 88
4.13 В организации используется сервис Secret Management — Yandex Lockbox 88
4.14 Для Serverless Containers и Cloud Functions используются Lockbox секреты 89
4.15 При работе Container Optimized Image используется шифрование секретов 90
5. Сбор, мониторинг и анализ аудитных логов 91
Введение 91
5.1 Включён сервис Audit Trails на уровне организации 91
5.2 События Audit Trails экспортируются в SIEM-системы 92
5.3 Настроено реагирование на события Audit Trails 93
5.4 Выполнен hardering бакета Object Storage, где хранятся аудит логи Audit Trails 94
5.5 Выполняется сбор аудит логов с уровня ОС 94
5.6 Выполняется сбор аудит логов с уровня приложений 95
5.7 Выполняется сбор логов с уровня сети 95
6. Управление уязвимостями 96
Введение 96
6.1 Для образов контейнеров используется сканер уязвимостей 96
6.2 Выполняется сканирование уязвимостей на уровне облачных IP-адресов 96
6.3 Внешние сканирования безопасности выполняются по правилам облака 97
6.4 Выстроен процесс обновлений безопасности 97
6.5 Используется Web Application Firewall 98
7. Резервное копирование 99
7.1 Используется Cloud Backup или механизм snapshot по расписанию 99
8. Физическая безопасность 100

1. Аутентификация и управление доступом

В Yandex Cloud управление идентификацией, аутентификацией и контролем доступа осуществляется сервисами Yandex Identity and Access Management (IAM) и Yandex Cloud Organization.

Платформа работает с тремя категориями пользователей:

- Аккаунты на Яндексе — учётные записи в системе Яндекс ID.
- Федеративные аккаунты — учётные записи в корпоративной SAML-совместимой федерации удостоверений, например Active Directory.
- Сервисные аккаунты — учётные записи, от имени которых программы могут управлять ресурсами.

Аккаунты Yandex ID и федеративные аккаунты аутентифицируются в собственных системах. Yandex Cloud не имеет доступа к паролям этих пользователей и аутентифицирует только сервисные аккаунты с помощью сервиса IAM.

Доступ пользователей к ресурсам облака регулируется с помощью ролей. Сервисы Yandex Cloud могут предлагать разный уровень гранулярности назначения прав: в одних случаях роль можно назначить непосредственно на сам ресурс в сервисе, в других случаях права назначаются только на уровне каталога или облака, в котором размещён ресурс сервиса.

Таким образом, в инфраструктуре Yandex Cloud взаимодействуют различные категории ресурсов, ролей и пользователей. Контроль доступа к ресурсам выполняется сервисом IAM. Сервис IAM контролирует каждый запрос, чтобы все операции над ресурсами выполнялась только пользователями с необходимыми правами.

1.1 Настроена федерация удостоверений (Single Sign-On, SSO)

Yandex Cloud Organization — это единый сервис для управления составом организации, настройки интеграции с каталогом сотрудников и разграничения доступов пользователей к облачным ресурсам организации.

Для централизованного управления учётными данными используйте SAML- совместимые федерации удостоверений. С помощью федераций удостоверений компания может настроить Single Sign-On аутентификацию в Yandex Cloud через свой сервер IdP. При таком подходе сотрудники могут

использовать свои корпоративные аккаунты, на которые распространяются политики безопасности компании, такие как:

- отзыв и блокирование аккаунтов;
- парольные политики;
- ограничение количества неуспешных попыток входа;
- блокирование сеанса доступа после установленного времени бездействия;
- двухфакторная аутентификация.

> Используйте федеративные аккаунты вместо аккаунтов Яндекс ID, где это возможно.
>
> Помните, что для управления федерацией существует отдельная роль organization-manager.federations.admin.

Чтобы все запросы аутентификации от Yandex Cloud содержали цифровую подпись, включите опцию **Подписывать запросы аутентификации**.
Для завершения настройки потребуется скачать и установить сертификат Yandex Cloud. Скачать сертификат можно в поле **Подписывать запросы аутентификации** сразу после сохранения федерации.

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в вашем браузере.
- Перейдите во вкладку **Все сервисы** → **Cloud Organization** → **Федерации**.
- Если в списке есть хотя бы одна настроенная федерация удостоверений, то рекомендация выполнена.
  Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Посмотрите информацию о настроенных федерациях:

```
export ORG_ID=<ID_организации>
yc organization-manager federation saml list
  --organization-id=$ORG_ID
```

- Если в списке есть хотя бы одна настроенная федерация удостоверений, то рекомендация выполнена.
  Если нет, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

- Инструкция по настройке SAML федерации удостоверений.
- Инструкция по настройке SAML федерации с KeyCloak.

1.2 Учётные записи Яндекс ID используются только в исключительных случаях

Наиболее правильный с точки зрения безопасности подход к управлению учётными записями — это использование федерации удостоверений (подробнее в рекомендации № 1.1). В связи с этим необходимо стремиться к тому, чтобы в списке пользователей вашей организации находились только федеративные пользователи (пользователи с атрибутом "FEDERATION ID") и минимум учётных записей с Яндекс ID. Список допустимых исключений:

- Учётная запись с правами billing.accounts.owner (технически на текущий момент данную роль может иметь только учётная запись Яндекс ID).
- Учётная запись с правами organization-manager.organizations.owner и resource-manager.cloud.owner, если вы используете её только для аварийного применения, например, когда сломалась настройка федерации. При необходимости можно удалить привилегированный паспортный аккаунт с ролью organization-manager.organizations.owner из организации.
- Внешние учётные записи, например, контрагентов или подрядчиков, которые по каким-либо причинам вы не можете завести в вашей IdP.

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в браузере.
- Перейдите во вкладку **Все сервисы** → **Cloud Organization** → **Пользователи**.
- Если у всех учётных записей в колонке **Федерация** выставлено значение **federation** (кроме записей, указанных в списке допустимых исключений выше), то рекомендация выполняется. Если нет, перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска нефедеративных учётных записей в вашей организации, за исключением ID учётной записи, которая входит в список валидных исключений:

```
export ORG_ID=<ID_организации>
yc organization-manager user list
--organization-id=$ORG_ID
--format=json | jq -r '.[] |
select(.subject_claims.sub!="<ID_учетной_записи_которая_входит_в_
список_валидных_исключений>")' | jq -r 'select(.subject_claims.
federation | not)'
```

- Если в списке нет учётных записей, то рекомендация выполнена. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Удалите из вашей организации все учётные записи с Яндекс ID, кроме случаев из списка допустимых исключений.

1.3 Только необходимые администраторы управляют членством в IAM-группах

Для управления доступом к ресурсам удобно использовать группы пользователей. Необходимо контролировать права доступа к самой группе как к ресурсу. Пользователь с правами доступа к группе может управлять членством других пользователей. Случаи, в которых пользователь получает такие права:

- Пользователю назначена роль organization-manager.groups.memberAdmin на организацию.
- Пользователю назначена роль organization-manager.groups.memberAdmin на конкретную группу как на ресурс.
- Пользователю назначена роль organization-manager.organizations.owner или admin или другая привилегированная роль на всю организацию.
- Пользователю назначена роль admin или editor на конкретную группу как на ресурс (не рекомендованный сценарий).

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в вашем браузере.
- Перейдите во вкладку **Все сервисы** → **Cloud Organization** → **Группы** → Выберите нужную группу → **Права доступа к группе**.

- Нажмите тумблер **Наследуемые роли**.
- Если в списке отсутствуют учётные записи, которые не должны иметь прав управления членством в группе, то рекомендация выполнена. Если нет, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Удалите права на доступ к группе у учётных записей, которым это не требуется.

1.4 Используются сервисные роли вместо примитивных: admin, editor, viewer

Принцип минимальных привилегий требует назначать минимально необходимые для работы роли. Не рекомендуется использовать примитивные роли admin, editor и viewer, действующие во всех сервисах, так как это противоречит принципу минимальных привилегий. Для более избирательного управления доступом и реализации принципа минимальных привилегий используйте сервисные роли, которые содержат разрешения только для определенного типа ресурсов в указанном сервисе. Со списком всех сервисных ролей можно ознакомиться на странице Роли сервиса IAM.

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в вашем браузере.
- Выберите **Все сервисы** → **Cloud Organization** → **Пользователи**.
- Если у всех учётных записей в колонке **Права доступа** отсутствуют примитивные роли admin, editor и viewer, то рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».
- Далее перейдите в общее меню облака (нажать на облако в исходном меню облака). Выберите вкладку **Права доступа**.
- Если у всех учётных записей в колонке **Роли** отсутствуют примитивные роли admin, editor и viewer, то рекомендация выполняется. Если нет, перейдите к п. «Инструкции и решения по выполнению».
- Далее перейдите в каждый каталог каждого облака и по аналогии перейдите во вкладку **Права доступа**.
- Если у всех учётных записей в колонке роли отсутствуют примитивные роли: admin, editor, viewer, то рекомендация выполняется. Если нет, перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска учётных записей с назначенными примитивными ролями на уровне организации:

```
export ORG_ID=<ID_организации>
yc organization-manager organization list-access-bindings
--id=${ORG_ID}
--format=json | jq -r '.[] | select(.role_id=="admin" or
.role_id=="editor" or .role_id=="viewer")'
```

- Если в списке отсутствуют учётные записи, то рекомендация выполнена. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

- Выполните команду для поиска учётных записей с назначенными примитивными ролями на уровне облаков:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list
--organization-id=${ORG_ID}
--format=json | jq -r '.[].id');
do yc resource-manager cloud list-access-bindings
--id=$CLOUD_ID
--format=json | jq -r '.[] | select(.role_id=="admin" or
.role_id=="editor" or .role_id=="viewer")'
done
```

- Если в списке отсутствуют учётные записи, то рекомендация выполнена. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

- Выполните команду для поиска учётных записей с назначенными примитивными ролями на уровне всех каталогов в ваших облаках:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list \
--organization-id=${ORG_ID} \
--format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list \
--cloud-id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do yc resource-manager folder list-access-bindings \
--id=$FOLDER_ID \
--format=json | jq -r '.[] | select(.role_id=="admin" or
.role_id=="editor" or .role_id=="viewer")' \
done; \
done
```

- Если в списке отсутствуют учётные записи, то рекомендация выполнена. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Проанализируйте найденные учётные записи с назначенными примитивными ролями: admin, editor и viewer и замените их на сервисные гранулярные роли в соответствии с вашей матрицей ролей.

1.5 Облачные сущности с сервисными аккаунтами учтены и ограничены

Сервисный аккаунт — аккаунт, от имени которого программы могут управлять ресурсами в Yandex Cloud. Сервисный аккаунт служит для выполнения запросов от имени приложения.

- Не используйте вместо сервисных аккаунтов аккаунты сотрудников. Например, если сотрудник уволится или сменит подразделение, его аккаунт потеряет права, что может привести к сбою приложения.
- Не записывайте ключи сервисных аккаунтов напрямую в код приложения, конфигурационные файлы или переменные окружения.

При использовании сервисных аккаунтов:

- Применяйте механизм назначения сервисного аккаунта виртуальной машине и получения токена через сервис метаданных.
- Дополнительно: Настройте локальный файрвол на ВМ так, чтобы только необходимые процессы и пользователи системы имели доступ к сервису метаданных (IP-адрес: 169.254.169.254). Пример блокирования доступа от всех пользователей, кроме указанного (в данном случае — root):

```
sudo iptables --append OUTPUT --proto tcp \
--destination 169.254.169.254 --match owner ! --uid-owner root \
--jump REJECT
```

Облачные сущности, на которые назначены сервисные аккаунты должны быть учтены и ограничены, т.к., например, если сервисный аккаунт назначен на ВМ, то злоумышленник может получить токен сервисного аккаунта из сервиса метаданных изнутри ВМ.

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в вашем браузере.
- Перейдите в нужный каталог и откройте настройки нужной ВМ.
- Нажмите **Изменить**.
- На экране появится информация о сервисном аккаунте.
- Повторите действия для всех ВМ во всех каталогах.

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска ВМ, на которые назначены сервисные аккаунты в вашей организации:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for VM_ID in $(yc compute instance list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id'); \
do yc compute instance get --id=$VM_ID --format=json | jq -r '. |
select(.service_account_id)' | jq -r '.id'
done;
done;
done
```

- Если в списке отсутствуют строки или показаны только учтённые сущности, то рекомендация выполнена. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Удалите сервисные аккаунты у облачных сущностей, которым они не требуются.

1.6 В сервисе метаданных ВМ отсутствуют облачные ключи в открытом виде

Не записывайте ключи сервисных аккаунтов и другие ключи в метаданные виртуальной машины напрямую. Используйте механизм назначения сервисного аккаунта виртуальной машине и получения токена через сервис метаданных. Чувствительные данные могут находиться в любом поле метаданных, но самое распространённое — user-data (за счёт использования в утилите cloud-init).

Ознакомьтесь со списком всех регулярных выражений для поиска секретов учётных данных облачных аккаунтов:

- **yandex_cloud_iam_cookie_v1 :** `c1\.[A-Z0-9a-z_-]+[=]{0,2}\.[A-Z0-9a-z_-]{86}[=]{0,2}` Yandex.Cloud Session Cookie
- **yandex_cloud_iam_token_v1 :** `t1\.[A-Z0-9a-z_-]+[=]{0,2}\.[A-Z0-9a-z_-]{86}[=]{0,2}` Yandex.Cloud IAM token

- **yandex_cloud_iam_api_key_v1 :** `AQVN[A-Za-z0-9_\-]{35,38}`
  Yandex.Cloud API Keys (Speechkit, Vision, Translate)
- **yandex_passport_oauth_token :** `y[0-6]_[-_A-Za-z0-9]{55}`
  Yandex Passport OAuth token
- **yandex_cloud_iam_access_secret :** `YC[a-zA-Z0-9_\-]{38}`
  Yandex.Cloud AWS API compatible Access Secret

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска облачных ключей в сервисе метаданных в открытом виде, на примере Yandex Cloud AWS API Compatible Access Secret:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list
--organization-id=${ORG_ID}
 --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list
--cloud-id=$CLOUD_ID
--format=json | jq -r '.[].id');
do for VM_ID in $(yc compute instance list
--folder-id=$FOLDER_ID
--format=json | jq -r '.[].id');
do yc compute instance get
--id=$VM_ID
--full
--format=json | jq -r '. | select(.metadata."user-data")|
.metadata."user-data" | \ match("YC[a-zA-Z0-9_\\-]{38}") | .string' &&
echo $VM_ID
done;
done;
done
```

- Если в списке отсутствуют строки, то рекомендация выполнена.
  Если нет, перейдите к п. «Инструкции и решения по выполнению».
- Выполните команду для поиска облачных ключей в сервисе метаданных в открытом виде, на примере Yandex Cloud IAM token:

```
export ORG_ID=<ID_организации> \
for CLOUD_ID in $(yc resource-manager cloud list \
--organization-id=${ORG_ID} \
--format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list \
--cloud-id=$CLOUD_ID \
--format=json | jq -r '.[].id'); \
do for VM_ID in $(yc compute instance list \
--folder-id=$FOLDER_ID \
--format=json | jq -r '.[].id'); \
do yc compute instance get \
--id fhm2i4a72v44kdqaqhid \
--full \
```

```
--format=json | jq -r '. | select(.metadata."user-data")|
.metadata."user-data" | \ match("t1\\.[A-Z0-9a-z_-]+[=]{0,2}\\.[A-Z0-
9a-z_-]{86}[=]{0,2}") | .string' \
done; \
done; \
done
```

- Если в списке отсутствуют строки, то рекомендация выполнена.
  Если нет, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Удалите ключи из метаданных ВМ, у которых были найдены отклонения.

1.7 На ВМ отключено получение токена через AWS IMDSv1

В облаке есть сервис метаданных, предоставляющий сведения о работе виртуальных машин.

Изнутри виртуальной машины метаданные доступны в следующих форматах:

- Google Compute Engine (поддерживаются не все поля).
- Amazon EC2 (поддерживаются не все поля).

Формат Amazon EC2 Instance Metadata Service version 1 (IMDSv1) имеет ряд недостатков. Наиболее критичный из них — это риск компрометации токена сервисного аккаунта через сервис метаданных с помощью SSRF-атаки. Подробности в официальном блоге AWS. В связи с этим AWS выпустили вторую версию сервиса метаданных — IMDSv2.

В Yandex Cloud пока нет поддержки второй версии, поэтому строго рекомендуется технически отключать возможность получения токена сервисного аккаунта через Amazon EC2 сервис метаданных.

Сервис метаданных Google Compute Engine использует дополнительный заголовок для защиты от SSRF и повышения безопасности.

Отключить получение токена сервисного аккаунта через Amazon EC2 сервис метаданных можно с помощью параметра ВМ aws_v1_http_token:DISABLED.

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска ВМ, у которых включено использование IMDSv1 для получения токена:

```
export ORG_ID=<ID_организации> \
for CLOUD_ID in $(yc resource-manager cloud list \
--organization-id=${ORG_ID} \
--format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list \
--cloud-id=$CLOUD_ID  \
--format=json | jq -r '.[].id'); \
do for VM_ID in $(yc compute instance list \
--folder-id=$FOLDER_ID \
--format=json | jq -r '.[].id'); do yc compute instance get  \
--id=$VM_ID \
--format=json | jq -r '. |
select(.metadata_options.aws_v1_http_token=="ENABLED")' | \
jq -r'.id' \
done; \
done; \
done
```

- Если в списке отсутствуют строки, то рекомендация выполнена.
  Если нет, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

В блоке metadata_options задайте параметру aws_v1_http_token значение DISABLED у найденных ВМ:

```
yc compute instance update fhm2i4a72v44kdqaqhid \
  --metadata-options aws-v1-http-token=DISABLED
```

1.8 Сервисным аккаунтам назначены минимальные привилегии

Следуйте принципу минимальных привилегий и назначайте сервисному аккаунту только те роли, которые необходимы для функционирования приложения.

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в вашем браузере.
- Перейдите в нужный каталог и откройте вкладку **Сервисные аккаунты**.
- Проверьте список сервисных аккаунтов.
- Повторите действия для других каталогов.

- Перейдите во вкладку **Права доступа** на уровнях облаков и каталогов.

Права доступа на уровне организации можно посмотреть только в YC CLI.

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для отображения всех сервисных аккаунтов организации в формате <id_сервисного_аккаунта>:<имя_сервисного_аккаунта>:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list \
--cloud-id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do for SA in $(yc compute instance list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id'); \ do yc iam service-account list \
--folder-id=$FOLDER_ID --format=json  | jq -r '.[].id + ":" +
.[].name' \
done; \
done; \
done
```

- Выполните команду для отображения всех прав доступа конкретного сервисного аккаунта на организацию:

```
export ORG_ID=<ID_организации>
yc organization-manager organization list-access-bindings --
id=${ORG_ID} \
--format=json | jq -r '.[] | select(.subject.type=="serviceAccount")'
```

- Просмотрите права доступа сервисного аккаунта на всех облаках:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do yc resource-manager cloud list-access-bindings \
--id=$CLOUD_ID  --format=json | jq -r '.[] |
select(.subject.type=="serviceAccount")' \
 && echo $CLOUD_ID \
done;
```

- Просмотрите права доступа сервисного аккаунта на всех каталогах:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list \
--cloud-id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do yc resource-manager folder list-access-bindings \
--id=$FOLDER_ID  --format=json | jq -r '.[] |
select(.subject.type=="serviceAccount")' && \
 echo $FOLDER_ID \
done; \
done
```

- Если в списках отсутствуют избыточные права, то рекомендация выполняется. Если нет, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:
Удалите избыточные права у сервисного аккаунта с помощью сервиса IAM.

1.9 Только доверенные администраторы имеют доступ к сервисным аккаунтам

Существует возможность назначать права на использование сервисного аккаунта от имени другого пользователя или сервисного аккаунта.

Следуйте принципу минимальных привилегий при выдаче доступа к сервисному аккаунту как к ресурсу: при наличии у пользователя прав на сервисный аккаунт, у него также появляется доступ и ко всем его правам. Назначайте роли на использование и управление сервисными аккаунтами минимальному кругу пользователей.

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в вашем браузере.
- Перейдите в каждый каталог и откройте вкладку **Сервисные аккаунты**.
- Нажмите на сервисный аккаунт и перейдите во вкладку **Права доступа**.
- Проверьте права, назначенные на сервисный аккаунт.
- Если в списке находятся только валидные администраторы, рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для отображения всех сервисных аккаунтов в облаках:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} \
--format=json | jq -r '.[].id'); \
do yc resource-manager cloud list-access-bindings \
--id=$CLOUD_ID  --format=json | jq -r '.[] |
select(.subject.type=="serviceAccount")' \
&& echo $CLOUD_ID \
done;
```

- Выполните команду для отображения всех прав доступа на конкретный сервисный аккаунт как на ресурс:

```
yc iam service-account list-access-bindings
--id <ID_сервисного_аккаунта>
```

- Если в списке находятся только валидные администраторы, рекомендация выполняется. Если нет, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Удалите избыточные права сервисного аккаунта с помощью сервиса IAM.

Проверка наличия NGFW вместо SG:

- Откройте консоль Yandex Cloud в вашем браузере.
- Перейдите в каждое облако и в каждый каталог и последовательно откройте все диски ВМ.
- В настройках дисков найдите параметр **Продукт Marketplace**.
- Рекомендация выполняется, если в параметрах **Продукт Marketplace** в диске указано одно из названий продуктов NGFW:
  - Check Point CloudGuard IaaS - Firewall & Threat Prevention PAYG;
  - UserGate NGFW.
- Если нет, перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска объектов облака без SG:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list  \
--cloud-id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do for VM_ID in $(yc compute instance list --folder-id=$FOLDER_ID  \
--format=json | jq -r '.[].id'); do yc compute instance get --
id=$VM_ID --format=json | jq -r '. |  \
 select(.network_interfaces[].security_group_ids | not)' | jq -r '.id'
\
done; \
done; \
done
```

- Если выдаётся пустая строка, то рекомендация выполняется. Если выдаётся результат с ID облачного объекта, то перейдите к п. «Инструкции и решения по выполнению».

Проверка наличия NGFW вместо SG:

- Выполните команду для поиска NGFW в облаке (по умолчанию ищет Checkpoint или Usergate. Если используете другой образ, то укажите его):

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list  \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list  \
--cloud-id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do for DISK_ID in $(yc compute disk list  \
--folder-id=$FOLDER_ID --format=json | jq -r '.[].id'); do yc compute
disk get  \
--id=$DISK_ID --format=json | jq -r '. |
select(.product_ids[0]=="f2ecl4ak62mjbl13qj5f" or \
.product_ids[0]=="f2eqc5sac8o5oic7m99k")' | jq -r '.id' \
done; \
done; \
done
```

- Если выдаётся id ВМ с NGFW, то рекомендация выполняется.
  Если выдаётся пустая строка, то перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

- Примените SG на все объекты, на которых SG отсутствуют;
- Для применения SG с помощью Terraform используйте настройку групп безопасности (dev/stage/prod) с помощью Terraform.
- Для использования NGFW установите в Yandex Cloud ВМ межсетевой экран (NGFW): Check Point.
- Инструкция по использованию UserGate NGFW в облаке.
- NGFW в режиме active-passive.

1.10 Выполняется периодическая ротация ключей сервисных аккаунтов

В Yandex Cloud поддерживаются следующие ключи доступа, которые могут быть созданы для сервисных аккаунтов:

- IAM-токены — действуют 12 часов.
- API-ключи — не имеют срока действия.
- Авторизованные ключи — не имеют срока действия.
- Статические ключи доступа, совместимые с AWS API — не имеют срока действия.

Ключи без срока действия требуется ротировать самостоятельно — удалять и создавать новые. Дату создания можно проверить в свойствах ключа. Рекомендуется ротировать ключи как минимум раз в 90 дней, в соответствии со стандартами информационной безопасности, например, PCI DSS.

Проверка в консоли управления:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Перейдите в нужный каталог и откройте вкладку **Сервисные аккаунты**.
- Нажмите на сервисный аккаунт и в разделе **Свойства ключей доступа** проверьте дату создания каждого ключа.
- Повторите действия в каждом из своих каталогов.
- Если даты создания ключей не старше 90 дней, то рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Проверьте дату создания статических ключей, совместимых с AWS API:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for SA in $(yc iam service-account list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id');
do yc iam access-key list --service-account-id=$SA --format=json | jq
-r '.[] | "key_id" + ":" + .id + "," + "sa_id" + ":" +
.service_account_id + "," + "created_at" + ":" + .created_at '
done;
done;
done
```

- Проверьте дату создания авторизованных ключей, совместимых с AWS API:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for SA in $(yc iam service-account list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id');
do yc iam key list --service-account-id=$SA --format=json | jq -r '.[]
| "key_id" + ":" + .id + "," + "sa_id" + ":" + .service_account_id +
"," + "created_at" + ":" + .created_at '
done;
done;
done
```

- Проверьте дату создания API-ключей доступа, совместимых с AWS API:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for SA in $(yc iam service-account list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id');
do yc iam api-key list --service-account-id=$SA --format=json | jq -r
'.[] | "key_id" + ":" + .id + "," + "sa_id" + ":" +
.service_account_id + "," + "created_at" + ":" + .created_at '
done;
done;
done
```

- Если в списке любого типа ключей отсутствуют ключи, у которых дата в поле created_at старше 90 дней, то рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Для ротации ключей в зависимости от их типа воспользуйтесь инструкцией.

1.11 Настроена двухфакторная аутентификация для привилегированных аккаунтов

Рекомендуется использовать двухфакторную аутентификацию (2FA) для доступа к облачной инфраструктуре, чтобы избежать рисков, связанных с компрометацией пользовательских учётных записей. Доступ в консоль Yandex Cloud может быть организован с помощью 2FA.

Чтобы подключить двухфакторную аутентификацию, обратитесь к поставщику удостоверений (identity provider) с поддержкой 2FA и настройте SAML-совместимую федерацию удостоверений. В Yandex Cloud нет своего IdP и задача идентификации пользователей решается с помощью внешних сервисов — Яндекс ID или корпоративных систем, интегрированных с помощью федерации удостоверений. Например, если вы используете IdP системы Active Directory или Keycloak, то настройте 2FA в данных системах. Необходимо настроить 2FA как минимум для привилегированных учётных записей облака.

Для аккаунта Яндекс ID настройте 2FA согласно инструкции.

Проверка в консоли управления:

- Откройте UI Яндекс ID в вашем браузере.
- Перейдите на вкладку Безопасность.

- Проверьте, что указан способ входа с помощью дополнительного ключа.
- Если способ входа с помощью ключа настроен, то рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».
- Если вы используете сторонние IdP, проверьте настройки по инструкциям.

Инструкции и решения по выполнению:

- Двухфакторная аутентификация — Яндекс ID
- KeyCloak — Creating other credentials
- Configure Additional Authentication Methods for AD FS

1.12 Привилегированные роли назначены только доверенным администраторам

К привилегированным пользователям Yandex Cloud относятся аккаунты, которым назначены роли:

- billing.accounts.owner;
- admin, назначенный на платежный аккаунт;
- organization-manager. organizations.owner;
- organization-manager.admin;
- resource-manager.clouds.owner;
- admin, назначенный на организацию;
- admin, назначенный на облако;
- admin, назначенный на каталог.

Роль billing.accounts.owner автоматически выдается при создании платёжного аккаунта и не может быть переназначена другому пользователю.
Роль позволяет выполнять любые действия с платёжным аккаунтом.

Роль billing.accounts.owner может быть назначена только аккаунту Яндекс ID. Аккаунт с ролью billing.accounts.owner используется при настройке способов оплаты и подключении облаков.

Безопасности этого аккаунта следует уделять повышенное внимание, поскольку он обладает значительными полномочиями и не может быть объединён с корпоративным аккаунтом.

Наиболее правильным подходом можно считать отказ от регулярного использования данного аккаунта:

- Используйте его только при первоначальной настройке и внесении изменений.
- На время активного использования данного аккаунта включите двухфакторную аутентификацию (2FA) в Яндекс ID.
- Затем, если вы не используете способ оплаты банковской картой (доступный только для данной роли), назначьте данному аккаунту сложный пароль (сгенерированный с помощью специализированного ПО), отключите 2FA и не используйте этот аккаунт без необходимости.
- После каждого использования меняйте пароль на сгенерированный заново.

Отключать 2FA рекомендуется только для этого аккаунта и в случае, если аккаунт не «закреплён» за конкретным сотрудником. Эта мера нужна, чтобы избежать привязки критически важного аккаунта к личному устройству.

Для управления платежным аккаунтом назначьте роль admin или editor на платёжный аккаунт выделенному сотруднику организации с федеративным аккаунтом.

Для просмотра данных биллинга назначьте роль viewer на платёжный аккаунт выделенному сотруднику организации с федеративным аккаунтом.

Роль organization-manager.organizations.owner по умолчанию получает владелец организации — пользователь, который ее создал. Роль даёт возможность назначать владельцев организации, а также пользоваться всеми полномочиями администратора.

Роль resource-manager.clouds.owner автоматически выдается при создании первого облака в организации. Пользователь с этой ролью может выполнять любые операции с облаком и ресурсами в нем, а также выдавать доступ к облаку другим пользователям: назначать роли и отзывать их.

Назначайте роль resource-manager.clouds.owner и organization-manager.organizations.owner одному или нескольким сотрудникам организации с федеративным аккаунтом. Аккаунту Яндекс ID, с которым создано облако, назначьте сложный пароль и используйте только в случае крайней необходимости, например, при поломке федерации.

Федеративный аккаунт с одной из привилегированных ролей, указанных выше, необходимо всесторонне защитить:

- Включите двухфакторную аутентификацию.
- Запретите аутентификацию с устройств, не управляемых организацией.
- Настройте мониторинг попыток входа и задайте пороги предупреждений.

Назначайте роли admin на облака, каталоги и платежные аккаунты федеративным аккаунтам. Минимизируйте количество аккаунтов с этими ролями и регулярно перепроверяйте потребность в этих ролях для тех аккаунтов, которым они назначены.

Проверка в консоли управления:

Проверка ролей на биллинг:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Перейдите во вкладку **Биллинг**.
- Проверьте кому назначены роли: billing.accounts.owner, admin.
- Проверка ролей на организацию:
- Откройте консоль управления Yandex Cloud в вашем браузере.
- Перейдите во вкладку **Все сервисы** → **Cloud Organization** → **Пользователи**.
- Проверьте кому назначены роли: admin, organization-manager.organizations.owner, organization-manager.admin, resource-manager.clouds.owner.

Проверка ролей на облако:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Перейдите в общее меню облака: нажмите на облако в исходном меню облака. Выберите вкладку **Права доступа**.
- Проверьте кому назначены роли: admin, resource-manager.clouds.owner.

Проверка ролей на каталоге:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Далее перейдите в каждый каталог каждого облака и по аналогии перейдите во вкладку **Права доступа**.
- Проверьте кому назначена роль admin.

- Если все привилегированные роли назначены доверенным администраторам, то рекомендация выполняется.
  Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Найдите привилегированные права на уровне организации:

```
export ORG_ID=<ID_организации>
yc organization-manager organization list-access-bindings --
id=${ORG_ID} \
  --format=json | jq -r '.[] | select(.role_id=="admin" or \
  .role_id=="organization-manager.organizations.owner" or \
  .role_id=="organization-manager.admin" or .role_id==
  "resource-    manager.clouds.owner")'
```

- Найдите привилегированные права на уровне облаков:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do yc resource-manager cloud list-access-bindings \
--id=$CLOUD_ID --format=json | jq -r '.[] |  \
select(.role_id=="admin" or .role_id=="resource-
manager.clouds.owner")' && \
echo $CLOUD_ID \
done
```

- Выполните команду для поиска привилегированных прав на уровне всех каталогов в ваших облаках:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list \
--cloud-id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do yc resource-manager folder list-access-bindings \
--id=$FOLDER_ID --format=json | jq -r '.[] |
select(.role_id=="admin")' \
 && echo $FOLDER_ID \
done; \
done
```

- Если все привилегированные роли назначены доверенным администраторам, то рекомендация выполняется.
  Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Если обнаружены роли, которые назначены недоверенным администраторам, необходимо провести расследование и удалить лишние права.

1.13 Для локальных пользователей управляемых БД задан стойкий пароль

Для работы с управляемыми БД в облаке на прикладном уровне помимо сервисных IAM ролей создается отдельный локальный пользователь — владелец БД. В отношении него действует следующая парольная политика:

- пароль должен содержать цифры, буквы в верхнем регистре, буквы в нижнем регистре, специальные символы.
- длина пароля — не менее 8 символов.

Ручная проверка:

Убедитесь, что пароль периодически ротируется в ручном режиме и соответствует парольным политикам вашей компании. Данный пароль хранится на стороне клиента и недоступен для просмотра в консоли управления, CLI и API.

1.14 Доступ для подрядных организаций и третьих лиц контролируется

Если вы предоставляете доступ к облакам внешним подрядным организациям, соблюдайте следующие меры безопасности:

- Назначайте права сотрудникам подрядных организаций с учетом принципа минимальных привилегий.
- По возможности создавайте отдельный аккаунт для сотрудников внешних организаций в вашем корпоративном IdP и назначайте ему необходимые политики.
- Предъявляйте требования к бережному обращению с секретами аккаунта.
- Перепроверяйте необходимость доступа внешних пользователей к вашей облачной инфраструктуре.

Ручная проверка:
Проверьте все учётные записи в вашей организации и убедитесь, что вы знаете обо всех учётных записях подрядных организаций и третьих лиц и выполняете рекомендации выше.

1.15 Используется корректная ресурсная модель

При разработке модели доступа для вашей инфраструктуры рекомендуется использовать следующий подход:

- Как минимум одна организация для компании.
- Группировать ресурсы по назначению и помещать их в отдельные каталоги. Для наиболее строгой изоляции — в отдельные облака.
- Все критичные ресурсы помещайте в отдельные каталоги или облака. К критичным относятся в том числе ресурсы, которые связаны с обработкой платежных данных, персональных данных, данных коммерческой тайны.
- Группы ресурсов, которые требуют различного административного доступа, например, DMZ, CDE, security, backoffice и т. д., поместите в разные каталоги или облака.
- При разработке приложений, необходимо разделять тестовые и промышленные среды.
- Общие ресурсы (например, сеть и группы безопасности) поместите в отдельный каталог для разделяемых ресурсов (в случае разделения компонентов по каталогам).

Ручная проверка:

Проанализируйте вашу ресурсную модель и убедитесь, что рекомендации, указанные выше, выполняются.

1.16 На ресурсах в организации отсутствует «публичный доступ»

В Yandex Cloud существует возможность предоставлять публичный доступ на ресурсы. Публичный доступ предоставляется путем назначения прав доступа для системных групп (allAuthenticatedUsers, allUsers).

Описание системных групп:

- allAuthenticatedUsers — все пользователи, прошедшие аутентификацию. Это все зарегистрированные пользователи или сервисные аккаунты Yandex Cloud: как из ваших облаков, так и из облаков других пользователей.
- allUsers — любой пользователь, аутентификация не требуется.

Важно: Сейчас allUsers поддерживается только в сервисах:
Object Storage при управлении доступом с помощью ACL,
Container Registry, Cloud Functions.
В остальных сервисах назначение роли для группы allUsers
эквивалентно назначению роли для allAuthenticatedUsers.

Убедитесь, что на ваши ресурсы — облака, каталоги, бакеты и т.д.,
не предоставлен публичный доступ для этих групп.

Проверка в консоли управления:

Проверка ролей в облаке:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Далее перейдите в общее меню облака (нажать на облако в исходном меню облака). Выберите вкладку **Права доступа**.
- Проверьте, есть ли среди пользователей allUsers и allAuthenticatedUsers.

Проверка ролей в каталоге:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Перейдите в нужный каталог нужного облака и перейдите во вкладку **Права доступа**.
- Проверьте, есть ли среди пользователей allUsers и allAuthenticatedUsers.
- Повторите действия для всех каталогов всех ваших облаков.

Проверка ролей в Object Storage:

- Откройте консоль управлния Yandex Cloud в вашем браузере.
- Перейдите в нужное облако и найдите сервис **Object Storage**.
- Нажмите на три точки напротив бакета и проверьте ACL бакета на наличие allUsers и allAuthenticatedUsers.
- Зайдите внутрь бакета и проверьте ACL на каждый объект бакета на наличие allUsers и allAuthenticatedUsers.
- Зайдите в глобальные настройки бакета и в разделе **Доступ на чтение объектов** проверьте, что параметр **Публичный** выключен.
- Повторите действия для всех бакетов и объектов во всех ваших облаках.

Проверка ролей в Container Registry:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Далее перейдите в каждое облако и найдите сервис **Container Registry**.
- Перейдите в необходимый реестр и слева нажмите **Права доступа**.
- Проверьте, есть ли среди пользователей allUsers и allAuthenticatedUsers.
- Повторите действия для всех ваших облаков.

Проверка ролей в Cloud Functions:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Далее перейдите в каждое облако и найдите сервис **Cloud Functions**.
- Перейдите во все облачные функции и проверьте, что параметр **Публичный доступ** выключен.
- Если в каждом указанном ресурсе нет субъектов allUsers и allAuthenticatedUsers, то рекомендация выполняется. Если есть, перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска учетных записей с назначенными примитивными ролями на уровне организации:

```
export ORG_ID=<ID_организации>
yc organization-manager organization list-access-bindings --
id=${ORG_ID} \
--format=json | jq -r '.[] | select(.role_id=="admin" or \
.role_id=="organization-manager.organizations.owner" or \
.role_id=="organization-manager.admin" or .role_id=="resource-
manager.clouds.owner")'
```

- Выполните команду для поиска прав доступа allUsers, allAuthenticatedUsers на уровне облаков:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do yc resource-manager cloud list-access-bindings \
--id=$CLOUD_ID --format=json | jq -r '.[] | \
select(.subject.id=="allAuthenticatedUsers" or
.subject.id=="allUsers")' && \
echo $CLOUD_ID \
done
```

- Выполните команду для поиска прав доступа allUsers, allAuthenticatedUsers на уровне каталогов:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list  \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list \
 --cloud-id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do yc resource-manager folder list-access-bindings  \
--id=$FOLDER_ID --format=json | jq -r '.[] |  \
select(.subject.id=="allAuthenticatedUsers" or
.subject.id=="allUsers")' && \
echo $FOLDER_ID \
done; \
done
```

- Выполните команду для поиска прав доступа allUsers, allAuthenticatedUsers на уровне Container Registry во всех каталогах:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list  \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list  \
--cloud-id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do for CR in $(yc container registry list  \
--folder-id=$FOLDER_ID --format=json | jq -r '.[].id'); \
do yc container registry list-access-bindings  \
--id $CR --format=json | jq -r '.[] |  \
select(.subject.id=="allAuthenticatedUsers" or
.subject.id=="allUsers")' &&  \
echo $CR \
done; \
done; \
done
```

- Выполните команду для поиска прав доступа allUsers, на уровне Cloud Functions во всех каталогах:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list  \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list \
--cloud-id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do for FUN in $(yc serverless function list \
--folder-id=$FOLDER_ID --format=json | jq -r '.[].id'); \ \
do yc serverless function  list-access-bindings  \
--id $FUN --format=json | jq -r '.[] |  \
select(.subject.id=="allAuthenticatedUsers" or
.subject.id=="allUsers")' &&  \
echo $FUN \
done; \
done; \
done
```

- Если в каждом указанном ресурсе отсутствуют субъекты: allUsers, allAuthenticatedUsers то рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению»

Инструкции и решения по выполнению:

Если обнаружено наличие прав доступа у allUsers, allAuthenticatedUsers, необходимо удалить данные права.

1.17 Контактные данные ответственного за организацию актуальны

В Yandex Cloud при регистрации облака клиент указывает контактные данные. Например, электронная почта используется для оповещений, связанных с инцидентами, плановыми работами и т.д.

Например, если со стороны облака были обнаружены аномальные активности в организации клиента или облачные ключи IAM становятся доступными во внешних репозиториях GitHub, клиенту будет направлено оповещение. Эта возможность реализована с помощью участия Yandex Cloud в программе Github Secret scanning partner program, а также анализа секретов в поиске Яндекса. В случае компрометации ключей в публичном репозитории, удалите секрет из репозитория, его истории и отзовите ключи.

Убедитесь, что контактные данные актуальны и указанный почтовый ящик рассылает сообщения нескольким ответственным.

Проверка в консоли управления:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Перейдите во вкладку **Биллинг**.
- Перейдите во вкладку **Данные аккаунта**.
- В самом низу будет кнопка Редактировать данные в Яндекс Балансе.
- Проверьте указанные контактные данные.
- Если указанные данные актуальны, то рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Укажите актуальные контактные данные по инструкции.

1.18 Таймаут жизни cookie в федерации меньше 6 часов

В настройках федерации удостоверений необходимо убедиться, что значение параметра **Время жизни cookie** меньше либо равно 6 часов.

Это необходимо, чтобы минимизировать риск компрометации рабочих станций пользователей облака.

Проверка в консоли управления:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Перейдите во вкладку **Organizations**.
- Далее перейдите во вкладку **Федерации** и выберите вашу федерацию.
- Найдите параметр **Время жизни cookie**.
- Если значение этого параметра меньше либо равно 6 часам, то рекомендация выполняется. Если нет, перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска учетных записей с назначенными примитивными ролями на уровне организации:

```
export ORG_ID=<ID_организации>
for FED in $(yc organization-manager federation saml list  \
--organization-id=${ORG_ID} -format=json | jq -r '.[].id'); \
do yc organization-manager federation saml get  \
--id bpfdshe1skaqcjp6uc50 -format=json | jq -r '. | \
 select(.cookie_max_age>"21600s")' \
done
```

- Если выдается пустая строка, то рекомендация выполняется. Если выдаётся результат с настройкой текущей федерации, где параметр cookie_max_age > 21600s, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Задайте значение параметра «Время жизни cookie» равным 6 часам (21600 секундам) или меньше.

1.19 Токен для облачных функций и ВМ выдаётся через сервисный аккаунт

Для получения IAM-токена в ходе выполнения функции необходимо назначить функции сервисный аккаунт. В этом случае функция получит IAM-токен с помощью встроенных механизмов Yandex Cloud, без необходимости передачи каких-либо секретов в функцию извне. Аналогично и для ВМ.

Ручная проверка:

Проанализируйте все ваши ВМ и облачные функции на предмет созданных вручную токенов сервисных аккаунтов. Правильно использовать токены путём назначения сервисного аккаунта на сущность и использовать токен аккаунта изнутри, через сервис метаданных.

2. Сетевая безопасность

Введение

В этом разделе представлены рекомендации пользователям по настройкам безопасности в Yandex Virtual Private Cloud.

Подробно о том, как настроить сетевую инфраструктуру, рассказывается в вебинаре Как работает сеть в Облаке.

Чтобы изолировать приложения друг от друга, поместите ресурсы в разные группы безопасности, а если требуется наиболее строгая изоляция — в разные сети. Трафик внутри сети по умолчанию разрешен, а между сетями — нет. Трафик между сетями можно передавать только через ВМ с двумя сетевыми интерфейсами в разных сетях, VPN или сервис Cloud Interconnect.

2.1 Для объектов облака используется межсетевой экран или группы безопасности

Встроенный механизм групп безопасности позволяет управлять доступом виртуальных машин к ресурсам и группам безопасности Yandex Cloud или ресурсам в интернете. Группа безопасности — это набор правил для входящего и исходящего трафика, который можно назначить на сетевой интерфейс виртуальной машины. Группы безопасности работают как stateful firewall, то есть отслеживают состояние сессий: если правило разрешает создать сессию, ответный трафик будет автоматически разрешен. Инструкцию по настройке групп безопасности см. в разделе Создать группу безопасности. Указать группу безопасности можно в настройках ВМ.

Группы безопасности могут использоваться для защиты:

- виртуальных машин;
- управляемых баз данных;
- балансировщиков в Application Load balancer;
- кластеров в Managed Service for Kubernetes.

Список доступных сервисов расширяется.

Вы можете управлять сетевым доступом без групп безопасности, например, с помощью отдельной виртуальной машины — межсетевой экран на основе образа NGFW из Cloud Marketplace, либо своего собственного образа.

Использование NGFW может быть критично для тех клиентов, которым необходима следующая функциональность:

- составление логов сетевых соединений;
- потоковый анализ трафика на предмет зловредного контента;
- обнаружение сетевых атак по сигнатурам;
- другая функциональность классических NGFW-решений.

Убедитесь, что в ваших облаках используются группы безопасности на каждом объекте облака, либо используется отдельная ВМ NGFW из Cloud Marketplace, либо по принципу «bring your own image» («используй своё устройство» — принцип, позволяющий использовать свое оборудование или образы системы).

Проверка в консоли управления:

Проверка наличия групп безопасности на объектах:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Перейдите в каждое облако и в каждый каталог и последовательно открывайте все перечисленные ресурсы в пункте «Объекты, на которые возможно применить группы безопасности».
- В настройках объектов найдите параметр **Группа безопасности** и убедитесь, что назначена хотя бы одна группа безопасности.
- Если в параметрах каждого объекта, который поддерживает SG указана хотя бы одна SG то рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Проверка наличия NGFW вместо SG:

- Откройте консоль управления Yandex Cloud в вашем браузере.
- Перейдите в каждое облако и в каждый каталог и последовательно откройте все диски ВМ.
- В настройках дисков найдите параметр **Продукт Marketplace**.
- Если в параметрах **Продукт Marketplace** в диске указано одно из названий продуктов NGFW: «Check Point CloudGuard IaaS - Firewall & Threat Prevention PAYG», «UserGate NGFW», рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска объектов облака без SG:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list  \
--cloud-id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do for VM_ID in $(yc compute instance list --folder-id=$FOLDER_ID  \
--format=json | jq -r '.[].id'); do yc compute instance get --
id=$VM_ID --format=json | jq -r '. |  \
 select(.network_interfaces[].security_group_ids | not)' | jq -r '.id'
\
done; \
done; \
done
```

- Если выдается пустая строка, то рекомендация выполняется. Если выдается результат с ID облачного ресурса, то перейдите к п. «Инструкции и решения по выполнению».

Проверка наличия NGFW вместо группы безопасности:

- Выполните команду для поиска NGFW в облаке (по умолчанию команда ищет Checkpoint или Usergate. Если используете свой образ, укажите его):

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list  \
--organization-id=${ORG_ID} --format=json | jq -r '.[].id'); \
do for FOLDER_ID in $(yc resource-manager folder list  \
--cloud-id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do for DISK_ID in $(yc compute disk list  \
--folder-id=$FOLDER_ID --format=json | jq -r '.[].id'); do yc compute
disk get  \
--id=$DISK_ID --format=json | jq -r '. |
select(.product_ids[0]=="f2ecl4ak62mjbl13qj5f" or \
.product_ids[0]=="f2eqc5sac8o5oic7m99k")' | jq -r '.id' \
done; \
done; \
done
```

- Если выдается id ВМ с NGFW, то рекомендация выполняется. Если выдается пустая строка, то перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

- Примените SG на все объекты, на которых SG отсутствуют:
- Для применения SG с помощью Terraform используйте настройку групп безопасности (dev/stage/prod) с помощью Terraform.

- Для использования NGFW установите в Yandex Cloud ВМ межсетевой экран (NGFW): Check Point.
- Инструкция по использованию UserGate NGFW в облаке.
- NGFW в режиме active-passive.

2.2 Как минимум одна Группа безопасности существует в VPC

Для возможности назначения SG на облачные объекты в VPC должна существовать как минимум одна Группа безопасности. Дополнительно существует функция создания группы безопасности по умолчанию — такая группа назначается облачным объектам при подключении к подсетям, если у них нет ни одной группы безопасности. Убедитесь в том, что, хотя бы одна группа безопасности существует в каждой сети.

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в вашем браузере.
- Перейдите в каждое облако, далее в каждый каталог и в каждую VPC.
- Перейдите в раздел **Группы безопасности**.
- Если обнаружена как минимум одна группа безопасности для каждой VPC, либо группа по умолчанию, рекомендация выполняется. Если нет, перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска каталогов без наличия SG:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do echo "SG_ID: " && yc vpc security-group list --folder-id=$FOLDER_ID
--format=json | jq -r '.[] | select(.id)' | jq -r '.id'  && echo
"FOLDER_ID: " $FOLDER_ID && echo "-----"
done;
done
```

- Если у каждого сочетания SG_ID напротив FOLDER_ID, в которой она находится указаны ID, рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Создайте группу безопасности в каждой VPC с ограниченными правилами доступа, чтобы ее можно было назначать на облачные объекты.

2.3 В Группах безопасности отсутствует слишком широкое правило доступа

В группе безопасности существует возможность открыть сетевой доступ для абсолютно всех ip адресов интернета и также по всем диапазонам портов. Опасное правило выглядит следующим образом:

- Диапазон портов: 0-65535 либо пусто.
- Протокол: любой либо TCP/UDP.
- Источник: CIDR.
- CIDR блоки: 0.0.0.0/0 (доступ со всех адресов) или ::/0 (ipv6).

> **Важно**: если диапазон портов не указан, то считается, что доступ предоставляется по всем портам (0-65535)

Открывать сетевой доступ необходимо только по тем портам, которые требуются для работы вашего приложения и для тех адресов, с которых необходимо подключаться к вашим объектам.

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в вашем браузере.
- Перейдите в каждое облако, далее в каждый каталог и в каждую VPC.
- Перейдите в раздел **Группы безопасности**.
- Если не обнаружено ни одной группы безопасности, в которой есть правила сетевого доступа, разрешающие доступ по всем портам для всех адресов (интерпретация указана выше) то рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Найдите группы безопасности с опасным правилом доступа:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do echo "SG_ID: " && yc vpc security-group list --folder-id=$FOLDER_ID
\
--format=json | jq -r '.[] | select(.rules[].direction=="INGRESS" and
.rules[].ports.to_port=="65535" and
.rules[].cidr_blocks.v4_cidr_blocks[]=="0.0.0.0/0")' | jq -r '.id'  \
&& echo "FOLDER_ID: " $FOLDER_ID && echo "-----"
done;
done
```

- Если SG_ID напротив FOLDER_ID указано пустое значение, рекомендация выполняется. Если вы видите не пустое SG_ID, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Удалите опасное правило в каждой SG или отредактируйте, указав доверенные IP-адреса.

2.4 Доступ по управляющим портам открыт только для доверенных IP-адресов

Рекомендуется открывать доступ к вашей облачной инфраструктуре по управляющим портам только с доверенных IP-адресов. Убедитесь, что в ваших правилах доступа в рамках SG отсутствуют широкие правила доступа по управляющим портам:

- Диапазон портов: 22, 3389, или 21
- Протокол: TCP
- Источник: CIDR
- CIDR блоки: 0.0.0.0/0 (доступ со всех адресов) или ::/0 (ipv6)

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в вашем браузере.
- Перейдите в каждое облако, далее в каждый каталог и в каждую VPC.
- Перейдите в раздел **Группы безопасности**.
- Если не обнаружено ни одной группы безопасности, в которой есть правила сетевого доступа, разрешающие доступ по управляющим портам для всех адресов (интерпретация указана выше), то рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска групп безопасности с опасным правилом доступа:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do echo "SG_ID: " && yc vpc security-group list --folder-id=$FOLDER_ID
\
--format=json | jq -r '.[] | select(.rules[].direction=="INGRESS" and
(.rules[].ports.to_port=="22" or .rules[].ports.to_port=="3389" or
.rules[].ports.to_port=="21") and
.rules[].cidr_blocks.v4_cidr_blocks[]=="0.0.0.0/0")' | jq -r '.id'  \
&& echo "FOLDER_ID: " $FOLDER_ID && echo "-----"
done;
done
```

- Если SG_ID напротив FOLDER_ID указано пустое значение, то рекомендация выполняется. Если вы видите не пустое SG_ID, то перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Удалите опасное правило в каждой SG или укажите доверенные IP-адреса.

2.5 Включена защита от DDoS атак

В Yandex Cloud существует базовая защита от DDoS и расширенная. Необходимо убедиться, что у вас используется как минимум базовая защита.

- Yandex DDoS Protection — это компонент сервиса VPC для защиты облачных ресурсов от DDoS-атак. DDoS Protection предоставляется в партнёрстве с Qrator Labs. Вы можете включать её самостоятельно на внешний IP адрес через инструменты управления облаком. Работает до L4 уровня модели OSI.
- Расширенная защита от DDoS-атак — работает на 3 и 7 уровнях модели OSI. Вы также можете отслеживать показатели нагрузки, параметры атак и подключить Solidwall WAF в личном кабинете Qrator Labs.
  Чтобы включить расширенную защиту, обратитесь к вашему менеджеру или в техническую поддержку.

Проверка из UI консоли (Базовая защита):

- Откройте консоль Yandex Cloud в вашем браузере.
- Откройте все созданные сети.

- Перейдите в раздел **IP-адреса**.
- Если у всех публичных адресов в столбце «Защита от DDoS-атак» установлено значение «Включена», то рекомендация выполняется. Если нет, то перейдите к п. «Инструкции и решения по выполнению».

Ручная проверка (Расширенная защита):

Обратитесь к вашему менеджеру со стороны облака и уточните подключена ли у вас «Расширенная защита от DDoS-атак».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска IP адресов без защиты от DDOS:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do echo "Address_ID: " && yc vpc address list --folder-id=$FOLDER_ID \
--format=json | jq -r '.[] |
select(.external_ipv4_address.requirements.ddos_protection_provider=="
qrator" | not)' | jq -r '.id' \
&& echo "FOLDER_ID: " $FOLDER_ID && echo "-----"
done;
done
```

- Если Address_ID напротив FOLDER_ID указано пустое значение, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению»

Инструкции и решения по выполнению:

- Вебинар Защита от DDoS в Yandex Cloud.
- Все материалы по защите от DDoS в Облаке.

2.6 Используется защищённый удалённый доступ

Чтобы обеспечить удаленное подключение администраторов к облачным ресурсам, используйте одно из следующих решений:

- Site-to-site VPN между удаленной площадкой (например, вашим офисом) и облаком. В качестве шлюза для удаленного доступа используйте ВМ с функцией site-to-site VPN на основе образа из Cloud Marketplace.

Варианты настройки:

- Создание туннеля IPSec VPN с использованием демона strongSwan.
- Создание site-to-site VPN-соединения с Yandex Cloud с помощью Terraform.
- Client VPN между удаленными устройствами и Yandex Cloud. В качестве шлюза для удаленного доступа используйте ВМ с функцией client VPN на основе образа из Cloud Marketplace.
  См. инструкцию в разделе Создание VPN-соединения с помощью OpenVPN. Возможно так же использование сертифицированных СКЗИ.
- Приватное выделенное соединение между удаленной площадкой и Yandex Cloud с помощью услуги Cloud Interconnect.

Для доступа в инфраструктуру по управляющим протоколам (например, SSH, RDP) рекомендуется создать бастионную виртуальную машину. Для этого можно использовать бесплатное решение Teleport. Доступ к бастионной виртуальной машине или VPN-шлюзу из интернета должен быть ограничен.

Для дополнительного контроля действий администраторов рекомендуется использовать решения PAM (Privileged Access Management) с записью сессии администратора (например, Teleport). Для доступа по SSH и VPN рекомендуется отказаться от паролей и вместо этого использовать открытые ключи, X.509-сертификаты и SSH-сертификаты. При настройке SSH для виртуальных машин рекомендуется использовать SSH-сертификаты, в том числе и для хостовой части SSH.

Для доступа к веб-сервисам, развернутым в облаке, рекомендуется использовать TLS версий 1.2 и выше.

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в вашем браузере.
- Откройте все созданные сети.
- Перейдите в раздел **Таблицы маршрутизации**.
- Если найдены маршруты в приватные сети удаленных площадок, которые направлены через ВМ с VPN шлюзом, то рекомендация выполняется.
- Также проверьте виртуальные машины в каждом облаке на предмет наличия VPN шлюзов и проверьте назначенные им Security Groups на предмет открытых известных портов для VPN.

Ручная проверка:

Обратитесь к вашему менеджеру со стороны облака и уточните подключена ли у вас услуга Cloud Interconnect. Если подключена, проанализируйте выполняется ли удаленный доступ.

2.7 Исходящий доступ в интернет контролируется

Возможные варианты организации исходящего доступа в интернет:

- Публичный IP-адрес. Адрес назначается ВМ по принципу one-to-one NAT.
- Egress NAT (NAT-шлюз). Включает доступ в интернет для подсети через общий пул публичных адресов Yandex Cloud. Не рекомендуется использовать Egress NAT для критичных взаимодействий, так как IP-адрес NAT-шлюза может использоваться несколькими клиентами одновременно. Следует учитывать эту особенность при моделировании угроз для инфраструктуры. Подробнее про настройку.
- NAT-инстанс. Функцию NAT выполняет отдельная ВМ. Для создания такой ВМ можно использовать образ NAT-инстанс из Cloud Marketplace.

Сравнение способов доступа в интернет:

Публичный IP-адрес	Egress NAT	NAT-инстанс
Плюсы:	**Плюсы:**	**Плюсы:**
- Не требует настройки- Выделенный адрес для каждой ВМ	- Не требует настройки - Работает только на исходящих соединениях	- Возможность фильтровать трафик на NAT-инстансе - Возможность использовать собственный файрвол - Экономия IP-адресов
Минусы:	**Минусы:**	**Минусы:**
- Выставлять ВМ напрямую в интернет может быть небезопасно - Стоимость резервирования каждого адреса	- Общий пул IP-адресов - Функция на стадии Preview, поэтому не рекомендуется для продуктовых сред	- Требуется настройка - Стоимость использования ВМ (vCPU, RAM, диска)

Вне зависимости от выбранного варианта организации исходящего доступа в интернет, ограничивайте трафик с помощью одного из механизмов, описанных выше. Для построения защищённой системы необходимо использовать статические IP-адреса, так как их можно внести в список исключений файрвола принимающей стороны.

Проверка в консоли управления:

- Откройте консоль Yandex Cloud в вашем браузере.
- Перейдите в нужный каталог.
- Перейдите в раздел **IP-адреса**.
- Если у всех публичных адресов в столбце **Защита от DDoS-атак** установлено значение **Включена**, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска всех ВМ с публичными адресами:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do echo "VM_ID: " && yc compute instance list --folder-id=$FOLDER_ID -
-format=json | jq -r '.[] |
select(.network_interfaces[].primary_v4_address.one_to_one_nat.address
)' | jq -r '.id' \
&& echo "FOLDER_ID: " $FOLDER_ID && echo "-----"
done;
done
```

- Если VM_ID напротив FOLDER_ID указано пустое значение, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению»
- Выполните команду для поиска наличия Egress NAT (NAT-шлюз):

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do echo "NAT_GW: " && yc vpc gateway list --folder-id=$FOLDER_ID --
format=json | jq -r '.[] | select(.id)' | jq -r '.id'  && echo
"FOLDER_ID: " $FOLDER_ID && echo "-----"
done;
done
```

- Если NAT_GW напротив FOLDER_ID указано пустое значение, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению»
- Выполните команду для поиска наличия NAT-инстанса:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for DISK_ID in $(yc compute disk list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id'); do yc compute disk get --id=$DISK_ID --
format=json | jq -r '. |
select(.product_ids[0]=="fd8v7ru46kt3s4o5f0uo")' | jq -r '.id'
done;
done;
done
```

- Если результатом является пустая строка, то рекомендация выполняется. Если видите id NAT-инстанса, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

- В случае наличия публичных адресов на ВМ убедитесь, что они необходимы. В противном случае удалите внешний IP-адрес в настройках ВМ.
- В случае наличия NAT-Gateway убедитесь, что он необходим. В противном случае удалите его.
- В случае наличия NAT-инстанс убедитесь, что он необходим. В противном случае удалите его.

2.8 Запросы DNS не передаются в сторонние рекурсивные резолверы

Для повышения отказоустойчивости часть трафика может передаваться в сторонние рекурсивные резолверы. Если необходимо избежать этого, обратитесь в службу технической поддержки.

3. Безопасная конфигурация виртуальной среды

Введение

В данном разделе представлены рекомендации клиентам по настройкам безопасности в облачных сервисах Yandex Cloud, а также использованию дополнительных средств защиты данных в виртуальной среде.

Общее

3.1 Использование серийной консоли контролируется либо отсутствует

На виртуальных машинах доступ к серийной консоли по умолчанию отключен. Риски использования серийной консоли перечислены в разделе Начало работы с серийной консолью документации Yandex Compute Cloud.

При работе с серийной консолью:

- Убедитесь, что критичные данные не попадают в вывод серийной консоли.
- При включенном доступе к серийной консоли по SSH убедитесь, что работа с учётными данными и пароль для локального входа в операционную систему соответствуют стандартам регуляторов. Например, в инфраструктуре для хранения данных платежных карт пароль должен соответствовать требованиям стандарта PCI DSS: содержать как буквы, так и цифры, иметь длину не менее 7 символов и меняться не реже чем каждые 90 дней.

> Согласно стандарту PCI DSS, доступ к виртуальной машине через серийную консоль считается «неконсольным» (non-console), и Yandex Cloud применяет для него шифрование TLS.

Не рекомендуется использовать доступ к серийной консоли без крайней необходимости.

Проверка в консоли управления:

- В консоли управления выберите каталог, ВМ которого вы хотите проверить.
- В списке сервисов выберите **Compute Cloud**.
- Откройте настройки всех необходимых ВМ.
- В блоке **Доступ** найдите параметр **Дополнительно**.
- Опция **Доступ** к серийной консоли должна быть отключена.
- Если у всех ВМ опция отключена, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Найдите ВМ с включённым доступом к серийной консоли:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for VM_ID in $(yc compute instance list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id'); do echo "VM_ID: " && yc compute
instance get --id=fhm43b4b05132dasp5sd --full --format=json | jq -r '.
| select(.metadata."serial-port-enable"=="1")' | jq -r '.id' && echo
"FOLDER_ID: " $FOLDER_ID && echo "-----"
done;
done;
done
```

- Если VM_ID напротив FOLDER_ID указано пустое значение, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Если серийная консоль не должна быть использована на ВМ, отключите её.

3.2 Используется эталонный образ для развёртывания ВМ

При развёртывании виртуальных машин рекомендуется:

- Подготовить образ виртуальной машины, настройки системы в котором соответствуют вашей политике информационной безопасности. Создать образ можно с помощью Packer, см. раздел Начало работы с Packer.

- Использовать этот образ для создания виртуальной машины или группы виртуальных машин.
- В информации о виртуальной машине убедиться, что для создания диска использовался именно этот образ.

Проверка в консоли управления:

- В консоли управления выберите каталог, ВМ которого хотите проверить.
- В списке сервисов выберите **Compute Cloud**.
- Перейдите на вкладку **Диски**.
- Откройте настройки всех дисков.
- В блоке **Источник** найдите параметр **Идентификатор**.
- Если во всех дисках отображается ID вашего эталонного образа, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска дисков ВМ, которые не имеют ID вашего эталонного образа:

```
export ORG_ID=export ORG_ID=bpf4c0lctf2t734l95ui
export IMAGE_ID=<id вашего эталонного образа>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for DISK_ID in $(yc compute disk list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id'); \
do echo "DISK_ID: " && yc compute disk get --id=$DISK_ID \
--format=json | jq -r --arg IMAGE_ID $IMAGE_ID '. |
select(."source_image_id"==$IMAGE_ID | not)' | jq -r '.id' && echo
"FOLDER_ID: " $FOLDER_ID && echo "-----"
done;
done;
done
```

- Если DISK_ID напротив FOLDER_ID указано пустое значение, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

- Выясните, почему для данных дисков ВМ используется не эталонный образ.
- Пересоздайте ВМ с необходимым образом.

3.3 Инструмент Terraform используется в соответствии с лучшими практиками ИБ

Terraform позволяет управлять облачной инфраструктурой с помощью файлов конфигураций. При изменении файлов Terraform автоматически определяет, какая часть вашей конфигурации уже развернута, что следует добавить или удалить. Подробнее в разделе Начало работы с Terraform.

В файлах конфигураций Terraform не рекомендуется указывать приватную информацию: пароли, секреты, персональные данные, данные платежных систем и др. Вместо этого необходимо использовать сервисы для хранения и использования в конфигурации секретов, например: HashiCorp Vault из Cloud Marketplace или Lockbox (для передачи секретов в целевой объект без использования Terraform).

Если всё же требуется указать приватную информацию в конфигурации, необходимо принять меры безопасности:

- Указывать для приватной информации параметр sensitive = true, чтобы отключить её вывод в консоль при выполнении команд terraform plan, terraform apply.
- Использовать terraform remote state. Рекомендуется загружать состояние Terraform в Object Storage, а также настроить блокировку конфигурации с помощью Managed Service for YDB для предотвращения одновременного редактирования администраторами.
- Использовать механизм передачи секретов в Terraform через env вместо plain text либо использовать встроенную возможность Key Management Service по шифрованию данных в Terraform с помощью отдельного файла с приватными данными. Подробнее о данной технике.

Об обеспечении безопасности Object Storage читайте ниже в подразделе Object Storage.

> После развертывания конфигурации файл конфигурации с приватными данными можно удалить.

Проверяйте ваши Terraform-манифесты с помощью Checkov с поддержкой Yandex Cloud.

- Пример: сканирование tf-файлов с помощью Checkov.
- Пример: хранение состояния Terraform в Object Storage.

Ручная проверка:

Проведите точечный сбор данных об использовании лучших практик по безопасности Terraform.

3.4 Выполняется контроль целостности файлов

Множество стандартов по ИБ требуют выполнения контроля целостности критичных файлов. Для этого можно использовать бесплатные host-based решения:

- Wazuh
- Osquery

В маркетплейсе облака также доступны платные решения — например, Kaspersky Security.

Ручная проверка:

Проведите точечный сбор данных об использовании контроля целостности.

3.5 Учтены принципы защиты от атак по побочным каналам (side-chanel)

Для наилучшей защиты от атак по побочным каналам процессора (так называемым атакам side-channel на уровне CPU, например, Spectre или Meltdown) необходимо:

- Использовать полноядерные виртуальные машины, то есть виртуальные машины с долей ядра CPU в 100 процентов.
- Использовать виртуальные машины с чётным числом ядер (2 ядра, 4 ядра и т. д.).
- Устанавливать обновления операционной системы и ядра, которые обеспечивают защиту от атак с использованием побочных каналов (например, Kernel page-table isolation для Linux, приложения, собранные с Retpoline).

Для размещения нагрузок, наиболее критичных с точки зрения безопасности, рекомендуется использовать выделенные хосты (dedicated hosts).

Подробнее о защите от side-channel-атак в облачных окружениях.

3.6 Отсутствует публичный доступ к бакету Object Storage

Рекомендуется назначать минимальные роли на бакет с помощью IAM и дополнять или детализировать их с помощью BucketPolicy (например, для ограничения доступа к бакету по IP-адресам, выдачи гранулярных прав на объекты и т.д.)

Проверка доступа к ресурсам Object Storage происходит на трёх уровнях:

- проверки сервиса IAM;
- политики доступа (bucket policy);
- списки управления доступом (ACL).

Порядок проверки:

1. Если запрос прошел проверку IAM, к нему применяется проверка политики доступа.
2. Проверка правил политики доступа происходит в следующем порядке:
    2.1 Если запрос подошел хотя бы под одно из правил Deny, то доступ будет запрещён.
    2.2 Если запрос подошел хотя бы под одно из правил Allow, то доступ будет разрешён.
    2.3 Если запрос не подошел ни под одно из правил, то доступ будет запрещён.
3. Если запрос не прошёл проверку IAM или политики доступа, то применяется проверка доступа через ACL объекта.

В сервисе IAM бакет наследует такие же права доступа, как у каталога и облака, в котором он находится. Подробнее об этом в разделе Наследование прав доступа на бакет системными группами Yandex Cloud. Поэтому рекомендуется выдавать только минимально необходимые роли на определенные бакеты или объекты сервиса Object Storage.

Политики доступа используются для дополнительной защиты данных, например, для ограничения доступа к бакету по IP-адресам, выдачи гранулярных прав на объекты и т. д.

ACL позволяет предоставить доступ к объекту в обход проверок IAM и политик доступа. Рекомендуем установить строгие ACL на бакеты.

Пример безопасной конфигурации Object Storage: Terraform

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, бакеты которого вы хотите проверить.
- В списке сервисов выберите **Object Storage**.
- Нажмите на три точки напротив каждого бакета и проверьте ACL бакета на наличие allUsers и allAuthenticatedUsers.
- Зайдите внутрь бакета и проверьте ACL на каждый объект бакета на наличие allUsers и allAuthenticatedUsers.
- Проверьте, что в разделе **Доступ на чтение объектов** включен параметр **Публичный**. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Настройте aws cli на работу с облаком.
- Выполните команду для ACL бакета на наличие allUsers, allAuthenticatedUsers:

```
aws --endpoint-url=https://storage.yandexcloud.net s3api get-bucket-acl  <имя_бакета>
```

Инструкции и решения по выполнению:

Если публичный доступ включён, удалите его либо контролируйте (осознанно выдавайте для публичных данных).

3.7 В Object Storage используются политики доступа (Bucket Policy)

Политики доступа устанавливают права на действия с бакетами, объектами и группами объектов. Политика срабатывает, когда пользователь делает запрос к какому-либо ресурсу. В результате срабатывания политики запрос либо выполняется, либо отклоняется.

Примеры Bucket Policy:

- Политика, которая разрешает скачивать объекты только из указанного диапазона IP-адресов.
- Политика, которая запрещает скачивать объекты с указанного IP-адреса.
- Политика дает разным пользователям полный доступ только к определенным папкам, каждому пользователю — к своей.

- Политика дает каждому пользователю и сервисному аккаунту полный доступ к папке с названием, равным идентификатору пользователя или сервисного аккаунта.

Рекомендуется убедиться, что в вашем бакете Object Storage используется как минимум одна политика.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, политики доступа которых вы хотите проверить.
- В списке сервисов выберите **Object Storage**.
- Перейдите в раздел **Политика доступа**.
- Убедитесь, что как минимум одна политика включена. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Настройте aws cli на работу с облаком.
- Выполните команду для ACL бакета на проверку наличия allUsers, allAuthenticatedUsers:

```
aws --endpoint-url=https://storage.yandexcloud.net s3api get-bucket-policy --bucket <имя_бакета>
```

Инструкции и решения по выполнению:

Включите необходимую политику.

3.8 В Object Storage включена функция «Блокировка версии объекта» (object lock)

При обработке в бакетах критичных данных необходимо обеспечить их защиту от удаления и резервирование версий. Это возможно сделать с помощью механизмов версионирования и управления жизненным циклом и блокировки версии объекта.

Версионирование бакета — это возможность хранить историю версий объекта. Каждая версия является полной копией объекта и занимает соответствующий объём в Object Storage. С помощью управления версиями вы можете защитить ваши данные как от непреднамеренных действий пользователя, так и от сбоев приложений.

В случае удаления или модификации объекта с включённым версионированием на самом деле создается новая версия объекта

с новым id. В случае удаления объект становится недоступен для чтения, но его версия хранится и подлежит восстановлению.

Настройка версионирования описана в статье Версионирование бакета документации Object Storage.

Настройка жизненного цикла описана в статьях Жизненные циклы объектов в бакете и Конфигурация жизненных циклов объектов в бакете документации Object Storage.

Также для защиты версий объекта от удаления необходимо использовать object lock. Подробнее про типы блокировок и как их включить читайте в документации.

Срок хранения критичных данных в бакете определяется требованиями ИБ компании клиента и требованиями стандартов ИБ. Например, стандарт PCI DSS устанавливает, что аудитные логи должны храниться не менее одного года, и как минимум три месяца должны быть доступны онлайн.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, бакеты которых вы хотите проверить.
- В списке сервисов выберите **Object Storage**.
- Откройте настройки всех бакетов.
- Перейдите во вкладку **Версионирование** и убедитесь, что оно включено. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Настройте aws cli на работу с облаком.
- Выполните команду, чтобы проверить, что версионирование включено:

```
aws --endpoint https://storage.yandexcloud.net \
s3api get-bucket-versioning \
--bucket <имя вашего бакета>
```

- Выполните команду, чтобы проверить, что версионирование включено:

```
aws --endpoint-url=https://storage.yandexcloud.net/ \
s3api get-object-lock-configuration \
--bucket <имя вашего бакета>
```

Инструкции и решения по выполнению:

Если публичный доступ включён, удалите или контролируйте его (включая только по необходимости и согласованию).

3.9 В Object Storage включён механизм логирования действий с бакетом

При использовании сервиса Object Storage для хранения критичных данных необходимо включать логирование действий с бакетами. Подробнее в статье Механизм логирования действий с бакетом документации Object Storage.

При этом будут записываться именно логи data-plane с объектами: PUT, DELETE, GET, POST, OPTIONS, HEAD.

Аналогично можно запросить запись данных логов в Audit Trails кроме чтения. В будущем все эти логи будут записываться в Audit Trails.

Дополнительно возможен анализ логов Object Storage при помощи DataLens. Подробнее в статье Анализ логов Object Storage при помощи DataLens.

Инструкции и решения по выполнению:

Проверить включён ли механизм логирования можно только через Terraform/API согласно инструкции.

3.10 В Object Storage настроено управление кросс-доменными запросами (CORS)

При необходимости кросс-доменных запросов к объектам в бакетах клиенту необходимо настроить политику Cross-origin resource sharing (CORS) в соответствии с требованиями ИБ компании клиента. Подробнее в разделе CORS-конфигурация бакетов документации Object Storage.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, бакеты которых вы хотите проверить.
- В списке сервисов выберите **Object Storage**.
- Откройте настройки всех бакетов.
- Перейдите во вкладку **CORS** — конфигурация должна быть настроена. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Настройте CORS.

3.11 На управляемых базах данных назначена Группа безопасности

Рекомендуется запретить доступ из интернета к базам данных, которые содержат критичные данные, в частности данные PCI DSS или персональные данные. Настройте группы безопасности, чтобы разрешить подключение к СУБД только с определенных IP-адресов, см. инструкцию в разделе Создать группу безопасности. Указать группу безопасности можно в настройках кластера или при его создании, в блоке сетевых настроек.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых вы хотите проверить базы данных.
- В списке сервисов выберите сервис(ы), где находятся управляемые базы данных.
- В настройках объектов найдите параметр **Группа безопасности** и убедитесь, что назначена хотя бы одна группа безопасности.
- Если в параметрах каждого объекта указана хотя бы одна группа безопасности, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Выполните команду для поиска managed postgres без SG:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for DB_ID in $(yc managed-postgresql cluster list --folder-
id=$FOLDER_ID --format=json | jq -r '.[].id'); do yc managed-
postgresql cluster get --id=$DB_ID --format=json | jq -r '. |
select(.security_group_ids | not)' | jq -r '.id'
done;
done;
done
```

- Если выдается пустая строка, то рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка наличия SG на управляемых базах данных:

- Выполните команду для поиска managed SQL без SG:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for DB_ID in $(yc managed-mysql cluster list --folder-id=$FOLDER_ID
--format=json | jq -r '.[].id'); do yc managed-mysql cluster get --
id=$DB_ID --format=json | jq -r '. | select(.security_group_ids |
not)' | jq -r '.id'
done;
done;
done
```

- Если выдается пустая строка, то рекомендация выполняется. Если нет, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Если найдены базы данных без групп безопасности, назначьте их либо включите функционал **Группа безопасности по умолчанию**.

3.12 На управляемых базах данных не назначен публичный IP-адрес

Назначение публичного IP-адреса на управляемую базу данных повышает риски ИБ. Рекомендуется не назначать внешний IP-адрес без крайней необходимости.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых вы хотите проверить базы данных.
- В списке сервисов выберите сервис(ы), где находятся управляемые базы данных.
- В настройках объектов перейдите во вкладку **Хосты**.
- Если в параметрах каждого объекта отключена опция **Публичный доступ**, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска кластеров управляемых БД с публичным адресом:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for DB_ID in $(yc managed-mysql cluster list --folder-id=$FOLDER_ID
--format=json | jq -r '.[].id'); do yc managed-mysql hosts list --
cluster-id=$DB_ID --format=json | jq -r '.[] |
select(.assign_public_ip)' | jq -r '.cluster_id'
done;
done;
done
```

- Если выдается пустая строка, рекомендация выполняется. Если нет, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Удалите публичный доступ, если он не требуется.

3.13 Включена настройка защиты от удаления (deletetion protection)

В управляемых базах данных в Yandex Cloud существует возможность включения функции защиты от удаления. Защита от удаления управляет защитой кластера от непреднамеренного удаления пользователем. Включённая защита не помешает подключиться к кластеру вручную и удалить данные.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых вы хотите проверить базы данных.
- В списке сервисов выберите сервис(ы), где находятся управляемые базы данных.
- В настройках объектов перейдите во вкладку **Дополнительные настройки**.
- Если в параметрах каждого объекта включена опция **Защита от удаления**, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска кластеров управляемых БД без включённой защиты от удаления:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for DB_ID in $(yc managed-mysql cluster list --folder-id=$FOLDER_ID
--format=json | jq -r '.[].id'); do yc managed-mysql cluster get --
id=$DB_ID --format=json | jq -r '. | select(.deletion_protection |
not)' | jq -r '.id'
done;
done;
done
```

- Если выдается пустая строка, то рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

- В консоли управления выберите облако или каталог, в которых хотите включить защиту от удаления.
- В списке сервисов выберите сервис(ы), где находятся управляемые базы данных.
- В настройках объектов перейдите во вкладку **Дополнительные настройки**.
- В параметрах объекта включите опцию **Защита от удаления**.

3.14 Выключена настройка доступа из DataLens без необходимости

Не следует без необходимости включать доступ к базам данных с критичными данными из консоли управления, DataLens и других сервисов. Доступ из DataLens может потребоваться для анализа и визуализации данных. Эти доступы осуществляются через служебную сеть Yandex Cloud, с аутентификацией и использованием шифрования TLS. Включить и отключить доступы из DataLens или других сервисов можно в настройках кластера или при его создании, в блоке дополнительных настроек.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых вы хотите проверить базы данных.
- В списке сервисов выберите сервис(ы), где находятся управляемые базы данных.
- В настройках объектов перейдите во вкладку **Дополнительные настройки**.
- Если в параметрах каждого объекта отключена опция **Доступ из DataLens**, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Найдите кластеры управляемых БД с включённым доступом из DataLens:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for DB_ID in $(yc managed-mysql cluster list --folder-id=$FOLDER_ID
--format=json | jq -r '.[].id'); do yc managed-mysql cluster get --
id=$DB_ID --format=json | jq -r '. | select(.config.access.data_lens)'
| jq -r '.id'
done;
done;
done
```

- Если выдается пустая строка, то рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

- В консоли управления выберите облако или каталог, в которых вы хотите выключить доступ из DataLens.
- В списке сервисов выберите сервис(ы), где находятся управляемые базы данных.
- В настройках объектов перейдите во вкладку **Дополнительные настройки**.
- В параметрах объекта отключите опцию **Доступ из DataLens.**

3.15 На управляемых БД выключен доступ из консоли управления

Доступ к БД из консоли управления может потребоваться для отправки SQL- запросов в БД и визуализации структуры данных.

Рекомендуется включать такой доступ только в случае необходимости, т.к. он увеличивает риски ИБ. В штатном режиме используйте стандартное подключение к БД под пользователем БД.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых вы хотите проверить базы данных.
- В списке сервисов выберите сервис(ы), где находятся управляемые БД.
- В настройках объектов перейдите во вкладку **Дополнительные настройки**.
- Если в параметрах каждого объекта отключена опция **Доступ из консоли управления**, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска кластеров управляемых БД с включённым доступом из консоли управления:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for DB_ID in $(yc managed-mysql cluster list --folder-id=$FOLDER_ID
--format=json | jq -r '.[].id'); do yc managed-mysql cluster get --
id=$DB_ID --format=json | jq -r '. | select(.config.access.web_sql)' |
jq -r '.id'
done;
done;
done
```

- Если выдается пустая строка, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

- В консоли управления выберите облако или каталог, в которых вы хотите выключить доступ из консоли.
- В списке сервисов выберите сервис(ы), где находятся управляемые базы данных.
- В настройках объектов перейдите во вкладку **Дополнительные настройки**.
- В параметрах объекта выключите опцию **Доступ из консоли**.

Cloud Functions и Yandex API Gateway

3.16 Публичные облачные функции применяются только в исключительных случаях

Во всех случаях, когда явно не требуется использование публичных функций, рекомендуется использовать приватные функции. Подробнее о настройке доступа к функциям см. в разделе Управление правами доступа к функции. Рекомендуется использовать приватные функции и назначать права на вызов функции конкретным пользователям облака.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых вы хотите проверить функции.
- В списке сервисов выберите **Cloud Functions**.
- Откройте все функции.
- В настройках функций перейдите во вкладку **Обзор**.
- Если в параметрах каждого объекта опция **Публичная функция** отключена, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка YC CLI:

- Выполните команду для поиска прав доступа allUsers, на уровне Cloud Functions во всех каталогах:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for FUN in $(yc serverless function list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id'); \
```

```
do yc serverless function  list-access-bindings --id $FUN --
format=json | jq -r '.[] | select(.subject.id=="allAuthenticatedUsers"
or .subject.id=="allUsers")' && echo $FUN
done;
done;
done
```

- Если в каждом указанном ресурсе отсутствуют субъекты allUsers и allAuthenticatedUsers, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Отключите публичный доступ.

3.17 Учтены атаки по побочным каналам в Cloud Functions

Хосты и гипервизоры, на которых выполняются Cloud Functions, содержат все необходимые обновления для защиты от атак по побочным каналам процессора (side-channel attacks). Однако следует иметь в виду, что функции различных клиентов не изолированы по ядрам и формально существует поверхность атаки со стороны функции одного пользователя на функцию другого пользователя. Специалисты по ИБ Yandex Cloud считают сценарий атаки по побочным каналам в контексте функций маловероятным, однако следует учитывать данный риск, в частности, при построении модели угроз и анализа рисков для инфраструктуры PCI DSS.

Ручная проверка:

Убедитесь, что наиболее критичные системы не используют Cloud Functions либо это учтено в модели угроз.

3.18 Учтены особенности синхронизации времени в Cloud Functions

Сервис Cloud Functions не гарантирует **синхронизацию** времени перед выполнением или в процессе выполнения запросов функциями. Чтобы получить лог выполнения функции с точными метками времени на стороне Cloud Functions, следует выводить лог в stdout. Также клиент может самостоятельно принимать логи выполнения функции и помечать их меткой времени на принимающей стороне, взятой из источника времени, синхронизированного с Yandex Cloud. Подробнее о синхронизации времени см. в разделе Настройка синхронизации часов с помощью NTP документации Compute Cloud.

3.19 Учтены особенности управления заголовками в Cloud Functions

Если функция вызывается для обработки HTTP-запроса, то возвращаемый результат должен представлять собой JSON-документ, содержащий код ответа HTTP, заголовки ответа и содержимое ответа. Cloud Functions автоматически обработает этот JSON, и пользователь получит данные в виде стандартного HTTP-ответа. Клиенту необходимо самостоятельно управлять заголовками ответа в соответствии с требованиями регуляторов и модели угроз. Инструкцию по обработке HTTP-запроса см. в статье Вызов функции в Cloud Functions документации Cloud Functions.

3.20 Serverless Containers/Cloud Functions использует внутреннюю сеть VPC

По умолчанию функция запускается в изолированной IPv4-сети с включённым NAT-шлюзом. Поэтому из функции доступны только публичные IPv4-адреса.

Если необходимо, в настройках функции можно указать облачную сеть. Тогда функция будет:

- Исполняться в указанной облачной сети.
- Иметь доступ не только в интернет, но и к пользовательским ресурсам, которые находятся в указанной сети, например, базам данных, виртуальным машинам и т.п.
- Иметь IP-адрес в диапазоне 198.19.0.0/16 при доступе к пользовательским ресурсам.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых вы хотите проверить функции.
- В списке сервисов выберите **Cloud Functions**.
- Откройте все функции.
- В настройках объектов перейдите во вкладку **Редактирование версии функции**.
- Если в параметрах каждого объекта значение опции **Сеть — VPC**, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка YC CLI:

- Выполните команду для поиска всех облачных функций, для которых не заданы настройки сети в VPC:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for VER in $(yc serverless function version list --folder-
id=$FOLDER_ID --format=json | jq -r '.[].id'); \
do yc serverless function version get $VER --format=json | jq -r '. |
select(.connectivity.network_id | not)' | jq -r '.id'
done;
done;
done
```

- Если выдается пустая строка, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

- Выберите облако или каталог, в которых хотите проверить функции, в консоли управления.
- Выберите **Cloud Functions** в списке сервисов.
- Откройте функцию.
- Перейдите во вкладку **Редактирование версии функции** в настройках объектов.
- Установите значение опции **Сеть — VPC.**

Managed Service for YDB

3.21 Учтены рекомендации по работе с конфиденциальными данными в YDB

Запрещается в качестве названий базы данных, таблиц, столбцов, директорий и т.д. использовать конфиденциальные данные. Запрещается отправлять критичные данные (например данные платежных карт) в Managed Service for YDB (как Dedicated, так и Serverless) в открытом виде.
Перед отправкой данных их необходимо шифровать на уровне приложения, для чего можно воспользоваться сервисом KMS или любым другим способом, соответствующим стандарту регуляторов. Если срок хранения данных известен заранее, рекомендуется настроить функцию Time To Live.

3.22 Учтены рекомендации по защите от sql- инъекций YDB

При работе с базой данных для защиты от SQL-инъекций необходимо использовать параметризованные подготовленные запросы.
Если в приложении используется динамическая генерация шаблонов запросов, необходимо следить, чтобы недоверенный пользовательский ввод не попадал в шаблон запроса.

3.23 Публичный доступ отсутствует для YDB

При работе с базой данных в режиме Dedicated рекомендуется использовать её внутри VPC и не открывать к ней доступ из интернета. В режиме Serverless база данных является доступной из интернета, что необходимо учитывать, в частности, при моделировании угроз при построении инфраструктуры. Подробнее о режимах работы см. в разделе Режимы работы Serverless и Dedicated документации Managed Service for YDB.

При настройке доступа к БД следует использовать принцип минимальных привилегий.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых вы хотите проверить базу данных.
- В списке сервисов выберите **Managed Service for YDB**.
- Откройте все базы данных.
- В настройках базы данных перейдите во вкладку **Сеть**.
- Если в параметрах каждого объекта отключена опция **Публичные IP- адреса**, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска кластеров управляемых БД с публичным адресом:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
```

```
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for DB_ID in $(yc ydb database list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id'); do yc ydb database get --id=$DB_ID --
format=json | jq -r '. | select(.assign_public_ips)' | jq -r '.id'
done;
done;
done
```

- Если выдается пустая строка, то рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Удалите публичный доступ, если он не требуется.

3.24 Учтены рекомендации по резервному копированию YDB

При использовании механизма создания резервных копий по требованию, необходимо убедиться, что данные резервной копии должным образом защищены.

При самостоятельном создании резервных копий в сервисе Object Storage необходимо следовать рекомендациям подраздела Object Storage выше — например, использовать встроенные возможности шифрования бакетов.

Yandex Container Registry

3.25 Настроен ACL по IP адресам для Yandex Container Registry

Доступ к вашему Container Registry рекомендуется ограничить до конкретных IP-адресов.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить реестр.
- В списке сервисов выберите **Container Registry**.
- В настройках конкретного реестра перейдите во вкладку **Доступ для IP- адресов**.

- Если в параметрах указаны конкретные адреса для доступа, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска CR без фильтров по IP:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for CR in $(yc container registry list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id'); do yc container registry list-ip-
permissions --id=$CR --format=json | jq -r '.[] | select(.ip)' | jq -r
'.action' && echo $CR "IF ACTION PULL/PUSH exist before CR then OK"
done;
done;
done
```

- Если перед каждым ID registry выдаётся PULL/PUSH, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Задайте конкретные адреса для доступа к реестрам.

Yandex Container Solution

Не рекомендуется использовать привилегированные контейнеры для запуска нагрузок, обрабатывающих недоверенный пользовательский ввод. Привилегированные контейнеры следует использовать для администрирования виртуальной машины или других контейнеров.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить ВМ.
- В списке сервисов выберите **Compute Cloud**.
- Зайдите в настройки конкретной ВМ с **Container Optimized Image**.
- В блоке **Настройки** Docker-контейнера найдите параметр **Привилегированный режим**.

Если параметр отключён, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска CR без фильтров по IP:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for VM_ID in $(yc compute instance list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id'); \
do yc compute instance get --id=$VM_ID --full --format=json | jq -r '.
| select(.metadata."docker-container-declaration")| .metadata."docker-
container-declaration" | match("privileged: true") | .string' && echo
$VM_ID
done;
done;
done
```

- Если перед каждым ID ВМ отсутствует «privileged: true», то рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

- В консоли управления выберите облако или каталог, в которых нужно проверить ВМ.
- В списке сервисов выберите **Compute Cloud**.
- Зайдите в настройки конкретной ВМ с **Container Optimized Image**.
- В блоке **Настройки** Docker-контейнера отключите параметр **Привилегированный режим**.

3.26 Срок действия сертификата Yandex Certificate Manager как минимум 30 дней

Сервис Yandex Certificate Manager позволяет управлять TLS-сертификатами для API-шлюзов сервиса API Gateway, а также для сайтов и бакетов в Object Storage. Сервис Application Load Balancer интегрирован с Certificate Manager для хранения и установки сертификатов. Рекомендуется использовать Certificate Manager для получения и автоматической ротации сертификатов.

При работе с TLS в приложении рекомендуется ограничивать список доверенных корневых сертификатов (root CA).

При использовании технологий certificate pinning следует учитывать, что сервис Let’s Encrypt выдает сертификаты со сроком действия в 90 дней.

Рекомендуется заблаговременно обновлять сертификат, если вы не используете автоматическое обновление.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить ВМ.
- В списке сервисов выберите **Yandex Certificate Manager**.
- Зайдите в настройки каждого сертификата и найдите параметр **Дата окончания**.
- Если в параметре указано, что сертификат проживет ещё как минимум 30 дней, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Найдите все сертификаты вашей организации с датой окончания:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for CERT_ID in $(yc certificate-manager certificate list --folder-
id=$FOLDER_ID --format=json | jq -r '.[].id'); \
do yc certificate-manager certificate get --id $CERT_ID --format=json
| jq -r '. | "Date of the end " + .not_after + " --- Cert_ID " + .id'
done;
done;
done
```

- Если перед каждым ID ВМ отсутствует «privileged: true», то рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Обновите сертификат либо настройте автоматическое обновление.

3.27 Рекомендации по настройке безопасности инстанса GitLab выполняются

Рекомендации представлены здесь.

Ручная проверка:

Необходимо проверить вручную.

4. Шифрование данных и управление ключами

Введение

Yandex Cloud предоставляет встроенные функции шифрования при использовании ряда сервисов. В зоне ответственности клиента находится включение шифрования в этих сервисах, а также самостоятельная реализация шифрования в других компонентах обработки критичных данных. Для шифрования данных и управления ключами шифрования предназначен сервис Key Management Service (KMS).

API сервисов Yandex Cloud поддерживают наборы алгоритмов (cipher suits) и версии протокола TLS, отвечающие требованиям стандарта PCI DSS и другим стандартам.

Шифрование в состоянии покоя (at rest)

По умолчанию все пользовательские данные в состоянии покоя (at rest) зашифрованы на уровне Yandex Cloud. Шифрование на уровне Yandex Cloud является реализации одной из лучших практик по защите данных пользователей и выполняется на ключах Yandex Cloud.

Если ваша корпоративная политика информационной безопасности предъявляет требования к длине ключа или частоте ротации ключей, вы можете шифровать данные собственными ключами. Для этого можно использовать сервис KMS и его интеграцию с другими сервисами Yandex Cloud, либо реализовать шифрование на data plane-уровне полностью самостоятельно.

Yandex Cloud предоставляет функции шифрования в состоянии покоя (at rest) для следующих сервисов:

- Object Storage;
- Managed Service for Kubernetes.

4.1 В Yandex Object Storage включено шифрование данных at rest с ключом KMS

Для защиты критичных данных в Yandex Object Storage рекомендуется использовать шифрование бакета на стороне сервера с помощью ключей

Yandex Key Management Service (server-side encryption). Такое шифрование защищает от случайной или намеренной публикации содержимого бакета в интернете. Подробнее см. в разделе Шифрование документации Object Storage.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить бакеты.
- В списке сервисов выберите **Object Storage**.
- Перейдите в настройки бакета.
- Перейдите на вкладку **Шифрование**.
- Убедитесь, что шифрование включено и указан KMS ключ шифрования.
- Если шифрование включено, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Настройте aws cli на работу с облаком.
- Выполните команду, чтобы проверить, что шифрование включено:

```
aws --endpoint-url=https://storage.yandexcloud.net/ \
s3api get-bucket-encryption \
--bucket action-log-k8s-audit-log-bucket
```

- Если шифрование включено, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Настройте шифрование бакета согласно инструкции.

Шифрование в состоянии передачи (in transit)

В большинстве случаев соединение с сервисами Yandex Cloud возможно только с использованием HTTPS. Однако в некоторых сценариях data plane доступ к сервисам может быть осуществлен и по HTTP, без шифрования соединения на прикладном уровне. Во всех таких сценариях у пользователя есть возможность выбрать в настройках сервиса, какой протокол использовать при data plane-операциях: HTTP или HTTPS, а в случае выбора HTTPS указать собственный TLS-сертификат.

Убедитесь, что используете для работы (или соединения) с API сервисов Yandex Cloud протокол TLS версии 1.2 и выше, так как более ранние версии подвержены уязвимостям.

Например, использование gRPC - интерфейсов Yandex Cloud гарантирует работу по TLS 1.2 и выше, так как протокол HTTP/2, на основе которого работает gRPC, устанавливает TLS 1.2 в качестве минимальной поддерживаемой версии протокола TLS.

Поддержка устаревших протоколов TLS в сервисах Yandex Cloud будет постепенно прекращена.

Yandex Cloud предоставляет возможность использования собственных TLS - сертификатов для следующих сервисов:

- Object Storage
- Application Load Balancer
- Virtual Private Cloud (VPC)
- API Gateway
- Cloud CDN

4.2 В Yandex Object Storage включено HTTPS для хостинга статического сайта

Object Storage поддерживает безопасное подключение по протоколу HTTPS. Вы можете загрузить собственный сертификат безопасности, если к сайту в Object Storage требуется доступ по протоколу HTTPS. Также доступна интеграция с сервисом Certificate Manager. См. инструкции в документации Object Storage:

- Настройка HTTPS
- Бакет

При работе с сервисом Object Storage необходимо убедиться, что в клиенте отключена поддержка протоколов TLS ниже версии 1.2. При помощи политики (bucket policy) aws:securetransport необходимо проверить, что для бакета настроен запрет на работу без протокола TLS.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить бакеты.
- В списке сервисов выберите **Object Storage**.
- Перейдите в настройки бакета.
- Перейдите во вкладку **HTTPS**.
- Убедитесь, что доступ по протоколу включён и указан сертификат.
- Если доступ по HTTPS включён, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Включите доступ по HTTPS, если бакет используется для хостинга статического сайта.

4.3 В Yandex Application Load Balancer используется HTTPS

Сервис Application Load Balancer поддерживает HTTPS-обработчик с загрузкой сертификата из Certificate Manager. См. описание настройки обработчика в документации Application Load Balancer.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить балансировщики.
- В списке сервисов выберите **Application Load Balancer**.
- Перейдите в настройки балансировщика.
- Убедитесь, что у обработчика указан протокол **HTTPS**.
- Если указан HTTPS, рекомендация выполняется. Если нет, перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для вывода списка всех балансировщиков без https:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
```

```
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for ALB in $(yc application-load-balancer load-balancer list --
folder-id=$FOLDER_ID --format=json | jq -r '.[].id'); \
do yc application-load-balancer load-balancer get --id $ALB --format
json | jq -r '. | select(.listeners[0].tls | not)' | jq -r '.id'
done;
done;
done
```

- Если выведен пустой список, рекомендация выполняется.
  В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Включите HTTPS обработчик согласно инструкции.

4.4 В Yandex API Gateway используется HTTPS и собственный домен

API Gateway обеспечивает безопасное подключение по протоколу HTTPS. Вы можете привязать собственный домен и загрузить собственный сертификат безопасности для доступа к вашему API-шлюзу по протоколу HTTPS.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить шлюзы.
- В списке сервисов выберите **API Gateway** ⟶ **Настройки шлюза** ⟶ **Домены**.
- Убедитесь, что домен и сертификат подключены.
- Если домен и сертификат активны, рекомендация выполняется.
  В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для вывода списка всех api gateway без подключённых доменов и сертификатов:

```
export ORG_ID=yc.organization-manager.yandex
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
```

```
do for APIGW in $(yc serverless api-gateway list --folder-
id=$FOLDER_ID --format=json | jq -r '.[].id'); \
do yc serverless api-gateway get --id $APIGW --format json | jq -r '.
| select(.attached_domains[0].certificate_id | not)' | jq -r '.id'
done;
done;
done
```

- Если выведен пустой список, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

- В консоли управления выберите облако или каталог, в которых необходимо подключить домены и сертификаты.
- В списке сервисов выберите API Gateway → Настройки шлюза → Домены.
- Подключите домены и сертификаты.

4.5 В Yandex Cloud CDN используется HTTPS и собственный ssl сертификат

Cloud CDN поддерживает безопасное подключение по протоколу HTTPS к источникам. Также вы можете загрузить собственный сертификат безопасности для доступа к вашему CDN-ресурсу по протоколу HTTPS.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить ресурсы.
- В списке сервисов выберите **Cloud CDN**.
- Перейдите в настройки ресурса, в вкладку **Дополнительно**.
- Убедитесь, что в поле **Протокол для источников** указан протокол **HTTPS**.
- Убедитесь, что в поле **Сертификат** указан собственный сертификат либо **Let’s encrypt**.
- Если указан HTTPS и собственный сертификат, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для вывода списка всех ресурсов без подключённых сертификатов и HTTPS до источников:

```
export ORG_ID=bpf4c0lctf2t734195ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for CDN in $(yc cdn resource list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id'); \
do yc cdn resource get --id $CDN --format json | jq -r '. |
select(.origin_protocol=="HTTPS" and .ssl_certificate.type=="CM" |
not)' | jq -r '.id'
done;
done;
done
```

- Если выведен пустой список, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Подключите сертификат и HTTPS согласно инструкции.

Самостоятельное шифрование

При использовании сервисов, которые не имеют встроенных функций шифрования, шифрование критичных данных является ответственностью клиента.

4.6 Для критичных ВМ настроено шифрование диска с помощью KMS

Если вам необходимо шифрование диска, разместите файлы приложения на дополнительном (не загрузочном) диске виртуальной машины и настройте для этого диска полное шифрование (full disk encryption).

Решение: Шифрование диска ВМ с помощью KMS

Ручная проверка:

Необходимо вручную проверить, используется ли данное решение для критичных ВМ.

4.7 Для критичных данных используется шифрование с помощью KMS

Если шифрование данных необходимо, следует шифровать их на уровне приложения перед записью в базы данных, например, с помощью KMS.

Ручная проверка:

Убедитесь, что данные хранятся в зашифрованном виде.

4.8 Используется шифрование данных на уровне приложения

Для шифрования данных на уровне приложения (client-side encryption) перед их отправкой в бакет Yandex Object Storage вы можете использовать следующие подходы:

- Интеграция Object Storage с сервисом Key Management Service для шифрования данных на уровне приложения (client-side encryption). Подробнее смотрите в разделе Рекомендуемые криптографические библиотеки.
- Шифрование данных на уровне приложения перед отправкой их в Object Storage с помощью сторонних библиотек. При использовании сторонних библиотек и собственных способов управления ключами следует убедиться, что схема работы, используемые алгоритмы и длины ключей соответствуют требованиям регуляторов.

Для шифрования данных на уровне приложения (client-side encryption) рекомендуется использовать следующие библиотеки:

- AWS Encryption SDK и его интеграцию с KMS;
- Google Tink и ее интеграцию с KMS;
- SDK Yandex Cloud вместе с любой другой криптографической библиотекой, совместимой с PCI DSS или другими стандартами, применяемыми в вашей компании.

Сравнение библиотек представлено в разделе Какой способ шифрования выбрать документации KMS.

Ручная проверка:

Убедитесь, что данные хранятся в зашифрованном виде.

Управление ключами

Для шифрования данных и управления ключами рекомендуется использовать Key Management Service. KMS предназначен для защиты данных в инфраструктуре Yandex Cloud, а также подходит для шифрования и расшифровки любых ваших данных.

KMS использует схему шифрования AES-GCM. Вы можете выбрать длину ключа: 128, 192 или 256 — и настроить период ротации ключей в зависимости своих потребностей.

4.9 В KMS используется аппаратный модуль безопасности (HSM)

Вы также можете создать ключ, все крипто-операции с которым будут выполняться только внутри специализированного аппаратного устройства, см. статью Аппаратный модуль безопасности (HSM).

Чтобы использовать HSM, при создании ключа выберите тип алгоритма AES-256 HSM. Все операции с этим ключом будут выполняться внутри HSM, дополнительные действия не требуются.

Рекомендуется использовать HSM для ключей KMS, это увеличивает уровень безопасности.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить ключи.
- В списке сервисов выберите **Key Management Service**.
- Перейдите на вкладку **Ключи**.
- Убедитесь, что в поле **Алгоритм шифрования** указан **AES-256 HSM**.
- Если указан AES-256 HSM, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для вывода списка всех ключей KMS организации и их алгоритмов шифрования:

```
export ORG_ID=yc.organization-manager.yandex
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do yc kms symmetric-key list --folder-id=$FOLDER_ID --format json | jq
-r '.[] | "KEY_ID " + .id + "FOLDER_ID " + .folder_id + "ALGORITM_ID "
+ .default_algorithm'
done;
done
```

- Если выведен пустой список, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Установите алгоритм шифрования для ключей KMS «AES-256 HSM».

4.10 Права управление ключами в KMS выданы контролируемым пользователям

Для доступа к сервису KMS необходимо использовать IAM-токен.

В случае автоматизации работы с KMS рекомендуется создать сервисный аккаунт и выполнять команды и скрипты от его имени. Если вы используете виртуальные машины, получите IAM-токен для сервисного аккаунта через механизм назначения сервисного аккаунта виртуальной машине. Другие способы получения IAM-токена для сервисного аккаунта приведены в статье Получение IAM-токена для сервисного аккаунта документации IAM.

Рекомендуется выдавать пользователям и сервисным аккаунтам гранулярные доступы на конкретные ключи сервиса KMS, см. статью Управление доступом в Key Management Service документации KMS.

Подробнее о мерах безопасности при управлении доступом читайте в статье Аутентификация и управление доступом.

Для того, чтобы проверить права доступа на ключ KMS необходимо проверить, у кого есть права:

- на организацию, облако, каталоги с правами: admin, editor, kms.admin, kms.editor, kms.keys.encrypterDecrypter;
- на ключи: kms.keys.encrypterDecrypter и kms.editor.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить права на ключ.
- Перейдите на вкладку **Права доступа**.
- Убедитесь, что роли admin, editor, kms.admin, kms.editor, kms.keys.encrypterDecrypter имеют только контролируемые пользователи.
- Проверить права доступа на сами ключи возможно только через CLI.

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска учётных записей на уровне организации:

```
export ORG_ID=bpf4c0lctf2t734l95ui
yc organization-manager organization list-access-bindings --
id=${ORG_ID} --format=json | jq -r '.[] | select(.role_id=="admin" or
.role_id=="editor" or .role_id=="kms.admin" or .role_id=="kms.editor"
or .role_id=="kms.keys.encrypterDecrypter")'
```

- Если в списке отсутствуют учётные записи, рекомендация выполнена. В противном случае перейдите к п. «Инструкции и решения по выполнению».
- Найдите учётные записи с назначенными ролями на уровне облаков:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do yc resource-manager cloud list-access-bindings --id=$CLOUD_ID --
format=json | jq -r '.[] | select(.role_id=="admin" or
.role_id=="editor" or .role_id=="kms.admin" or .role_id=="kms.editor"
or .role_id=="kms.keys.encrypterDecrypter")'
done
```

- Если в списке отсутствуют учётные записи, рекомендация выполнена. В противном случае перейдите к п. «Инструкции и решения по выполнению».

- Выполните команду для поиска учётных записей с назначенными примитивными ролями на уровне всех каталогов в ваших облаках:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id'); \
do yc resource-manager folder list-access-bindings --id=$FOLDER_ID --
format=json | jq -r '.[] | select(.role_id=="admin" or
.role_id=="editor" or .role_id=="kms.admin" or .role_id=="kms.editor"
or .role_id=="kms.keys.encrypterDecrypter")' && echo $FOLDER_ID
done;
done
```

- Если в списке отсутствуют учётные записи, рекомендация выполнена. В противном случае перейдите к п. «Инструкции и решения по выполнению».

- Найдите учётные записи с назначенными ролями на уровне ключей:

```
export ORG_ID=yc.organization-manager.yandex
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for KEY in $(yc kms symmetric-key list --folder-id=$FOLDER_ID --
format=json | jq -r '.[].id'); \
do yc kms symmetric-key list-access-bindings --id $KEY --format json
done;
done;
done
```

Инструкции и решения по выполнению:

Проконтролируйте, кому предоставлен доступ к ключам KMS.

4.11 Для KMS ключей включена ротация

Для повышения безопасности инфраструктуры рекомендуется разделить ключи шифрования на две группы:

1. Ключи для сервисов, которые обрабатывают критичные данные, но не хранят их. Например, Message Queue, Cloud Functions.
2. Ключи для сервисов, которые хранят критичные данные. Например, Managed Services for Databases.

Для первой группы рекомендуется настроить автоматическую ротацию с периодом ротации больше, чем срок обработки данных в этих сервисах. По истечении периода ротации старые версии должны быть удалены. При автоматической ротации и удалении старых версий ключей ранее обработанные данные не могут быть восстановлены и расшифрованы.

Для сервисов хранения данных рекомендуется использовать либо ручные процедуры ротации, либо автоматическую ротацию ключей в зависимости от внутренних процедур обработки критичных данных.

Безопасным значением для AES-GCM является шифрование 4 294 967 296 (=232) блоков. После достижения этого количества шифрованных блоков необходимо создать новую версию ключа шифрования данных. Подробнее про режим работы AES-GCM см. в материалах NIST.

> Удаление какой-либо версии ключа равносильно уничтожению всех данных, зашифрованных с ее помощью. Ключ можно защитить от удаления с помощью установки параметра deletionProtection, однако этот параметр не защищает от удаления отдельных версий.

Подробнее о ротации ключей см. в разделе Версия ключа документации KMS.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить ключи.
- В списке сервисов выберите **Key Management Service**.
- Перейдите в настройки ключа.
- Найдите параметр **Период ротации**.
- Если в параметре указано любое значение, отличное от **Нет ротации**, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для вывода списка всех ключей KMS организации и их алгоритмов шифрования:

```
export ORG_ID=yc.organization-manager.yandex
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do yc kms symmetric-key list --folder-id=$FOLDER_ID --format=json | jq
-r '.[] | select(.rotation_period | not)' | jq -r '.id'
done;
done
```

- Если выведен пустой список, то рекомендация выполняется.
  Если нет, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Установите период ротации для ключей.

4.12 Убедитесь, что для KMS ключей включена защита от удаления

Удаление KMS ключа приводит к гарантированному удалению данных, поэтому необходимо защищать ключи от непреднамеренного удаления. В KMS существует соответствующая функция.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить ключи.
- В списке сервисов выберите **Key Management Service**.
- Перейдите в настройки ключа.
- Найдите параметр **Защита от удаления**.
- Если в параметре указано **Да**, рекомендация выполняется.
  В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для вывода списка всех ключей KMS без защиты от удаления:

```
export ORG_ID=yc.organization-manager.yandex
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do yc kms symmetric-key list --folder-id=$FOLDER_ID --format=json | jq
-r '.[] | select(.deletion_protection | not)' | jq -r '.id'
done;
done
```

- Если выведен пустой список, рекомендация выполняется.
  В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Установите защиту от удаления.

Управление секретами

Критичные данные и секреты для доступа к данным (токены аутентификации, API-ключи, ключи шифрования и т. п.) не следует использовать в открытом виде в коде, в названиях и описаниях объектов облака, в метаданных виртуальных машин и т. д. Вместо этого используйте сервисы для хранения секретов, такие как Lockbox или HashiCorp Vault.

4.13 В организации используется сервис Secret Management — Yandex Lockbox

Критичные данные и секреты для доступа к данным (токены аутентификации, API-ключи, ключи шифрования и т. п.) не следует использовать в открытом виде в коде, в названиях и описаниях объектов облака, в метаданных виртуальных машин и т. д. Вместо этого используйте сервисы для хранения секретов, такие как Lockbox или HashiCorp Vault (из Cloud Marketplace).

Сервис Lockbox обеспечивает безопасное хранение секретов только в зашифрованном виде. Шифрование выполняется с помощью KMS. Для разграничения доступа к секретам используйте сервисные роли.

Инструкции по работе с сервисом см. в документации Lockbox.

Vault позволяет использовать KMS в качестве доверенного сервиса для шифрования секретов. Реализуется это через механизм Auto Unseal.

Для хранения секретов с помощью Vault можно использовать виртуальную машину на основе образа из Cloud Marketplace с предустановленной сборкой HashiCorp Vault и поддержкой Auto Unseal. Инструкция по настройке Auto Unseal приведена в статье Auto Unseal в Hashicorp Vault документации KMS.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить секреты.
- В списке сервисов выберите **Lockbox**.
- Убедитесь, что используется как минимум один секрет Lockbox.
- Найдите параметр **Защита от удаления**.
- Если используется Lockbox, либо среди виртуальных машин или сущностей k8s находится установленный Hashicorp Vault, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка через CLI:

- Посмотрите доступные вам организации и зафиксируйте необходимый ID:

```
yc organization-manager organization list
```

- Выполните команду для поиска как минимум одного секрета Lockbox:

```
export ORG_ID=<ID_организации>
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do yc lockbox secret list --folder-id=$FOLDER_ID --format=json
done;
done
```

- Если выведен пустой список, рекомендация выполняется. Если нет, перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Храните секреты в Lockbox либо Hashicorp Vault из Marketplace.

4.14 Для Serverless Containers и Cloud Functions используются Lockbox секреты

При работе с Serverless Containers или Cloud Functions часто возникает необходимость использовать секрет (токен, парол и т.д.).

Если указать секретную информацию в переменных окружения, она может быть доступна для просмотра любого пользователю облака с правами на просмотр и использование функции и влечёт за собой риски ИБ.

Рекомендуется использовать для этих целей интеграцию Serverless с Lockbox. Вы можете указать конкретный секрет из сервиса Yandex Lockbox и сервисный аккаунт с правами на данный секрет для использования его в функции или контейнере.

Рекомендуется убедиться, что секреты используются именно таким образом.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить функции.
- В списке сервисов выберите **Cloud Functions**.
- Перейдите в настройки функции, на вкладку **Редактор**.
- Найдите параметр **Защита от удаления**.

- Если в параметрах каждого объекта указано **Секреты Lockbox** или отсутствуют env с секретными данными, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Проверка YC CLI:

- Выполните команду для поиска всех облачных функций, которые не используют секреты Lockbox и убедитесь, что в данных функциях не используются секретные данные в env:

```
export ORG_ID=bpf4c0lctf2t734l95ui
for CLOUD_ID in $(yc resource-manager cloud list --organization-
id=${ORG_ID} --format=json | jq -r '.[].id');
do for FOLDER_ID in $(yc resource-manager folder list --cloud-
id=$CLOUD_ID --format=json | jq -r '.[].id');
do for VER in $(yc serverless function version list --folder-
id=$FOLDER_ID --format=json | jq -r '.[].id'); \
do yc serverless function version get $VER --format=json | jq -r '. |
select(.secrets | not)' | jq -r '.id'
done;
done;
done
```

- Если выведен пустой список, рекомендация выполняется. В противном случае перейдите к п. «Инструкции и решения по выполнению».

Инструкции и решения по выполнению:

Удалите секретные данные из env и воспользуйтесь функционалом интеграции с Lockbox.

4.15 При работе Container Optimized Image используется шифрование секретов

KMS предоставляет возможность шифрования секретов, используемых в конфигурации Terraform, в частности, для передачи секретов на виртуальную машину в зашифрованном виде. См. инструкцию в разделе Шифрование секретов в Hashicorp Terraform документации KMS. Передача секретов через переменные окружения в открытом виде небезопасна, поскольку они отображаются в свойствах ВМ.

Решение: Шифрование секретов в Terraform для передачи в ВМ с Container Optimized Image

Другие рекомендации по безопасному использованию Terraform см. в статье Безопасная конфигурация: Terraform.

5. Сбор, мониторинг и анализ аудитных логов

Введение

Аудитные логи (журналы аудита) — это записи обо всех событиях в системе, включая доступ к ней и выполненные операции. Сбор и проверка аудитных логов позволяют контролировать соблюдение установленных процедур и стандартов безопасности и выявить изъяны в механизмах безопасности.

События в аудитных логах относятся к различным уровням:

- уровень Yandex Cloud — события, происходящие с ресурсами Yandex Cloud;
- уровень ОС;
- уровень приложений;
- уровень сети (Flow Logs).

> О событиях Kubernetes читайте в разделе Сбор, мониторинг и анализ аудитных логов в Managed Service for Kubernetes.

5.1 Включён сервис Audit Trails на уровне организации

Основным инструментом сбора логов уровня Yandex Cloud является сервис Audit Trails. Сервис позволяет собирать аудитные логи о происходящих с ресурсами Yandex Cloud событиях и загружать эти логи в бакет Object Storage, Data Streams или лог-группу Cloud Logging для дальнейшего анализа или экспорта. См. инструкцию, как запустить сбор логов, а также формат и справочник событий.

Для сбора метрик, анализа некоторых событий уровня Yandex Cloud и настройки оповещений рекомендуется использовать сервис Monitoring. С его помощью возможно отслеживать, например, резкое возрастание нагрузки на Compute Cloud, RPS сервиса Application Load Balancer, значительные изменения в статистике событий сервиса Identity and Access Management.

Кроме того, Monitoring можно применять для мониторинга работоспособности самого сервиса Audit Trails и мониторинга событий безопасности. Выгрузка метрик в SIEM-систему возможна через API, см. инструкцию.

Решение: Мониторинг Audit Trails и событий безопасности с помощью Monitoring

Аудитные логи возможно экспортировать в лог-группу Cloud Logging или Data Streams и в SIEM-систему клиента для анализа информации о событиях и инцидентах.

Список важных событий уровня Yandex Cloud для поиска в аудитных логах:

Решение: поиск важных событий безопасности в аудитных логах

Audit Trails возможно включить на уровне каталога, облака и организации. Рекомендуется включать Audit Trails на уровне всей организации — это позволит централизованно собирать аудитные логи, например, в отдельное облако безопасности.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить функции.
- В списке сервисов выберите **Audit Trails**.
- Убедитесь, что в параметре **Фильтр** находится значение **Организация**.
- Дополнительно убедитесь, что назначение логов: bucket Object Storage, log группа Cloud Logging, Data Stream в рабочем состоянии и логи доступны для дальнейшего анализа.

5.2 События Audit Trails экспортируются в SIEM-системы

Решения для экспорта аудитных логов Yandex Cloud подготовлены для следующих SIEM-систем:

- Managed Service for Elasticsearch (ELK/Opensearch) — Сбор, мониторинг и анализ аудитных логов в Managed Service for Elasticsearch
- ArcSight — Сбор, мониторинг и анализ аудитных логов в SIEM ArcSight
- Splunk — Сбор, мониторинг и анализ аудитных логов в SIEM Splunk
- MaxPatrol SIEM — Сбор, мониторинг и анализ аудитных логов в MaxPatrol SIEM
- Wazuh — Решение: Сбор, мониторинг и анализ аудитных логов в Wazuh

Вы можете подробнее ознакомиться с MaxPatrol в разделе.

Для настройки экспорта в любые SIEM подходят утилиты GeeseFS или s3fs. Она позволяет смонтировать бакет Object Storage как локальный диск виртуальной машины. Далее на ВМ необходимо установить коннектор для SIEM и настроить вычитывание JSON-файлов из бакета. Либо утилиты совместимые с AWS Kinesis datastreams в случае, если вы направляете аудит логи в Yandex Data Streams.

Вы также можете анализировать аудит логи вручную, если у вас отсутствует SIEM система, одним из следующих образов (в порядке удобства):

- Поиск событий Yandex Cloud в Yandex Query;
- Загрузка аудитных логов в Yandex Managed Service for ClickHouse и визуализация данных в Yandex DataLens;
- Поиск событий Yandex Cloud в Cloud Logging;
- Поиск событий Yandex Cloud в Object Storage.

Ручная проверка:

Убедитесь, что аудитные логи из Audit Trails экспортируются для анализа в SIEM систему либо анализируются в облаке одним из способов.

5.3 Настроено реагирование на события Audit Trails

Вы можете реагировать на события Audit Trails средствами вашей SIEM системы либо вручную. Либо вы можете использовать автоматическое реагирование.

С помощью Cloud Functions можно настроить оповещения о событиях Audit Trails, а также автоматическое реагирование на вредоносные действия, например удаление опасных правил или прав доступа.

Решение: уведомления и реагирование на события ИБ Audit Trails с помощью IAM / Cloud Functions + Telegram

5.4 Выполнен hardering бакета Object Storage, где хранятся аудит логи Audit Trails

Убедитесь, что в случае записи аудит логов Audit Trails в bucket Object Storage сам бакет настроен в соответствии с лучшими практиками безопасности:

- 4.1 В Yandex Object Storage включено шифрование данных at rest с помощью ключа KMS.
- 3.8 В Object Storage включён механизм логирования действий с бакетом.
- 3.8 В Object Storage включена функция **Блокировка версии объекта** (object lock).
- 3.7 В Object Storage используются политики доступа (Bucket Policy).
- 3.6 Отсутствует публичный доступ к бакету Object Storage.

Вы можете воспользоваться решением для настройки безопасного бакета Object Storage с помощью Terrafrom.

Ручная проверка:

Выполните проверку вручную.

5.5 Выполняется сбор аудит логов с уровня ОС

При использовании облачных сервисов по модели IaaS и использовании групп узлов Kubernetes клиент отвечает за безопасность ОС и выполняет сбор событий уровня ОС самостоятельно. Для сбора стандартных событий, которые генерирует ОС, и их экспорта в SIEM-систему клиента существуют бесплатные инструменты, такие как:

- Osquery;
- Filebeat (ELK);
- Wazuh.

Дополнительные опции генерации событий можно реализовать с помощью утилиты Auditd для Linux, Sysmon для Windows.

Системные метрики Linux (процессор, память, диск) можно собирать с помощью компонента Unified Agent сервиса Monitoring. Также события ОС можно экспортировать в Cloud Logging с помощью плагина Fluent bit либо в Data Streams.

Для описания событий, которые нужно искать в логах, рекомендуем использовать формат Sigma, поддерживаемый популярными SIEM-системами. Репозиторий Sigma содержит библиотеку событий, описанных в этом формате.

Чтобы получать точную хронологию событий уровня ОС и приложений, настройте синхронизацию часов по инструкции.

Ручная проверка:

Выполните проверку вручную.

5.6 Выполняется сбор аудит логов с уровня приложений

Сбор событий уровня приложений, развёрнутых на ресурсах Compute Cloud, клиент может выполнять самостоятельно. Например, записывать логи приложения в файл и передавать их в SIEM-систему с помощью инструментов, перечисленных в подразделе выше.

Ручная проверка:

Выполните проверку вручную.

5.7 Выполняется сбор логов с уровня сети

Запись событий о сетевом трафике VPC (Flow Logs) на текущий момент может выполняться только средствами клиента. Для сбора и передачи событий могут использоваться решения из Cloud Marketplace (например, NGFW, IDS/IPS, сетевые продукты) либо бесплатное ПО. Также сбор логов уровня сети возможно выполнять с помощью различных агентов - HIDS и др.

Ручная проверка:

Выполните проверку вручную.

6. Управление уязвимостями

Введение

Yandex Cloud отвечает за управление уязвимостями и обновлениями безопасности в управляемых сервисах. Клиент отвечает за управление уязвимостями и обновлениями безопасности для всех остальных компонентов системы.

Пример разделения ответственности за управление уязвимостями и обновлениями безопасности см. в разделе **Requirement 5** матрицы разделения ответственности PCIDSS.

6.1 Для образов контейнеров используется сканер уязвимостей

Рекомендуется использовать сканер уязвимостей в образах, встроенный в Container Registry.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить образы.
- В списке сервисов выберите **Container Registry**.
- Перейдите в каждый образ и проверьте столбец **Статус сканирования**.

6.2 Выполняется сканирование уязвимостей на уровне облачных IP-адресов

Рекомендуем клиентам самостоятельно выполнять сканирование хостов на наличие уязвимостей. Облачные ресурсы поддерживают возможность установки собственных виртуальных образов сканеров уязвимостей или программных агентов на хостах.

Сетевые сканеры выполняют сканирование хостов, доступных по сети. Как правило, в сетевых сканерах возможна настройка аутентификации.

Примеры бесплатных сетевых сканеров:

- Nmap;
- OpenVAS;
- OWASP ZAP.

Пример бесплатного сканера, который работает в виде агента на хостах: Wazuh. Wazuh может также использоваться в качестве системы обнаружения вторжений (host-based intrusion detection system — IDS).

Вы также можете воспользоваться решением из Cloud Marketplace.

Ручная проверка:

Выполните проверку вручную.

6.3 Внешние сканирования безопасности выполняются по правилам облака

Клиенты, размещающие в Yandex Cloud собственное программное обеспечение, могут проводить для размещенного ПО внешние сканирования безопасности, в том числе тесты на проникновение. Вы можете проводить сканирование самостоятельно либо с привлечением подрядчиков. Подробнее в разделе Правила проведения внешних сканирований безопасности.

Ручная проверка:

Выполните проверку вручную.

6.4 Выстроен процесс обновлений безопасности

Клиент должен самостоятельно выполнять обновления безопасности в своей зоне ответственности. Возможно применение различных автоматизированных инструментов для централизованных автоматических обновлений ОС и ПО.

Yandex Cloud публикует бюллетени безопасности, чтобы оповещать клиентов о новых найденных уязвимостях и обновлениях безопасности.

6.5 Используется Web Application Firewall

Для снижения рисков, связанных с веб-атаками, рекомендуем использовать Web Application Firewall (WAF). Клиент может установить и обслуживать WAF самостоятельно, либо воспользоваться услугой Managed WAF.

Самостоятельная установка WAF

Образы WAF доступны в Cloud Marketplace. Типы лицензий и другая необходимая информация доступны в описаниях продуктов.

Решение: Отказоустойчивая эксплуатация PT Application Firewall на базе Yandex Cloud

Возможна также установка Wallarm WAF в Managed Service for Kubernetes. См. инструкцию в документации Wallarm. Тип лицензии — BYOL (лицензия, приобретенная у стороннего поставщика).

Managed WAF

Клиент получает облачный WAF как услугу у Yandex Cloud. Предоставляется доступ в личный кабинет, возможность просмотра статистики и управления. Для подключения услуги и получения детальной информации обратитесь к своему менеджеру, отдел продаж или в службу поддержки. Услуга оказывается в партнёрстве с Qrator.

Решение: Установка уязвимого веб-приложения (DVWA) в Yandex Cloud с помощью Terraform для тестирования Managed WAF

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить образы.
- В списке сервисов выберите **Compute Cloud**.
- Убедитесь, что среди ВМ есть хотя бы одна с образом WAF.

Ручная проверка:

Обратитесь к вашему менеджеру со стороны облака либо к службе безопасности вашей компании, чтобы узнать, используется ли managed WAF для вашей организации.

7. Резервное копирование

7.1 Используется Cloud Backup или механизм snapshot по расписанию

Убедитесь, что в вашей организации все виртуальные машины резервируются с помощью:

- снимков по расписанию;
- сервиса Cloud Backup.

Проверка в консоли управления:

- В консоли управления выберите облако или каталог, в которых необходимо проверить ВМ.
- В списке сервисов выберите **Compute Cloud**.
- Убедитесь, что на ВМ настроена политика снимков по расписанию.
- В списке сервисов выберите **Cloud Backup**.
- Убедитесь, что он включён.

8. Физическая безопасность

Платформа Yandex Cloud обеспечивает физическую безопасность дата-центров, см. подробное описание мер физической безопасности.

Если критичные данные передаются за пределы Yandex Cloud, то клиент отвечает за управление физическим доступом для всех мест обработки данных.