



# Пеликан

## Система Управления Рисками

### Инструкция по установке

*7 января 2025 г.*

## Содержание

---

Предустановочные действия .....	2
База данных .....	2
Веб приложение .....	3
Установка веб-приложения Pelican .....	6
Общие рекомендации по настройке параметров, влияющих на безопасность приложения .....	7
Параметры безопасности уровня конфигурационных файлов .....	8
Параметры безопасности, доступные к настройке из интерфейса приложения .....	10
Общие рекомендации по резервному копированию данных приложения: .....	11
Обновление веб-приложения Pelican .....	12
Первая инициализация и быстрые проверки .....	12
Системные Требования .....	13
Взаимодействие элементов системы, сервисы, порты и протоколы .....	13
Веб-приложение и сервер БД .....	13
Сервисы приложения .....	14

## Предустановочные действия

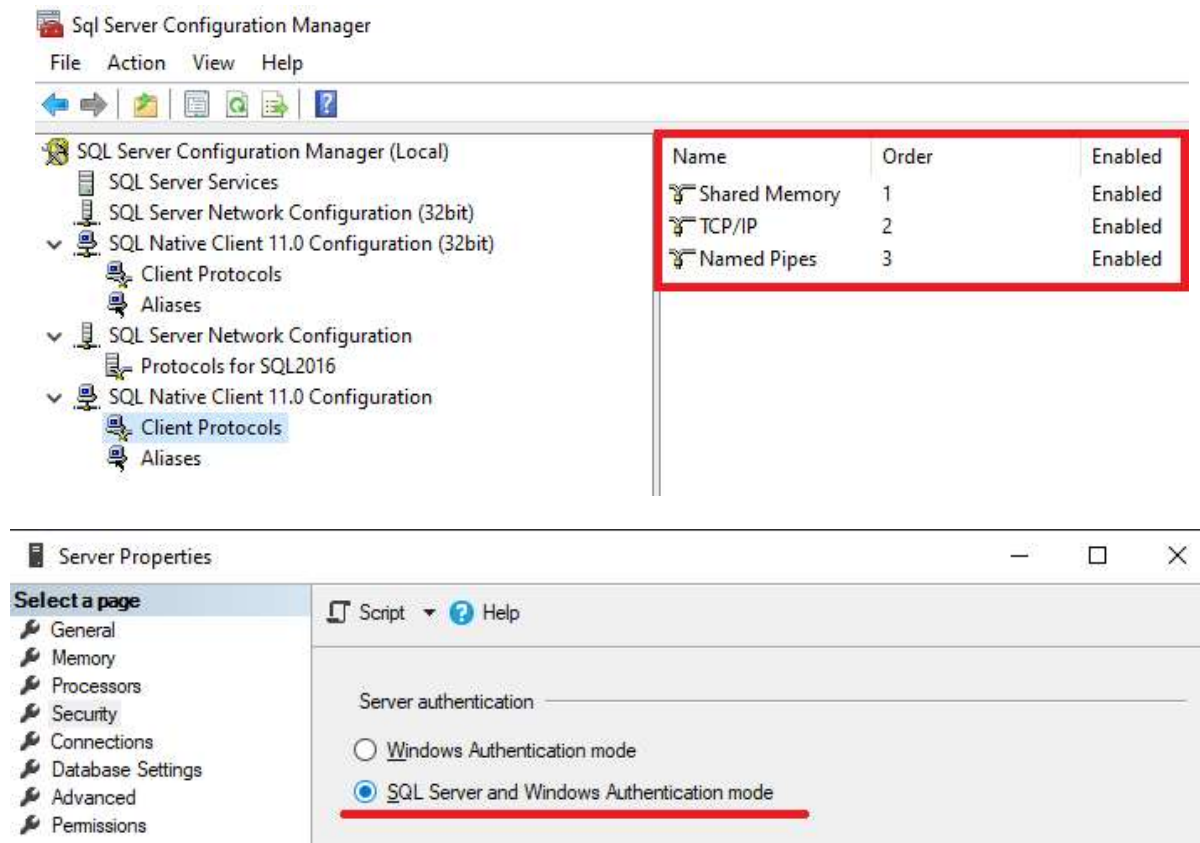
### База данных

Есть 2 сценария, в которых Pelican необходимо подключиться к базе данных SQL Server:

1. **Процесс установки и обновления** . Для завершения процесса установки/обновления пользователю базы данных должна быть назначена роль *системного администратора* .
2. **Регулярное использование**. Пользователю базы данных должна быть назначена обычная роль (права доступа на чтение/запись).

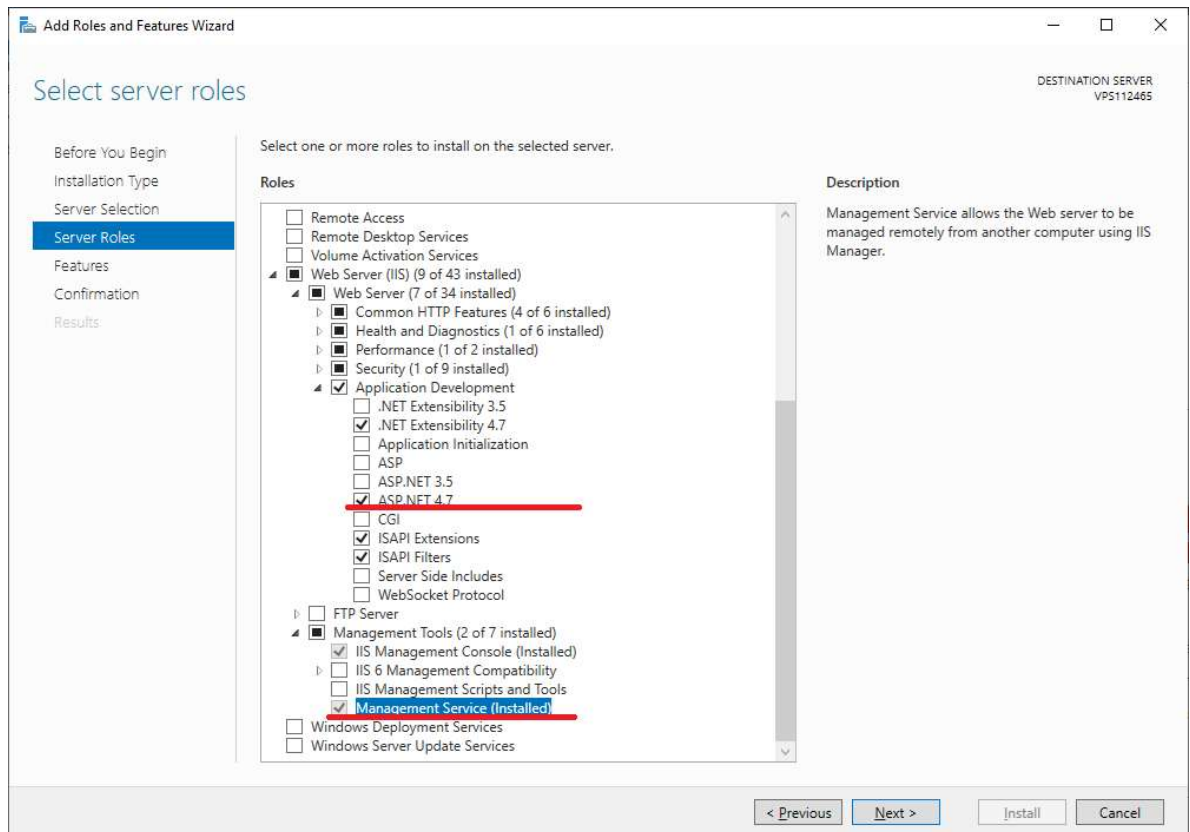
Перед установкой Pelican необходимо создать/выбрать пользователей базы данных для обоих приведенных выше сценариев. Информация для входа будет использоваться позже при настройке строк подключения к приложению. Также, роль **db\_owner** для созданной базы данных назначается пользователю постоянно, роль **sysadmin** должна быть назначена пользователю ТОЛЬКО на время установки приложения.

**Важное примечание.** Общая память, именованные каналы и протоколы TCP/IP должны быть включены в диспетчере конфигурации SQL Server перед установкой Pelican. Перезапустите экземпляр SQL Server после внесения любых изменений в диспетчере конфигурации SQL Server, чтобы изменения конфигурации вступили в силу.

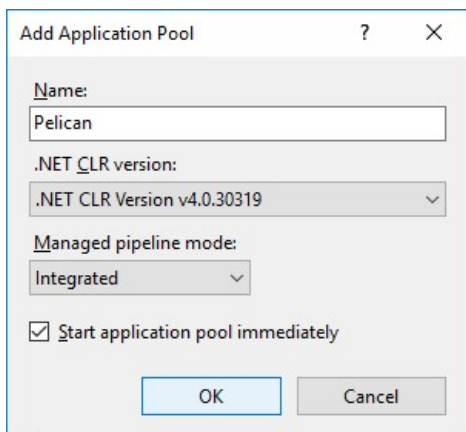


## Веб приложение

1. Установите IIS на сервер веб-приложений, что можно сделать с помощью «Мастера добавления ролей и компонентов» на сервере. Также должна быть включена поддержка ASP.NET 4.7 (или выше) в разделе «Роли сервера» (Веб-сервер (IIS) -> Веб-сервер -> Служба разработки и управления приложениями в разделе «Роли сервера» Веб-сервер (IIS) -> Средства управления).



2. Важно: Функция «Публикация WebDAV» должна быть отключена на сервере приложений, чтобы избежать конфликтов с Pelican WebAPI: Роли сервера -> Веб-сервер (IIS) -> Веб-сервер -> Общие функции HTTP -> Публикация WebDAV.
3. Загрузите и установите средство веб-развертывания 3.5 (или более поздней версии). Это можно сделать, загрузив автономный установщик Microsoft.  
(Официальная ссылка на версию 3.6: <https://www.microsoft.com/en-us/download/details.aspx?id=43717>)
4. Так же установите SQL DacFX от 18.8.  
(Официальная ссылка на версию 18.8: <https://go.microsoft.com/fwlink/?linkid=2164920>)
5. В диспетчере IIS создайте новый пул приложений для приложения Pelican. Задайте для удостоверения пула приложений учетную запись LocalSystem, разрешите взаимодействие с рабочим столом. Установите «Время простоя (минуты)» на 1000.



Add Application Pool

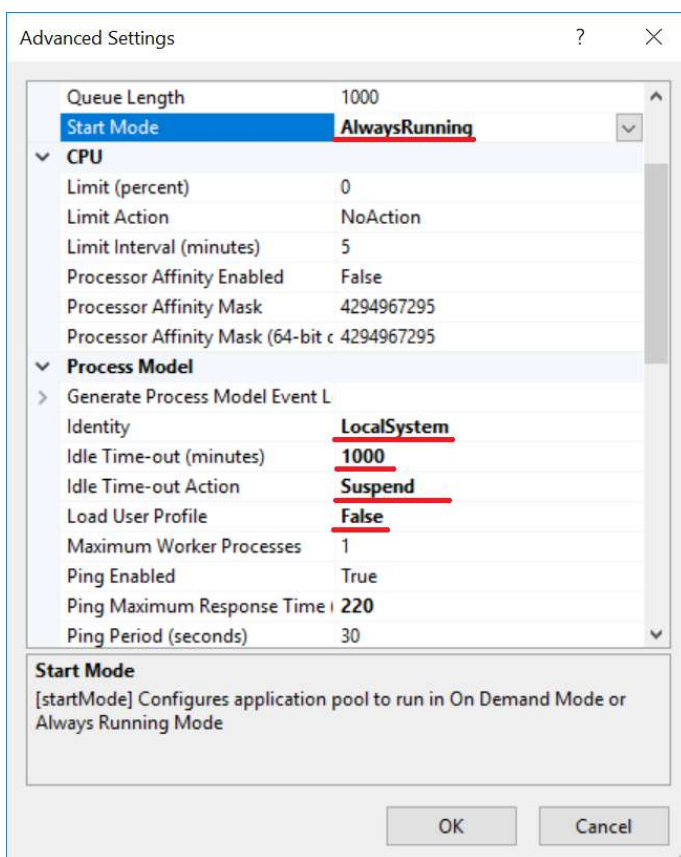
Name: Pelican

.NET CLR version: .NET CLR Version v4.0.30319

Managed pipeline mode: Integrated

Start application pool immediately

OK Cancel



Advanced Settings

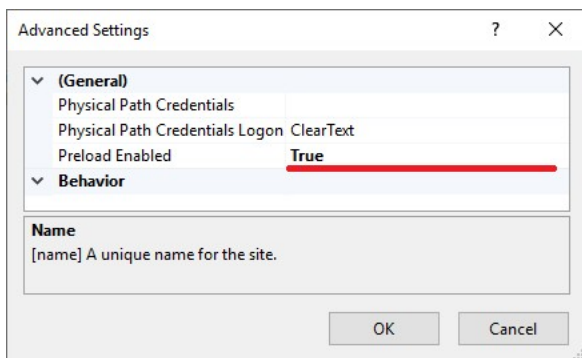
Queue Length	1000
Start Mode	<u>AlwaysRunning</u>
<b>CPU</b>	
Limit (percent)	0
Limit Action	NoAction
Limit Interval (minutes)	5
Processor Affinity Enabled	False
Processor Affinity Mask	4294967295
Processor Affinity Mask (64-bit)	4294967295
<b>Process Model</b>	
Generate Process Model Event L	
Identity	<u>LocalSystem</u>
Idle Time-out (minutes)	<u>1000</u>
Idle Time-out Action	<u>Suspend</u>
Load User Profile	<u>False</u>
Maximum Worker Processes	1
Ping Enabled	True
Ping Maximum Response Time	220
Ping Period (seconds)	30

**Start Mode**  
[startMode] Configures application pool to run in On Demand Mode or Always Running Mode

OK Cancel

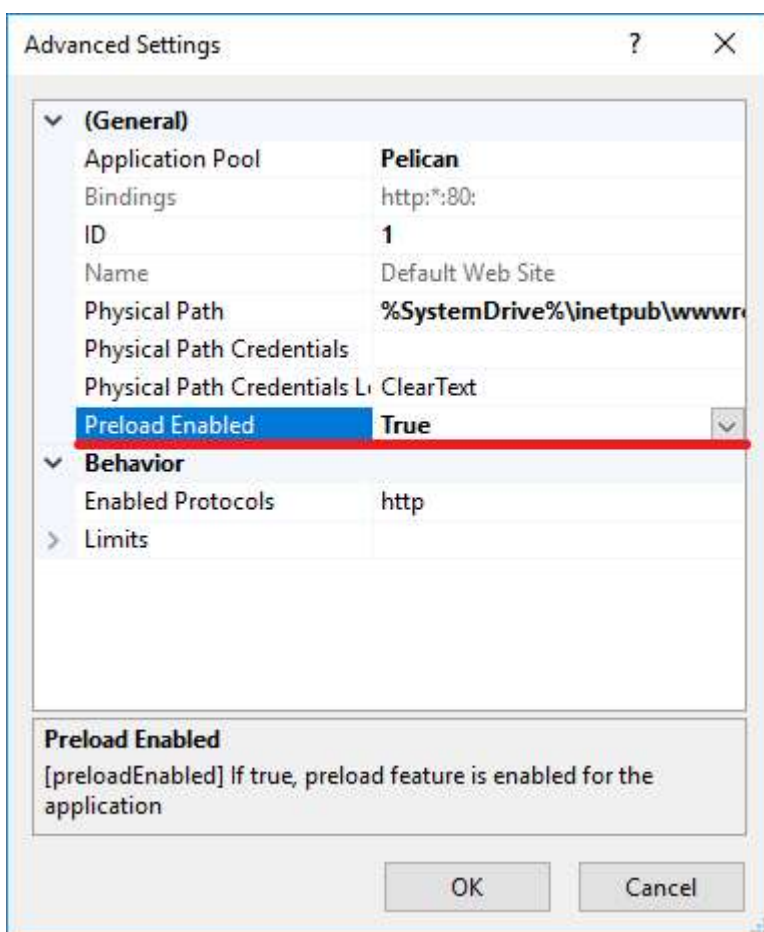
6. В диспетчере IIS создайте новый веб-сайт для веб-приложения или выберите существующий. Привяжите этот веб-сайт к пулу приложений, созданному на предыдущем шаге.
7. Включить предварительную загрузку для этого веб-сайта (см. Дополнительные параметры веб-сайта, панель «Действия» в диспетчере IIS)





8. Установите на сервер NuGet Provider (если есть доступ к интернету, можно использовать следующую команду PowerShell):

`Install-PackageProvider -Name "NuGet" -Force`



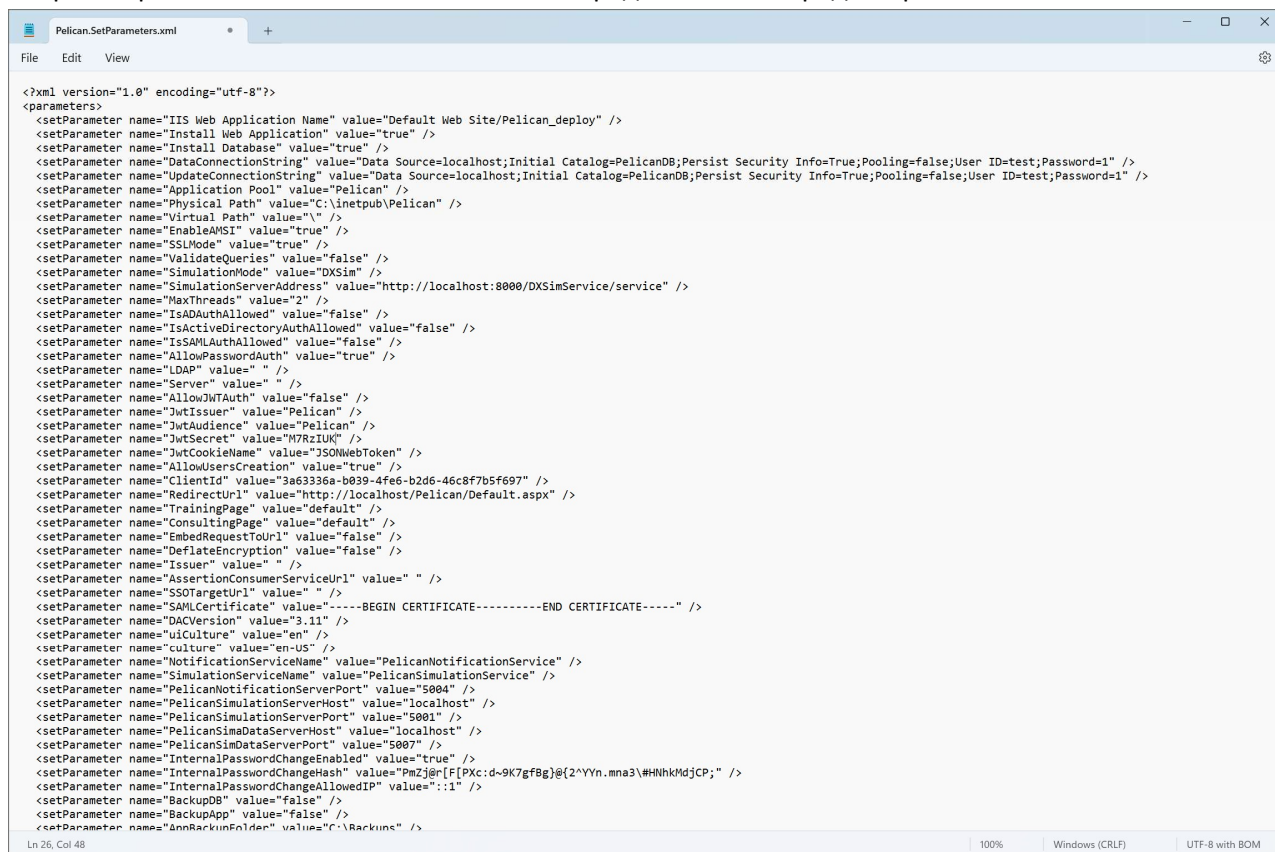
## Установка веб-приложения Pelican

Чтобы установить Pelican, выполните следующие действия:

1. Подключиться к серверу веб-приложений
2. Скопируйте и разархивируйте установочный пакет Pelican
3. Следующие три файла инсталляционного пакета должны быть размещены в одном каталоге:



4. Откройте файл «Pelican.SetParameters.xml» в предпочитаемом редакторе:



```
<?xml version="1.0" encoding="utf-8"?>
<parameters>
  <setParameter name="IIS Web Application Name" value="Default Web Site/Pelican_deploy" />
  <setParameter name="Install Web Application" value="true" />
  <setParameter name="Install Database" value="true" />
  <setParameter name="DataConnectionString" value="Data Source=localhost;Initial Catalog=PelicanDB;Persist Security Info=True;Pooling=false;User ID=test;Password=1" />
  <setParameter name="UpdateConnectionString" value="Data Source=localhost;Initial Catalog=PelicanDB;Persist Security Info=True;Pooling=false;User ID=test;Password=1" />
  <setParameter name="Application Pool" value="Pelican" />
  <setParameter name="Physical Path" value="C:\inetpub\Pelican" />
  <setParameter name="Virtual Path" value="" />
  <setParameter name="EnableAMSI" value="true" />
  <setParameter name="SSLMODE" value="true" />
  <setParameter name="ValidateQueries" value="false" />
  <setParameter name="SimulationMode" value="DXSim" />
  <setParameter name="SimulationServerAddress" value="http://localhost:8000/DXSimService/service" />
  <setParameter name="MaxThreads" value="2" />
  <setParameter name="IsADAuthAllowed" value="false" />
  <setParameter name="IsActiveDirectoryAuthAllowed" value="false" />
  <setParameter name="IsSAMLAuthAllowed" value="false" />
  <setParameter name="AllowPasswordAuth" value="true" />
  <setParameter name="LDAP" value="" />
  <setParameter name="Server" value="" />
  <setParameter name="AllowJwtAuth" value="false" />
  <setParameter name="JwtIssuer" value="Pelican" />
  <setParameter name="JwtAudience" value="Pelican" />
  <setParameter name="JwtSecret" value="M7RzIUk" />
  <setParameter name="JwtCookieName" value="JSONWebToken" />
  <setParameter name="AllowUsersCreation" value="true" />
  <setParameter name="ClientId" value="3a63336a-b039-4fe6-b2d6-46c8f7b5f697" />
  <setParameter name="RedirectUrl" value="http://localhost/Pelican/Default.aspx" />
  <setParameter name="TrainingPage" value="default" />
  <setParameter name="ConsultingPage" value="default" />
  <setParameter name="EmbedRequestToUrl" value="false" />
  <setParameter name="DeflateEncryption" value="false" />
  <setParameter name="Issuer" value="" />
  <setParameter name="AssertionConsumerServiceUrl" value="" />
  <setParameter name="SSOTargetUrl" value="" />
  <setParameter name="SAMLCertificate" value="-----BEGIN CERTIFICATE-----END CERTIFICATE-----" />
  <setParameter name="DACVersion" value="3.11" />
  <setParameter name="uiCulture" value="en" />
  <setParameter name="culture" value="en-US" />
  <setParameter name="NotificationServiceName" value="PelicanNotificationService" />
  <setParameter name="SimulationServiceName" value="PelicanSimulationService" />
  <setParameter name="PelicanNotificationServerPort" value="5004" />
  <setParameter name="PelicanSimulationServerHost" value="localhost" />
  <setParameter name="PelicanSimulationServerPort" value="5001" />
  <setParameter name="PelicanSimDataServerHost" value="localhost" />
  <setParameter name="PelicanSimDataServerPort" value="5007" />
  <setParameter name="InternalPasswordChangeEnabled" value="true" />
  <setParameter name="InternalPasswordChangeHash" value="PmZj8r[F]Pxc:d-9K7gF8g}@{2^Yn.mna3#HhKhdjCP; } />
  <setParameter name="InternalPasswordChangeAllowedIP" value="::1" />
  <setParameter name="BackupDB" value="false" />
  <setParameter name="BackupApp" value="false" />
  <setParameter name="AnnRackInFolder" value="C:\Rackinc" />

```

5. Заполните информацию об установке Pelican:
  - 5.1. Для параметра «Имя приложения IIS» введите значение в следующем формате: «Имя веб-сайта Pelican/имя приложения Pelican», например: «Веб-сайт по умолчанию/Pelican».
  - 5.2. Убедитесь, что для параметров «Установить веб-приложение» и «Установить базу данных» установлено значение true.
  - 5.3. Для параметра DataConnectionString введите строку подключения к базе данных Pelican (пользователь базы данных с обычными правами доступа). Например: «Data

Source=localhost;Initial Catalog=PelicanDB;Persist Security Info=True;Pooling=false;User ID=PelicanUser;Password=P@\$w0rd». Обратите внимание, что указанный пользователь SQL Server должен быть создан перед запуском сценария установки.

**Также обратите внимание, что строка подключения должна содержать *Pooling=false*; атрибут.**

- 5.4. Для параметра «UpdateConnectionString» введите строку подключения к базе данных Pelican (пользователь базы данных с правами доступа sysadmin). Это соединение будет использоваться только в процессе установки и обновления. Например: «Data Source=localhost;Initial Catalog=PelicanDB;Persist Security Info=True;Pooling=false;User ID=PelicanUser;Password=P@\$w0rd». Обратите внимание, что указанный пользователь SQL Server должен быть создан перед запуском сценария установки. **Также обратите внимание, что строка подключения должна содержать *Pooling=false*; атрибут.**
- 5.5. Для параметра «Физический путь» укажите путь к папке приложения. Например: «C:\inetpub\wwwroot\Pelican».
- 5.6. Параметр «SSLMode» должен иметь значение true, чтобы разрешить работу через https-соединение для экземпляра Pelican.
- 5.7. Параметры IsActiveDirectoryAuthAllowed (по умолчанию: false), IsSAMLAuthAllowed (по умолчанию: true), AllowJWTAuth (по умолчанию: false), AllowPasswordAuth (по умолчанию: true) можно использовать для настройки типов аутентификации для приложения. Эти настройки также можно изменить позже через приложение AppSettings.config.
- 5.8. Параметр SimulationServiceName можно использовать для указания имени службы моделирования Pelican. Это важно, когда на одном сервере развернуто несколько экземпляров Pelican.
- 5.9. Параметр NotificationServiceName можно использовать для указания имени службы уведомлений Pelican. Это важно, когда на одном сервере развернуто несколько экземпляров Pelican.
- 5.10. Параметры PelicanNotificationServerPort, PelicanSimulationServerPort и PelicanSimDataServerPort задают порты для сервисов Pelican. Все эти значения не должны мешать друг другу.
- 5.11. Сохраните и закройте файл Pelican.SetParameters.xml.
6. Откройте PowerShell с правами администратора и запустите скрипт pelicandeploy.ps1.
7. Если на экране появляется всплывающее сообщение с вопросом о «доверенном издателе», выберите вариант [A] (всегда запускать).

## Общие рекомендации по настройке параметров, влияющих на безопасность приложения

В нашем приложении предусмотрен набор параметров безопасности, которые обеспечивают защиту данных и предотвращение несанкционированного доступа. Однако, поскольку приложение разворачивается в периметре заказчика, необходимо также учитывать и использовать существующие и уже развернутые средства безопасности. Это включает в себя



такие аспекты, как аутентификация пользователей, шифрование данных, использование межсетевых экранов (WAF), а также регулярное резервное копирование данных.

Ниже мы расскажем про доступные в системе параметры с указанием наших рекомендаций.

### Параметры безопасности уровня конфигурационных файлов

Набор параметров из файла «Pelican.SetParameters.xml» содержит различные настройки, влияющие на функционирование приложения. В процессе установки значения этих параметров используются скриптом для подстановки в конфигурационные файлы приложения. Кроме того, после проведения установки можно дополнительно изменить значение параметров если, например, есть необходимость в смене протокола, портов взаимодействия с сервисом симуляции или шифровании конфигурационных файлов.

Основная рекомендация по настройке безопасности приложения – использовать доступные к изменению параметры приложения для настройки в соответствии с политикой безопасности организации.

Ниже мы подробнее остановимся на параметрах безопасности и наших рекомендациях.

### Шифрование строк подключения

Мы рекомендуем шифровать конфигурационные файлы приложения после завершения настройки. Такое шифрование делается встроенными средствами Microsoft IIS и позволяет защитить, например, пароли, находящиеся в конфигурационном файле ConnectionStrings.config в открытой форме.

При установке приложения для шифрования можно использовать параметр по умолчанию: `<setParameter name="EncryptConnectionString" value="true" />`, однако часто после установки требуется дополнительная настройка, которая может быть затруднена шифрованием. В этом случае рекомендуется не шифровать файл при установке и выполнить шифрование позже, с помощью команды `aspnet_regiis.exe` с ключом `-ref`.

**Важно!** Данный параметр является директивой для скрипта установки, используется только в момент установки и не хранится в конфигурационном файле приложения.

### Проверка файлов на вредоносное содержимое

Мы рекомендуем включать интерфейс проверки программ на вредоносное содержимое. В этом случае при загрузке файлов в систему будет выполняться их проверка тем инструментом, который установлен в системе как основной.

Параметр можно задать либо во время установки ключом `<setParameter name="EnableAMSI" value="true" />`, либо уже в `AppSettings.config` в секции:

```
<!--AMSI validation of the uploading files-->  
<!--Changing the EnableAMSI requires restarting the application-->  
<add key="AMSIEnabled" value="true" />
```

## Протокол SSL

Мы рекомендуем настраивать доступ к веб-приложению по протоколу https. Если на момент установки приложения все необходимые сертификаты уже есть, то достаточно использовать ключ `<setParameter name="SSLMode" value="true" />`. Однако, зачастую необходимые для настройки сертификаты выпускаются уже после установки. В таком случае есть возможность включить режим SSL в конфигурационных файлах – для этого в файле `AppSettings.config` потребуется изменить следующий параметр:

```
<!--For SSL-->  
<add key="SSLMode" value="false" />  
<!--/For SSL-->
```

В файле `web.config` установить настройку безопасности для cookie:

```
<httpCookies httpOnlyCookies="true" requireSSL="true" sameSite="Strict" />
```

## Валидация SQL запросов

В функционал приложения включены конструкторы отчетов, дэшбордов, КПЭ. Симуляционные модели также предоставляют возможность использовать язык запросов SQL для получения данных из базы. В целях безопасности мы рекомендуем включать в `AppSettings.config` валидацию запросов к базе данных – это позволяет ограничить возможности используемых запросов чтением разрешенной к использованию в отчетах, моделях, дашбордах информации. (нет доступа к логам, данным авторизации и т.д.)

```
<!--SQL queries validation-->  
<!--Changing the ValidateQueries requires restarting the application-->  
<add key="ValidateQueries" value="true" />
```

Если помимо белого списка хранимых процедур и представлений есть необходимость разрешить написание запросов к базе, то можно установить в `AppSettings.config` следующий параметр:

```
<add key="EnableCustomSQL" value="true" />
```

В этом случае будет дополнительно использоваться валидация запросов Devexpress, которая исключает некоторые ключевые слова SQL. Она исключает и обобщенные табличные выражения. Если есть необходимость использовать их, можно использовать следующий параметр (также в AppSettings.config):

```
<add key="DisableDefaultCustomSQLValidation" value="true" />
<!--/SQL queries validation-->
```

### Пароль внутренних резервных копий приложения

Резервные копии, создаваемые в приложении, защищены паролем. Данный пароль можно изменить в AppSettings.config:

```
<!--For Backups Password-->
  <add key="BackupsPassword" value="tGd=A6D#CHj7h-
K2fCzMK9%u5p#R6?MpTc~w" />
<!--/For Backups Password-->
```

**Важно!** Пароль не передается в интерфейс состояния объектов ИБ, он доступен только физически в файле конфигурации.

### Отключение роли sysadmin у пользователя БД после установки

Если сервер БД используется одновременно для хранения баз нескольких приложений, мы рекомендуем отключать у пользователя SQL, под которым приложение Pelican осуществляет доступ, роль sysadmin. Эта роль предоставляет избыточный набор прав и необходима только при первой установке или обновлении приложения, но не во время его штатной работы.

## Параметры безопасности, доступные к настройке из интерфейса приложения

### Таймаут сессии (минуты)

Данный параметр определяет максимальное время сохранения активной сессии без запросов от пользователя. По умолчанию время сохранения сессии составляет 180 минут. Мы рекомендуем устанавливать для этого параметра значение от 60 до 180 минут. Слишком короткое время может мешать работе пользователя, слишком длинное является потенциальным риском безопасности.

### Срок действия пароля пользователя (дни)

Данный параметр определяет период изменения пользовательского пароля. По умолчанию его значение равно 0, что транслируется в систему как никогда не истекающий пароль. При использовании локальных пользователей и аутентификации

на уровне системы, мы рекомендуем устанавливать данный параметр в соответствии с политикой безопасности организации.

#### Период деактивации пользователя (дни)

Данный параметр определяет срок, после которого статус пользователя (при отсутствии успешных логинов в системе за период) меняется на «Приостановлен». Данный статус не позволяет входить в систему, разблокировка возможно только через административную панель. По умолчанию мы устанавливаем этот период в 90 дней. Рекомендуем здесь также устанавливать значение в соответствии с политикой безопасности организации.

### Общие рекомендации по резервному копированию данных приложения:

Регулярное резервное сохранение данных сможет защитить вас от их потери в случае возникновения внештатных ситуаций. Создавая дубликаты и храня их в надежных местах (например, на внешнем жестком диске, в облачном хранилище или на резервном сервере), вы обеспечиваете легкий доступ к данным в случае чрезвычайной ситуации.

Существует несколько способов резервного копирования данных в приложении Pelican – резервное копирование приложения, резервное копирование базы данных, резервное копирование образа сервера приложения:

1. **Резервное копирование приложения** является самым легковесным, оно сохраняет все пользовательские данные из приложения и базы данных. Но имейте в виду – если приложение было повреждено во время обновления, для восстановления потребуется полная переустановка Pelican, что может занять некоторое время. Также в системе нет функции запланированного резервного копирования, поэтому это можно сделать только вручную. Вы можете получить доступ к резервным копиям приложения через Настройки -> Резервные копии. (опция доступна администраторам системы Пеликан)
2. **Резервное копирование образа сервера приложения:** рекомендуется создавать такую резервную копию перед любыми серьезными взаимодействиями с серверной частью системы (например, обновлением, переносом и т.д.). Эта резервная копия включает операционную систему, приложения, настройки и все ваши данные. Это ваша страховка на случай, если что-то пойдет не так. Если вы сделаете резервную копию перед обновлением, вы сможете восстановить ее, вернув все в исходное состояние. Также есть возможность настроить запланированное резервное копирование образа системы приложения и хранить несколько последних образов в безопасном месте.
3. **Резервное копирование базы данных:** в Пеликане мы работаем с базой данных MS SQL, поэтому можем использовать инструменты резервного копирования, специфичные для базы данных. Эти инструменты обеспечивают создание согласованных резервных копий ваших данных, позволяя восстанавливать базы

данных до определенного момента времени. Легко настроить запланированные резервные копии для обеспечения непрерывной защиты от потери данных.

**Резервное копирование образа сервера приложения следует использовать в сочетании с резервным копированием базы данных**, так как они заботятся о разных частях приложения (исключительный случай – когда ваше приложение и база данных находятся на одном сервере, что редко).

Храните резервные копии на внешних дисках, отдельно от основной системы. Это предотвращает потерю данных из-за отказов оборудования или вредоносного ПО, затрагивающего как вашу систему, так и резервные копии. Используйте облачные сервисы для резервного копирования вне офиса. Облачное хранилище обеспечивает избыточность и доступность из любого места.

## Обновление веб-приложения Pelican

Выполните следующие действия, чтобы обновить Pelican:

1. Выполните шаги 1–5 из раздела «Установка веб-приложения Pelican».
2. Мы настоятельно рекомендуем сделать резервную копию вашей текущей базы данных, прежде чем делать что-либо, что может изменить или повредить ваши данные. Вы можете выполнять резервное копирование базы данных с помощью инструментов Microsoft SQL Server Management Studio или включить резервное копирование базы данных в сценарии развертывания. Это можно сделать, установив для параметров «BackupDB» (резервное копирование базы данных с SQL-сервера) и «BackupApp» (архивирование и резервное копирование папки приложения) значение «true» в файле «Pelican.SetParameters.xml» и указав папки для хранения резервных копий базы данных и папок приложений в параметрах «DBBackupFolder» и «AppBackupFolder» (папки должны существовать перед резервным копированием).
3. Выполните шаги 6–7 из раздела «Установка веб-приложения Pelican». Обратите внимание, что папки «bin» и «Update» в каталоге веб-приложения будут очищены во время обновления. Если вы сделали дополнительные настройки в этих папках, рассмотрите возможность резервного копирования ваших изменений.

## Первая инициализация и быстрые проверки

При первом запуске система заполняет базу минимальным необходимым набором исходных данных. При инициализации будет создан первый пользователь Pelican с полными правами доступа к функциям программного обеспечения. Логин и пароль для первого пользователя:

Логин: Admin

Пароль: Pelican



После первого входа пользователь будет автоматически перенаправлен на страницу менеджера лицензий, где необходимо активировать лицензию Pelican. После этого приложение Pelican будет настроено и готово к работе. Просмотрите несколько разделов в меню Pelican (например, Риски, Драйверы, Контрольные Механизмы в разделе Риски Галстук-Бабочка), чтобы убедиться, что страницы открываются правильно.

## Системные Требования

### Сервер

- ОС: Windows Server 2016+ (платформа x 64)
- Процессор: Pentium или Pentium-совместимый, от 4 ядер
- Оперативная память : 8 ГБ +
- Место на диске: 100+ Гб свободного места
- Требования к программному обеспечению: ASP.NET 4.7+, IIS с включенным веб-развертыванием Web Deploy
- PowerShell 5.1+ (установлены модули веб-администрирования и SqlServer)
- NuGet-провайдер
- .NET Framework 4.8+
- База данных: MS SQL Server 2016+ (стандартный или корпоративный), с установленным модулем SQL DacFx 18.8+

### Клиент

- Экран: разрешение 1024x768 или выше
- Веб-браузер: Edge, Chrome, Firefox

## Взаимодействие элементов системы, сервисы, порты и протоколы

Система состоит нескольких элементов:

- Веб приложение на платформе ASP Web Forms. Разворачивается на базе IIS
- База данных MSSQL
- Сервис симуляции
- Сервис симуляции моделей КОР
- Сервис уведомлений

### Веб-приложение и сервер БД

Вся коммуникация с пользователем осуществляется через веб-приложение. Пользователи взаимодействуют с веб-приложением посредством браузера, через протокол http (порт по умолчанию 80) или https (порт по умолчанию 443). Изменить настройки порта и протокола можно в IIS. Одновременное использование http и https невозможно.

Доступ веб приложения к базе данных осуществляется через стандартные протоколы связи с MS SQL сервер через строку подключения. Порты и протоколы по умолчанию: 1433(1434) TCP. Изменение настроек соединения может потребовать дополнительной настройки строки соединения в конфигурационном файле.

При включении авторизации LDAP Active Directory взаимодействие с контроллером осуществляется согласно внешней настройке, стандартный порт по умолчанию 389 (протокол TCP)

При включении авторизации LDAPS Active Directory взаимодействие с контроллером осуществляется согласно внешней настройке, стандартный порт по умолчанию 636 (протокол TCP SSL)

Для интеграции с внешними системами используется REST API, работающий по протоколам HTTP (порт 80) или HTTPS (порт 443, порт и протокол зависят от настроек приложения). Формат обмена сообщениями: JSON. Возможен ручной обмен данными с внешними системами с использованием импорта или экспорта данных в файлы в формате .xlsx.

### Сервисы приложения

Сервисы симуляции и симуляции КОР устанавливаются на тот же сервер, что и веб-приложение, взаимодействуют только с приложением.

Для работы необходимо назначить для сервисов пользователя с правами не ниже LocalSystem.

Порты взаимодействия по умолчанию:

- Сервис симуляции: порт 5001, протокол GRPC
- Сервис симуляции КОР: порт 8000, протокол WCF

Сервис уведомления устанавливается на тот же сервер, что и веб-приложение, взаимодействует с приложением и smtp сервером (после настройки).

Порты взаимодействия по умолчанию:

- Сервис уведомлений - приложение: порт 5004, протокол GRPC
- Сервис уведомлений – smtp сервер: порты 25 (TCP)/ 587 (UDP), возможно использовать SSL. Настройка связи с SMTP осуществляется через административный доступ.

Значения портов по умолчанию могут быть заменены. Для GRPC сервисов достаточно изменений в конфигурационных файлах, WCF сервис требует дополнительной настройки через PowerShell.

**Адрес:** Россия, 362003, г. Владикавказ, ул. Тургеневская 193

**Телефон:** +7 8672 25 94 00

**Электронная почта:** [info@riskstrategy.ru](mailto:info@riskstrategy.ru)

[www.riskstrategy.ru](http://www.riskstrategy.ru)