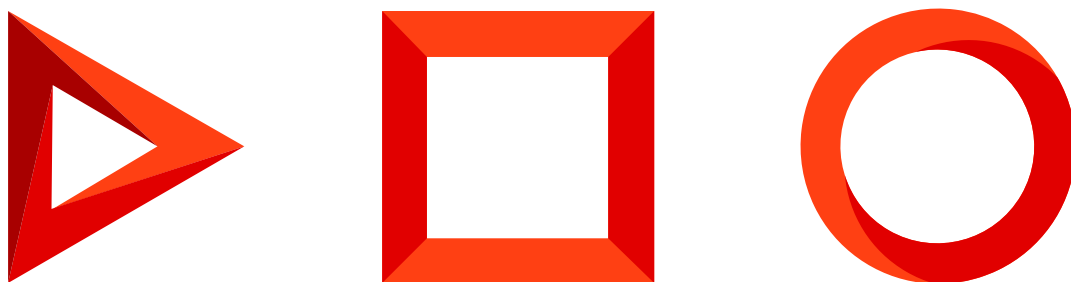


Сервер приложений .NET Framework на Windows

Версия 8.0



Эта документация предоставляется с ограничениями на использование и защищена законами об интеллектуальной собственности. За исключением случаев, прямо разрешенных в вашем лицензионном соглашении или разрешенных законом, вы не можете использовать, копировать, воспроизводить, переводить, транслировать, изменять, лицензировать, передавать, распространять, демонстрировать, выполнять, публиковать или отображать любую часть в любой форме или посредством любые значения. Обратный инжиниринг, дизассемблирование или декомпиляция этой документации, если это не требуется по закону для взаимодействия, запрещены.

Информация, содержащаяся в данном документе, может быть изменена без предварительного уведомления и не может гарантировать отсутствие ошибок. Если вы обнаружите какие-либо ошибки, сообщите нам о них в письменной форме.

Содержание

Проверить обязательные компоненты Windows	4
Проверить обязательные компоненты Windows Server 2016	5
Проверить обязательные компоненты Windows 10	15
Настроить ConnectionStrings.config для MS SQL Server	16
Настроить ConnectionStrings.config	16
Обязательные настройки ConnectionStrings.config	17
Необязательные настройки ConnectionStrings.config	18
Настроить ConnectionStrings.config для Oracle Database	20
Настроить ConnectionStrings.config	20
Обязательные настройки ConnectionStrings.config	20
Необязательные настройки ConnectionStrings.config	21
Настроить ConnectionStrings.config для PostgreSQL	22
Настроить ConnectionStrings.config	22
Обязательные настройки ConnectionStrings.config	22
Необязательные настройки ConnectionStrings.config	23
Настроить Web.config	24
Дополнительная настройка Web.config (только для Oracle)	24
Настроить сервер приложения на IIS	25
Настроить сайт приложения в IIS	25
Добавить пул приложения	27
Настроить вебсокет	33
Проверить настройки вебсокетов для Windows Server 2012 или Windows server 2016	34
Часто задаваемые вопросы по настройке вебсокетов	35
Перевести Creatio с HTTP на HTTPS	36
Настроить IIS	36
Настроить Web.config	41

Проверить обязательные компоненты Windows

ПРОДУКТЫ: [ВСЕ ПРОДУКТЫ](#)

Перед созданием и настройкой web-сайта убедитесь, что на веб-сервере установлены:

- Компоненты Windows. Обратите внимание, что обязательно нужно установить компонент Microsoft Visual C++ 2010.
- Компоненты Web Server IIS.

Для обеспечения корректной компиляции приложения необходимо скачать и установить .NET Core SDK 3.1.301 и .NET Framework SDK v 4.7.2.

Пользователю, от имени которого в IIS запущен пул приложения, необходимо предоставить права на чтение, запись и удаление файлов и вложенных каталогов для каталога \Terrasoft.WebApp\Terrasoft.Configuration.

[Скачать 64-разрядный .NET Core SDK 3.1.301](#)

[Скачать 64-разрядный .NET Framework SDK v 4.7.2](#)

Важно. Для промышленной эксплуатации приложение Creatio должно быть развернуто на семействе ОС Windows Server. ОС Windows 10 может использоваться только для разработки конфигурации.

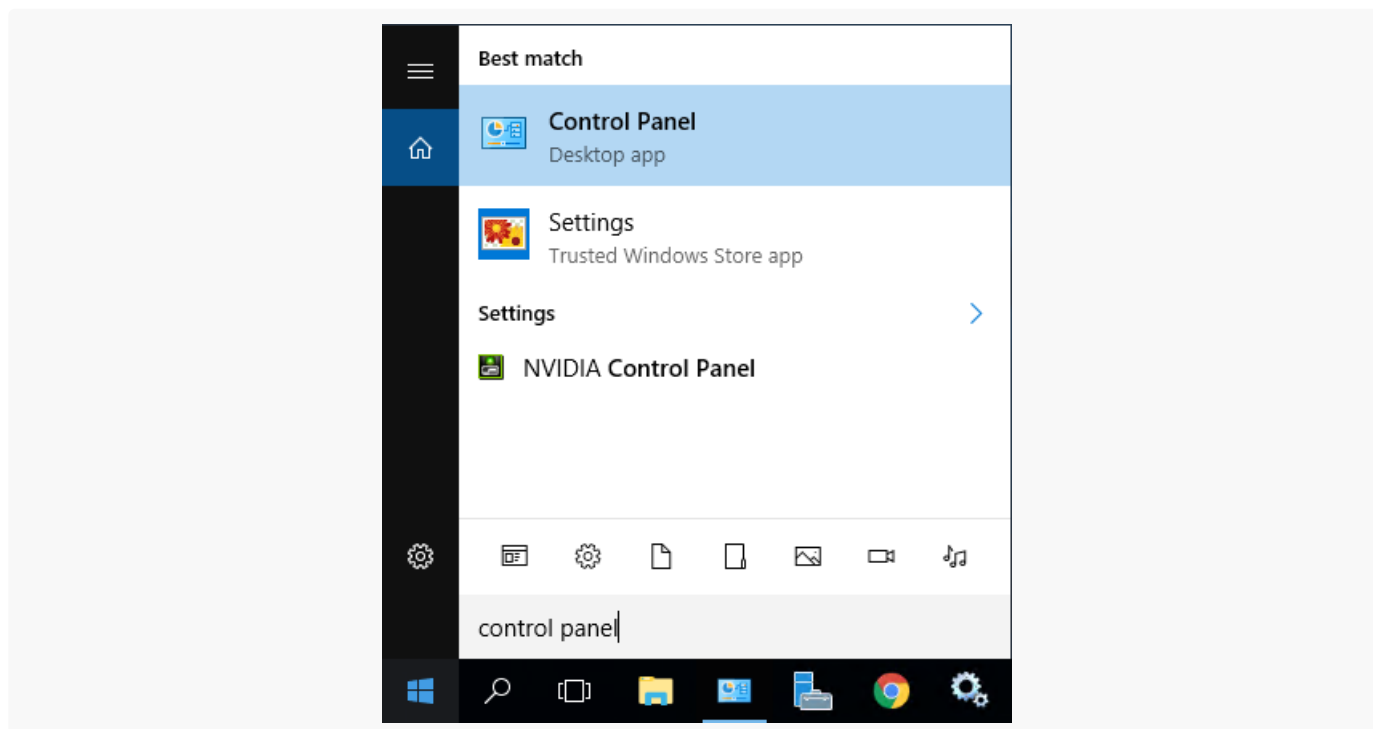
Название компонента	Элементы компонентов
Common HTTP Features	Static Content; Default Document; HTTP Errors; HTTP Redirection.
Application Development	ASP.Net; .Net Extensibility; ISAPI Extensions; ISAPI Filters. WebSocket Protocol.
Microsoft .Net Framework 3.5.1	Windows Communication Foundation HTTP Activation; Windows Communication Foundation Non-HTTP Activation.
Microsoft .Net Framework 4.7 Advanced Services и выше (Windows 8, Windows 10, Windows Server 2012, Windows Server 2016).	ASP.NET 4.6.2 или 4.7; WCF Services: HTTP Activation; Message Queuing (MSMQ) Activation; Named Pipe Activation; TCP Activation; TCP Port Sharing.
Health and Diagnostics	HTTP Logging; Logging Tools; Request Monitor; Custom Logging.
Security	Basic Authentication Request Filtering IP and Domain Restriction

Проверить обязательные компоненты Windows Server 2016

Чтобы проверить наличие необходимых компонентов:

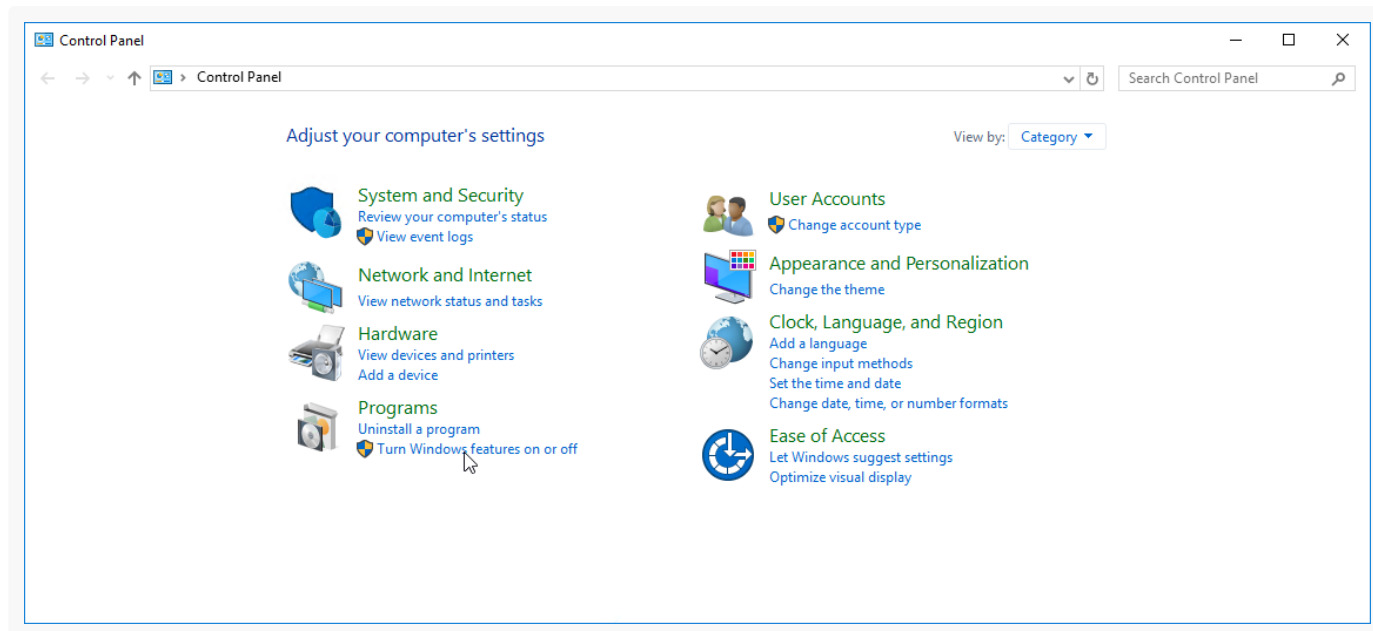
1. В поиске меню [*Start*] введите “control panel” и выберите пункт [*Control Panel*] ([Рис. 1](#)).

Рис. 1 — Раздел [*Control Panel*] меню [*Start*]



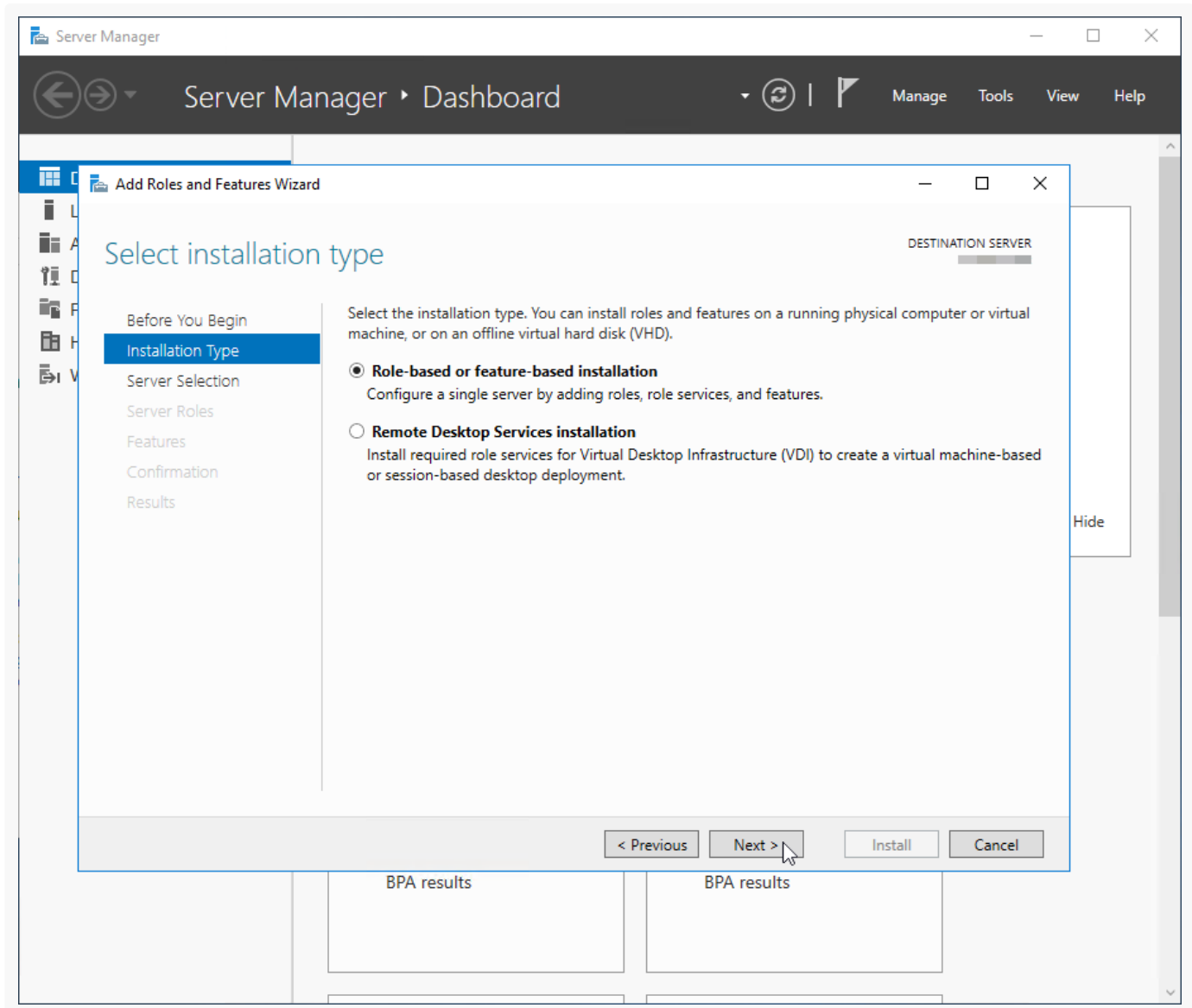
2. В открывшемся окне выберите команду [*Turn Windows features on or off*] ([Рис. 2](#)).

Рис. 2 — Команда [*Turn Windows features on or off*]



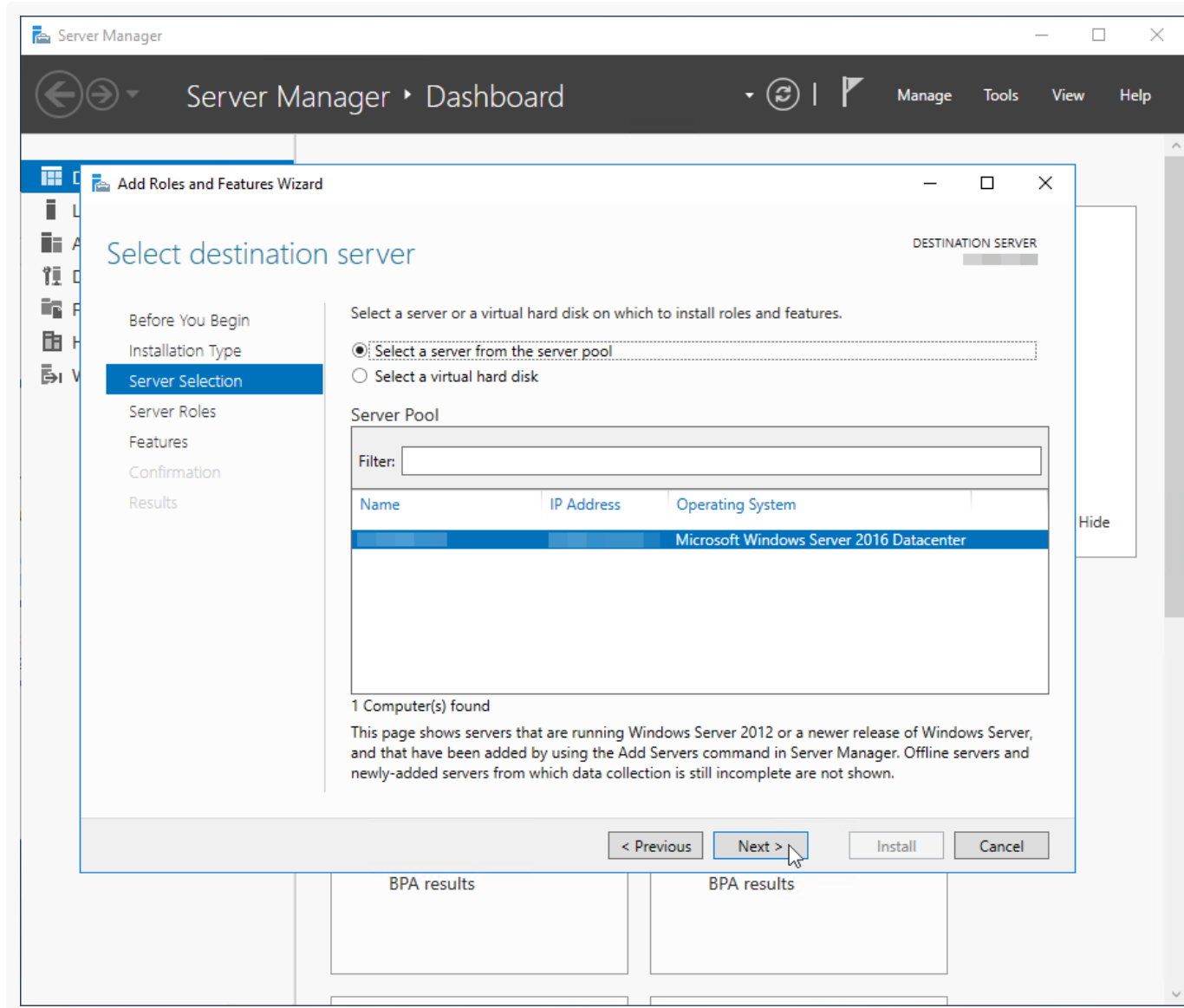
3. В окне [*Add Roles and Features Wizard*] выберите [*Role-based or feature-based installation*] и нажмите [*Next*] ([Рис. 3](#)).

Рис. 3 — Выбор установки на основе ролей



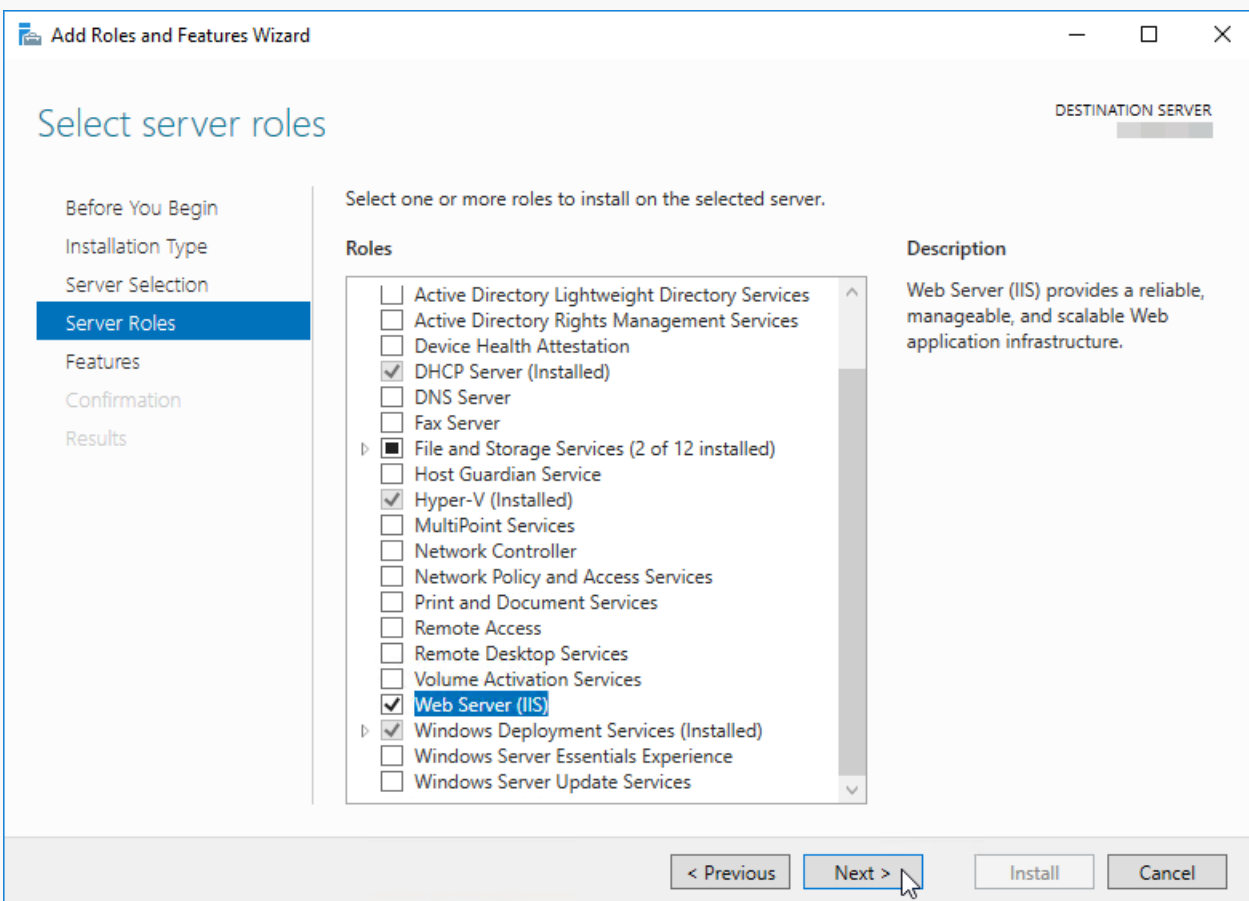
4. В области Server Pool выберите сервер, для которого необходимо установить компоненты, и нажмите [Next] ([Рис. 4](#)).

Рис. 4 — Выбор сервера



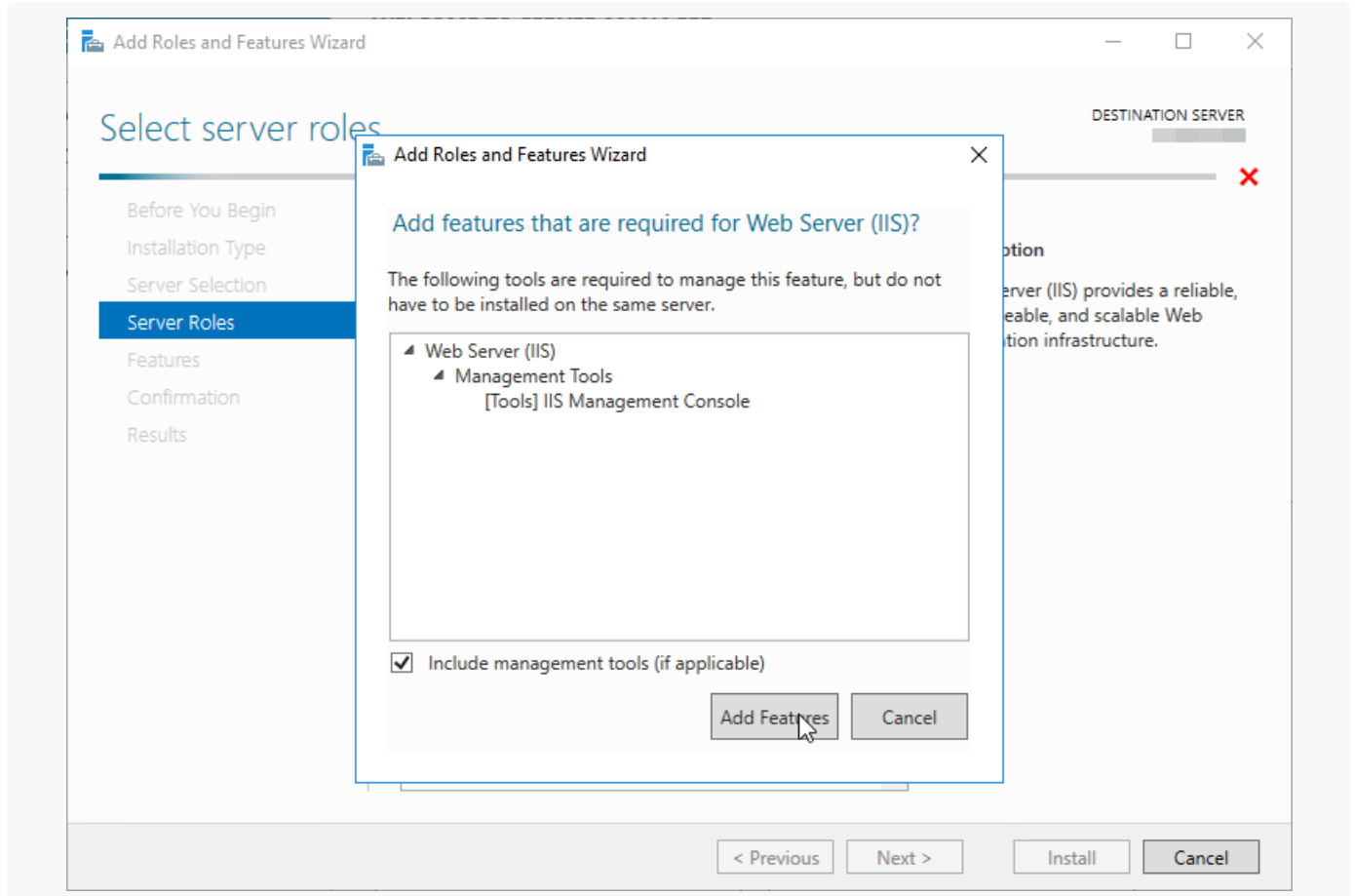
5. В списке ролей установите флажок возле значения Web Server (IIS). Нажмите [Next] ([Рис. 5](#)).

Рис. 5 — Выбор роли Web Server (IIS)



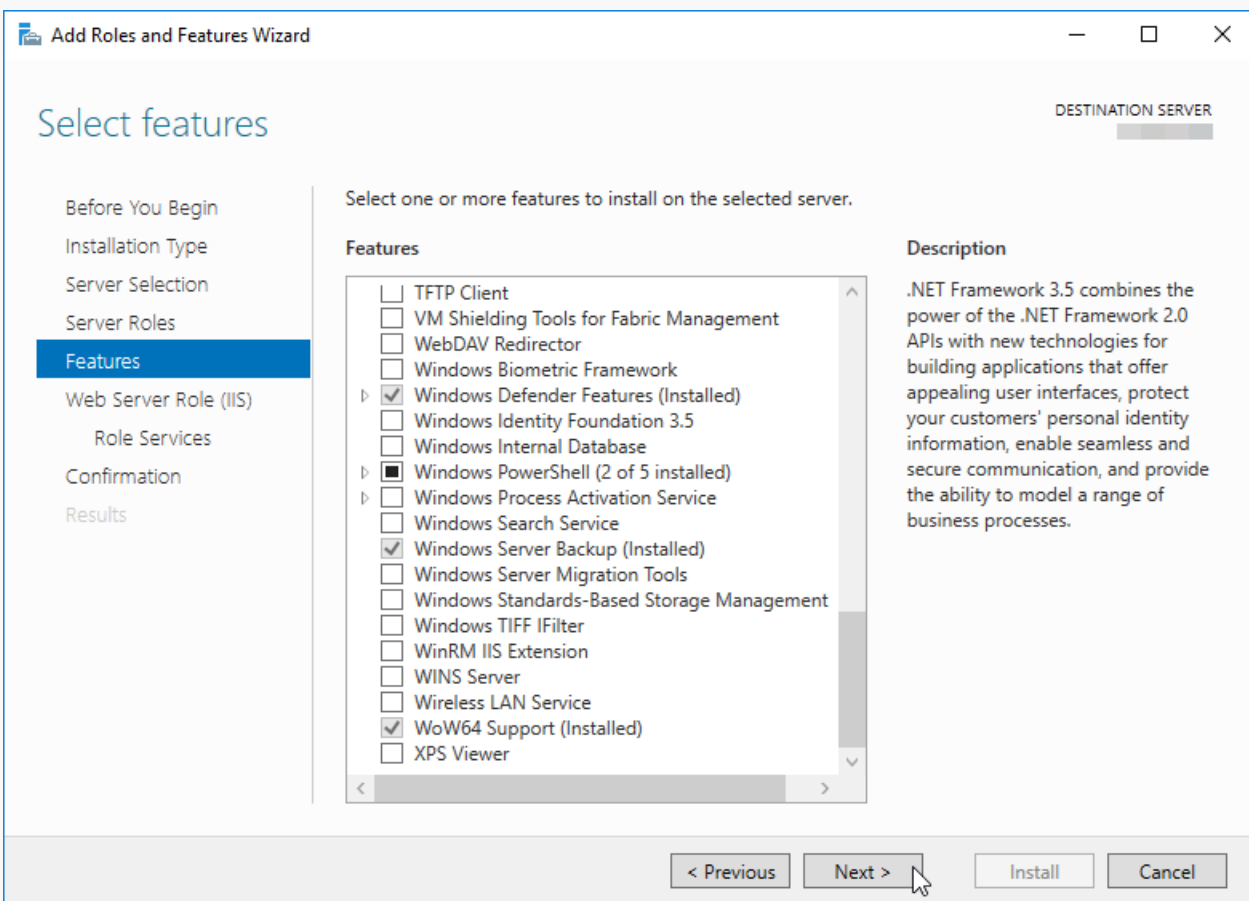
6. В открывшемся окне нажмите [*Add features*] ([Рис. 6](#)).

Рис. 6 — Подтверждение выбора функций



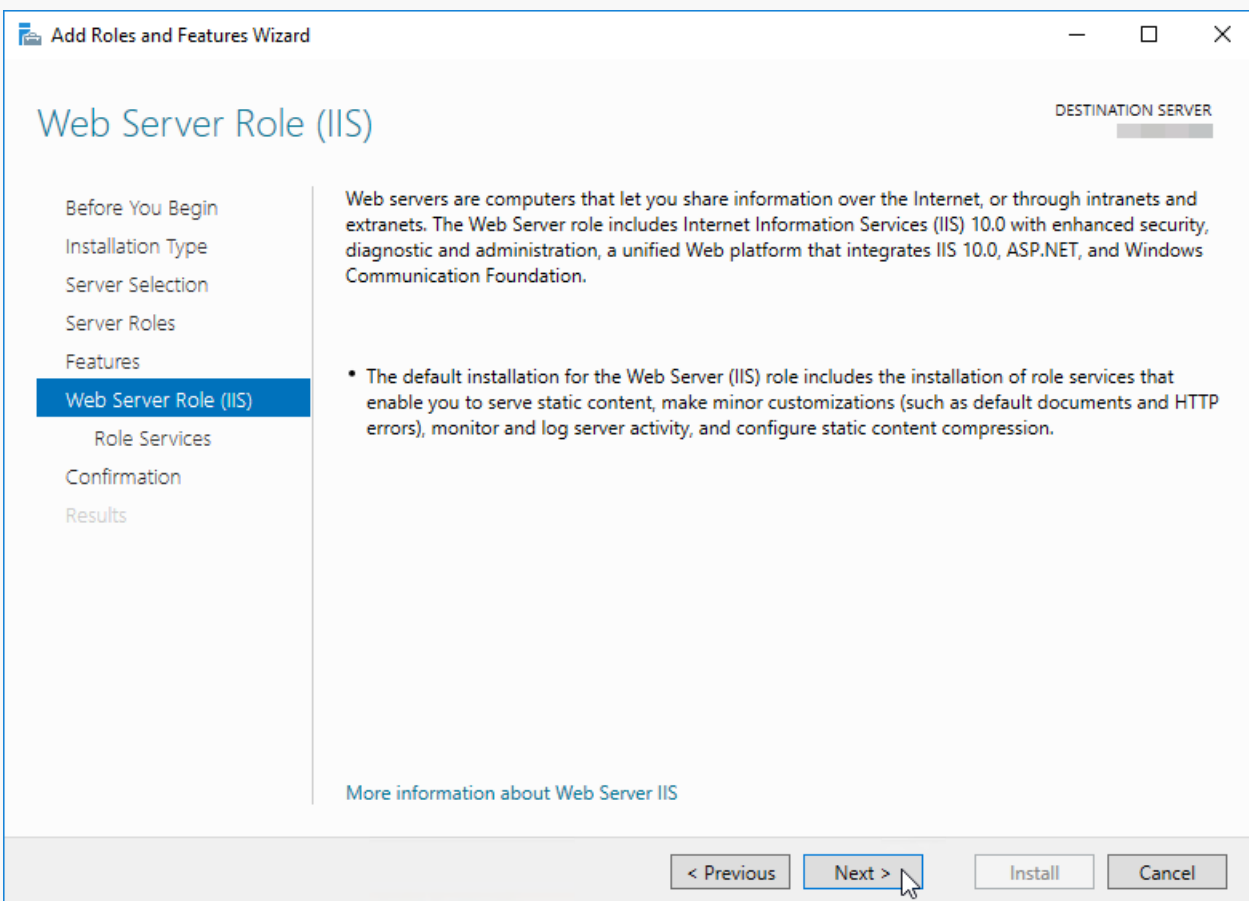
7. Выберите необходимые функции и нажмите [Next] ([Рис. 7](#)).

Рис. 7 — Выбор функций



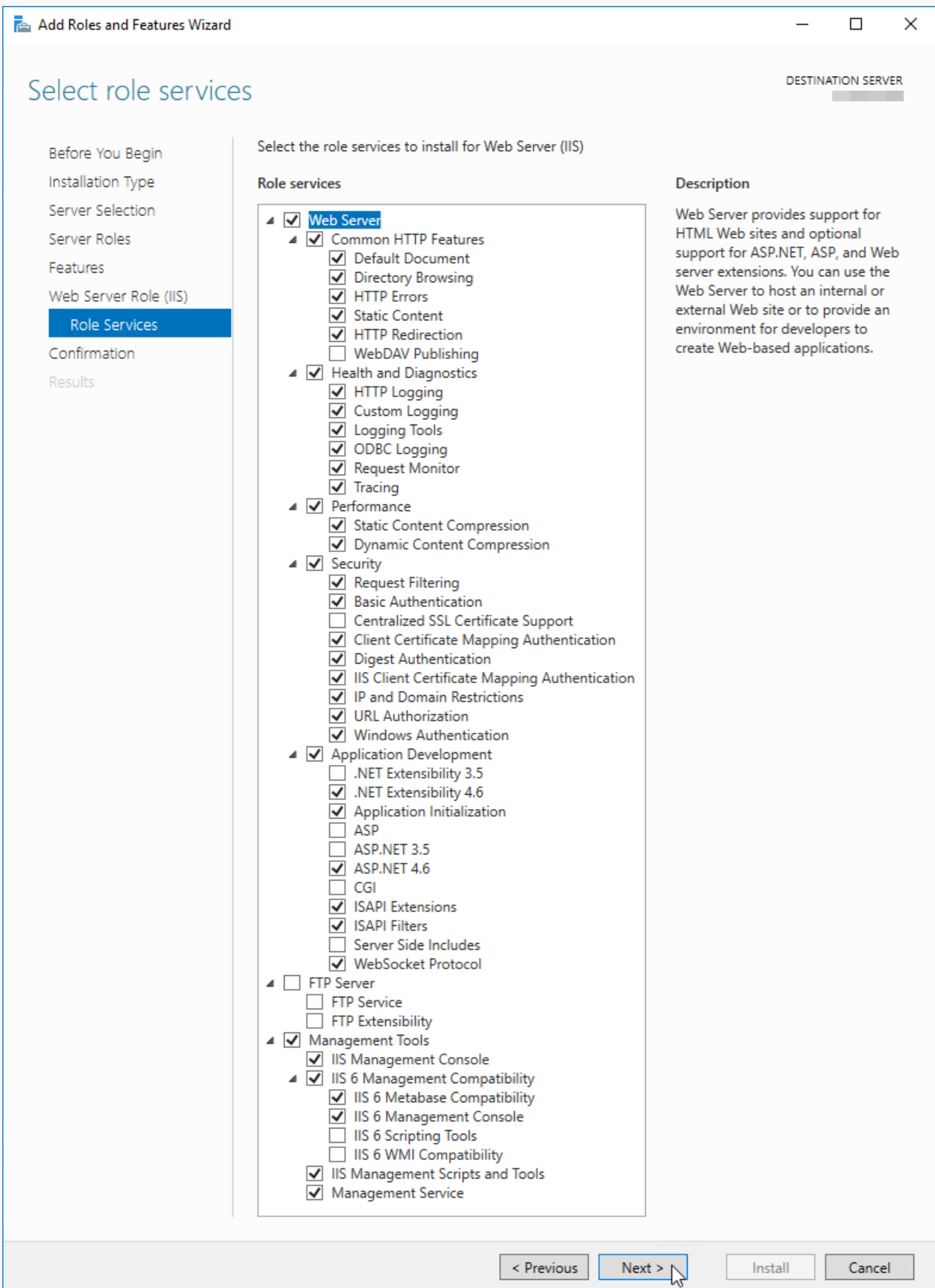
8. Нажмите [Next] для перехода к следующему шагу.

Рис. 8 — Подтверждение выбранной роли веб-сервера



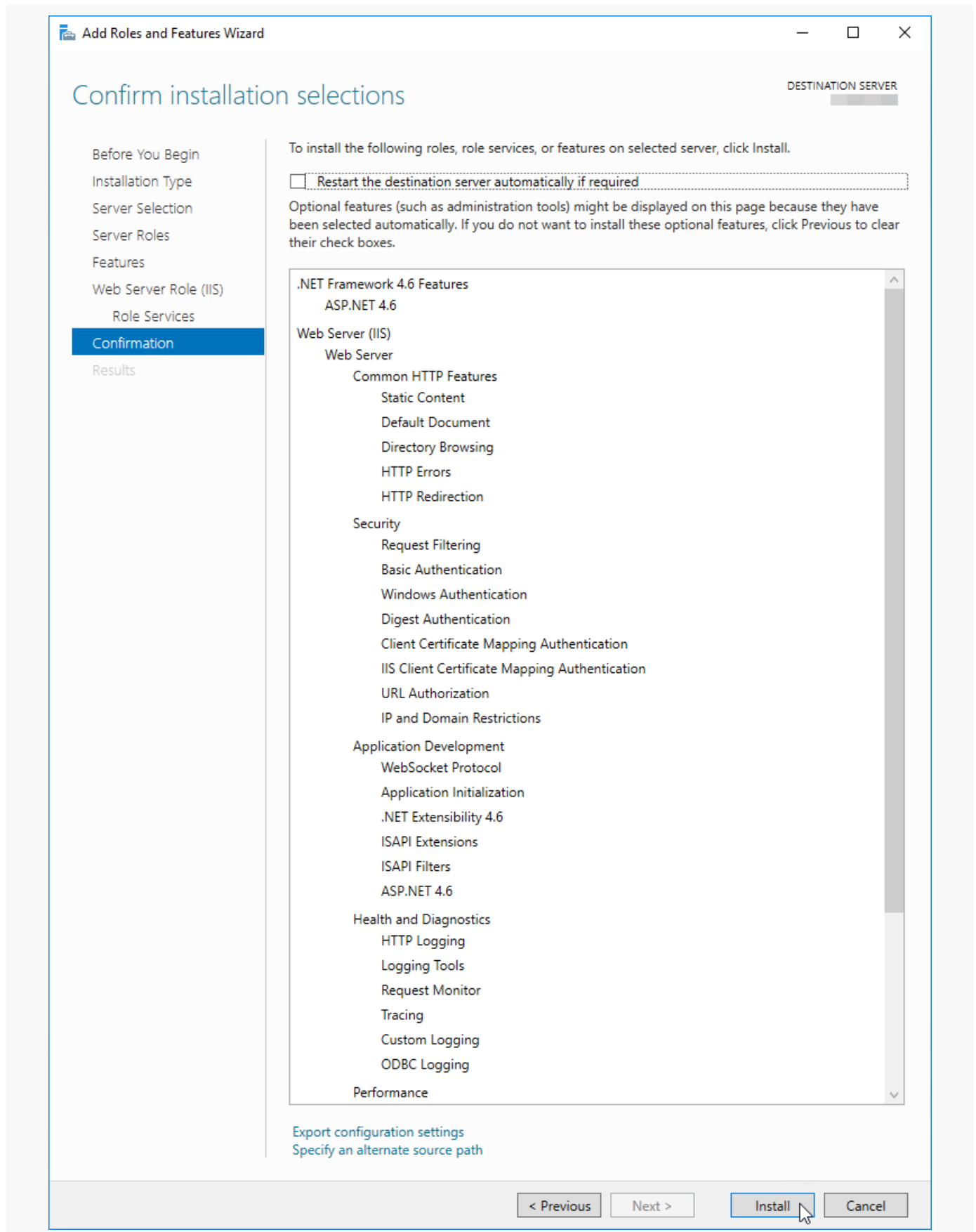
9. Убедитесь, что у вас выбраны все необходимые компоненты ([Рис. 9](#)).

Рис. 9 — Перечень необходимых компонентов Windows Server 2016



10. Нажмите [*Install*] (Рис. 10).

Рис. 10 — Подтверждение установки



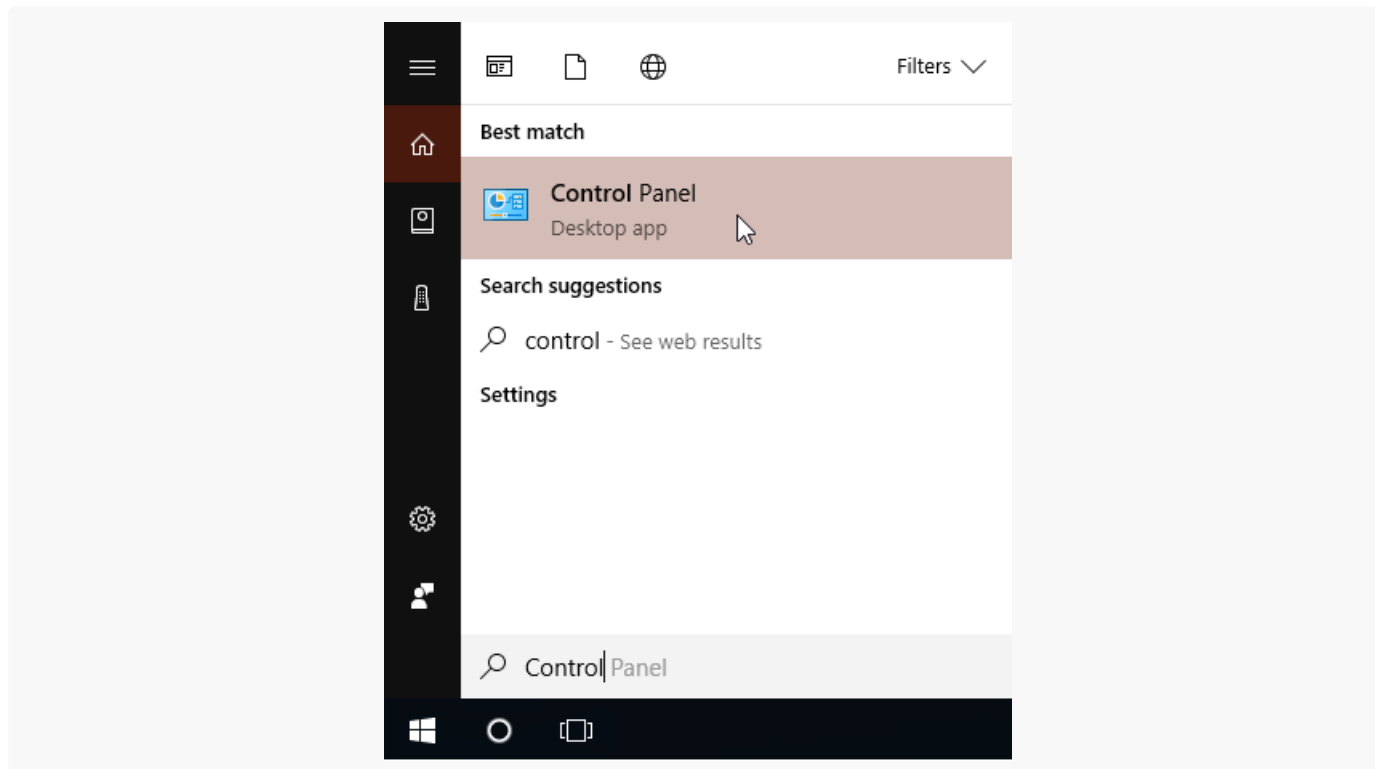
11.Перезагрузите сервер.

Проверить обязательные компоненты Windows 10

Чтобы проверить наличие необходимых компонентов:

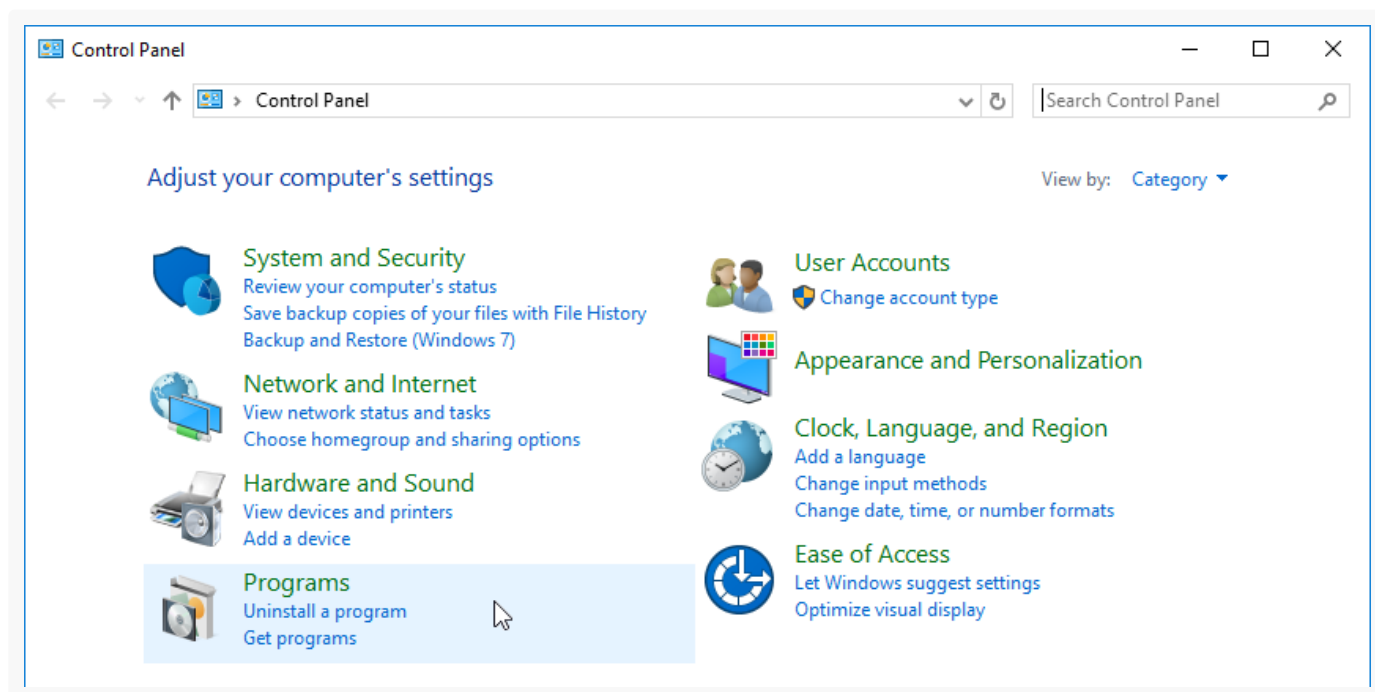
1. В поиске меню [*Start*] введите “control panel” и выберите пункт [*Control Panel*] ([Рис. 1](#)).

Рис. 1 — Раздел [*Control Panel*] меню [*Пуск*]



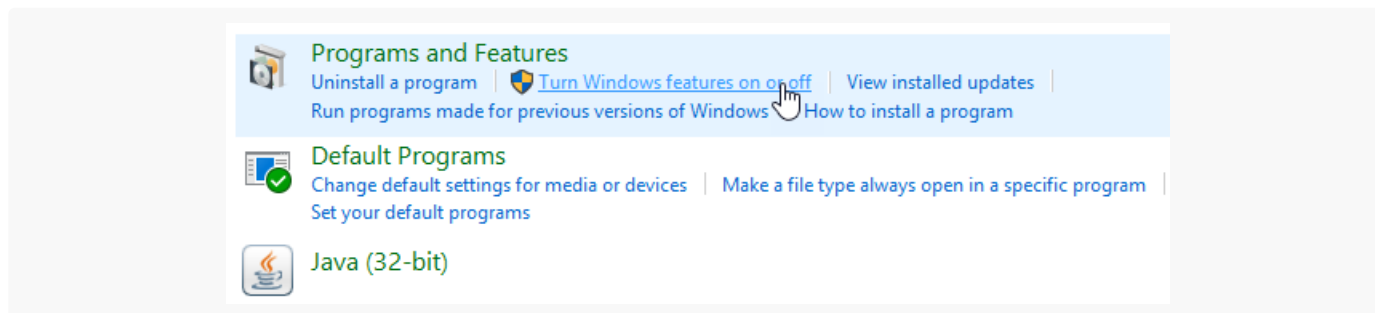
2. В открывшемся окне выберите пункт меню [*Programs*] ([Рис. 2](#)).

Рис. 2 — Меню [*Programs*] панели управления Windows



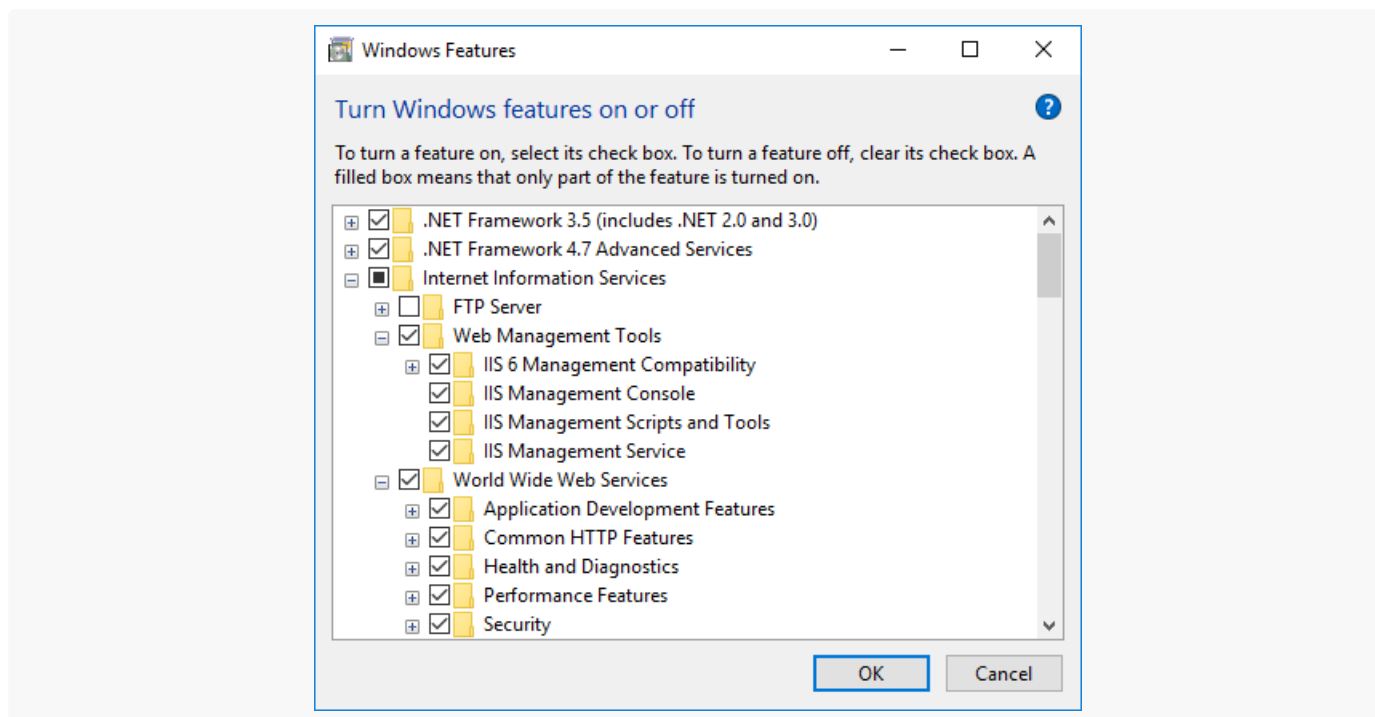
3. В разделе меню [*Programs and Features*] выберите команду [*Turn Windows features on or off*] ([Рис. 2](#)).

Рис. 3 — Команда [*Turn Windows features on or off*]



4. В окне [*Windows Features*] выберите все необходимые компоненты ([Рис. 4](#)).

Рис. 4 — Окно включения компонентов Web Server IIS и Windows



Настроить ConnectionStrings.config для MS SQL Server

ПРОДУКТЫ: [ВСЕ ПРОДУКТЫ](#)

В файле ConnectionStrings.config, который находится в корневой папке сайта Creatio, содержатся параметры подключения вашего приложения к базе данных и внешним сервисам.

Настроить ConnectionStrings.config

1. Перейдите в корневой каталог сайта Creatio ~\WebAppRoot\Creatio.
2. Откройте в текстовом редакторе файл ConnectionStrings.config.
3. Укажите параметры подключения (**connectionStrings**) вашего сайта.

Пример настроенного файла ConnectionStrings.config

```
<?xml version="1.0" encoding="utf-8"?>

<connectionStrings>

<add name="db" connectionString="Data Source=[ Имя сервера базы данных ]; Initial Catalog=[ Имя

<add name="redis" connectionString="host=[ Имя компьютера ];db=[ Номер БД Redis ];port=6379; max

<add name="defRepositoryUri" connectionString="" />

<add name="defWorkingCopyPath" connectionString="%TEMP%\%WORKSPACE%" />

<add name="defPackagesWorkingCopyPath" connectionString="%TEMP%\%APPLICATION%\%WORKSPACE%\Terras

<add name="clientUnitContentPath" connectionString="%TEMP%\%APPLICATION%\%WORKSPACE%\ClientUnits

<add name="sourceControlAuthPath" connectionString="%TEMP%\%APPLICATION%\%WORKSPACE%\Svn" />

<add name="elasticsearchCredentials" connectionString="User=[ Имя пользователя ElasticSearch ];

</connectionStrings>
```

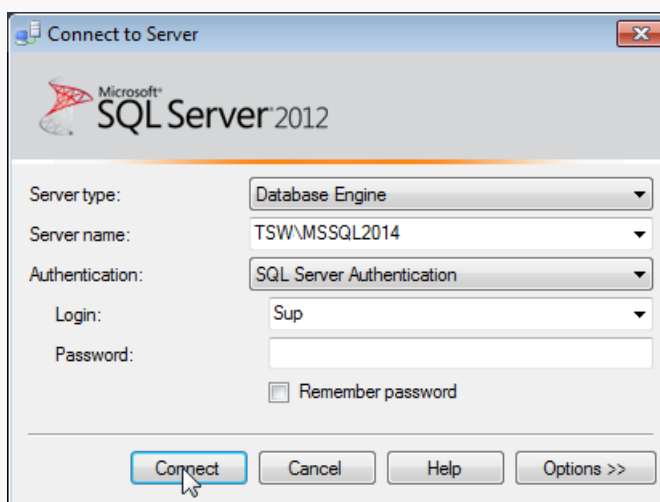
Обязательные настройки ConnectionStrings.config

Параметры подключения к базе данных и серверу кэширования необходимы для работы приложения Creatio.

- **name="db"** — отвечает за подключение к восстановленной базе данных .

Имя сервера баз данных (**Data Source**) можно увидеть в окне авторизации при подключении к серверу из Microsoft SQL Server Management Studio (Рис. 1).

Рис. 1 — Окно авторизации на SQL-сервер



Имя базы данных (Initial Catalog) должно совпадать со значением поля [*Database*], которое вы указали при восстановлении базы данных.

По умолчанию для подключения Creatio к серверу базы данных используется **аутентификация Windows** (Integrated Security), основанная на интерфейсе SSPI. Чтобы обеспечить успешное подключение к БД, необходимо для приложения указать пользователя Windows, под которым будет выполняться подключение к серверу базы данных.

```
<add name="db" connectionString="Data Source=[ Имя сервера базы данных ];
Initial Catalog=[ Имя базы данных ];
Persist Security Info=True; MultipleActiveResultSets=True;
Integrated Security=SSPI; Pooling = true; Max Pool Size = 100; Async = true" />
```

Если же вы хотите авторизоваться на сервере базы данных **при помощи логина и пароля пользователя MS SQL**, то необходимо предварительно создать логин и пароль такого пользователя на сервере MS SQL и внести изменения в файл ConnectionStrings.config. В строке подключения к базе данных (add name="db") замените переменную **Integrated Security=SSPI** на **User ID** и **Password**:

```
<add name="db" connectionString="Data Source=TSW\MSSQL2014;
Initial Catalog=7.10.2.1416_SalesEnterprise_Demo;
Persist Security Info=True; MultipleActiveResultSets=True;
User ID=Sup; Password=password; Pooling = true; Max Pool Size = 100; Async = true" />
```

- **name="redis"** — отвечает за взаимодействие с сервером Redis.

```
<add name="redis" connectionString="host=[ Имя компьютера ];db=[ Номер БД Redis ];port=6379;
maxReadPoolSize=10;maxWritePoolSize=500" />
```

Необязательные настройки ConnectionStrings.config

Параметры подключения к внешним сервисам опциональны, они заполняются только в том случае, когда необходимо, например, при подключении системы контроля версий.

- **tempDirectoryPath** — путь к временной директории, используемой механизмом установки пакетов:

```
<add name="tempDirectoryPath" connectionString="[ Путь к временной директории, используемой механизмом установки пакетов ]";
```

- **defPackagesWorkingCopyPath** — путь к рабочей копии пользовательских пакетов приложения (только при использовании системы контроля версий SVN). Содержимое рабочей копии представляет собой пользовательские пакеты в виде набора папок и файлов. Оно синхронизируется с репозиторием системы контроля версий SVN при помощи встроенного в Creatio клиента SVN. Данный параметр настраивается при подключении системы контроля версий и используется только в режиме разработки по умолчанию, без включения режима разработки в файловой системе. Значение по умолчанию указывает на временный каталог, который может быть очищен операционной системой. Рекомендуется установить значение, указывающее на пользовательский каталог, так как использование каталогов приложения, например, .\Terrasoft.WebApp\Terrasoft.Configuration\Pkg, может привести к ошибкам компиляции.

```
<add name="defPackagesWorkingCopyPath" connectionString="[ Путь к рабочей копии пользовательских пакетов ]";
```

- **sourceControlAuthPath** — путь к данным авторизации для встроенного клиента системы контроля версий SVN (если используется). Значение по умолчанию указывает на временный каталог, который может быть очищен операционной системой. При использовании системы контроля версий рекомендуем указать в данном параметре путь к фиксированному каталогу.

```
<add name="sourceControlAuthPath" connectionString="[ Путь к данным авторизации системы контроля версий ]";
```

- **influx** — отвечает за взаимодействие с сервисом сбора метрик сайта. Данный параметр заполняется только в том случае, когда необходимо собрать статистику использования функциональности для последующей отладки.

```
<add name="influx" connectionString="url=[ Адрес сервиса сбора метрик сайта ]; user=[ Пользователь ]; password=[ Пароль ]";
```

- **clientPerformanceLoggerServiceUri** — отвечает за взаимодействие с сервисом логгирования. Данный параметр заполняется только в том случае, когда необходимо собрать информацию о загрузке страниц приложения.

```
<add name="clientPerformanceLoggerServiceUri" connectionString="[ Адрес сервиса логгирования ]";
```

- **messageBroker** — отвечает за взаимодействие с сервисом RabbitMQ. Данный параметр заполняется только в том случае, когда необходимо настроить горизонтальное масштабирование нагрузки с помощью RabbitMQ.

```
<add name="messageBroker" connectionString="amqp://[ Имя пользователя MessageBroker ]:[ Парол
```

Настроить ConnectionStrings.config для Oracle Database

ПРОДУКТЫ: [ВСЕ ПРОДУКТЫ](#)

В файле ConnectionStrings.config, который находится в корневой папке сайта Creatio, содержатся параметры подключения вашего приложения к базе данных и внешним сервисам.

Настроить ConnectionStrings.config

1. Перейдите в корневой каталог сайта Creatio **~\WebAppRoot\Creatio**.
2. Откройте в текстовом редакторе файл ConnectionStrings.config.
3. Укажите параметры подключения (**connectionStrings**).

Пример настроенного файла ConnectionStrings.config

```
<?xml version="1.0" encoding="utf-8"?>
<connectionStrings>
  <add name="db" connectionString="Data Source=(DESCRIPTION =
  (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(HOST = [ Database server name ])(PORT = 1521))) (CC
  <add name="redis" connectionString="host=[ Имя компьютера ];db=[ Номер БД Redis ];port=6379;
maxReadPoolSize=10;maxWritePoolSize=500" />
  <add name="defRepositoryUri" connectionString="" />
  <add name="defWorkingCopyPath" connectionString="%TEMP%\%WORKSPACE%" />
  <add name="defPackagesWorkingCopyPath"
connectionString="%TEMP%\%APPLICATION%\%WORKSPACE%\TerrasoftPackages" />
  <add name="clientUnitContentPath"
connectionString="%TEMP%\%APPLICATION%\%WORKSPACE%\ClientUnitSrc" />
  <add name="sourceControlAuthPath"
connectionString="%TEMP%\%APPLICATION%\%WORKSPACE%\Svn" />
</connectionStrings>
```

Обязательные настройки ConnectionStrings.config

Параметры подключения к базе данных и серверу кэширования необходимы для работы приложения Creatio.

- **name="db"** — отвечает за подключение к восстановленной базе данных, где
 - [Database server name] — сетевой адрес сервера базы данных.

- [*Oracle service name*] — имя сервиса.
- [*Schema name*] — имя схемы восстановленной базы данных.
- [*Schema password*] — пароль к схеме восстановленной базы данных.

```
<add name="db" connectionString="Data Source=(DESCRIPTION =
(ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(HOST = [ Database server name ])(PORT = 1521)))
```

- **name="redis"** — отвечает за взаимодействие с сервером Redis.

```
<add name="redis" connectionString="host=[ Имя компьютера ];db=[ Номер БД Redis ];port=6379
maxReadPoolSize=10;maxWritePoolSize=500" />
```

Необязательные настройки ConnectionStrings.config

Параметры подключения к внешним сервисам опциональны, они заполняются только в том случае, когда необходимо, например, при подключении системы контроля версий.

- **tempDirectoryPath** — путь к временной директории, используемой механизмом установки пакетов:

```
<add name="tempDirectoryPath" connectionString=[ Путь к временной директории, используемой ме
```

- **defPackagesWorkingCopyPath** — путь к рабочей копии пользовательских пакетов приложения (только при использовании системы контроля версий SVN). Содержимое рабочей копии представляет собой пользовательские пакеты в виде набора папок и файлов. Оно синхронизируется с репозиторием системы контроля версий SVN при помощи встроенного в Creatio клиента SVN. Данный параметр настраивается при подключении системы контроля версий и используется только в режиме разработки по умолчанию, без включения режима разработки в файловой системе. Значение по умолчанию указывает на временный каталог, который может быть очищен операционной системой. Рекомендуется установить значение, указывающее на пользовательский каталог, так как использование каталогов приложения, например, .\Terrasoft.WebApp\Terrasoft.Configuration\Pkg, может привести к ошибкам компиляции.

```
<add name="defPackagesWorkingCopyPath" connectionString=[ Путь к рабочей копии пользовательск
```

- **sourceControlAuthPath** — путь к данным авторизации для встроенного клиента системы контроля версий SVN (если используется). Значение по умолчанию указывает на временный каталог, который может быть очищен операционной системой. При использовании системы контроля версий рекомендуем указать в данном параметре путь к фиксированному каталогу.

```
<add name="sourceControlAuthPath" connectionString=[ Путь к данным авторизации системы хранен
```

Настроить ConnectionStrings.config для PostgreSQL

ПРОДУКТЫ: [ВСЕ ПРОДУКТЫ](#)

В файле ConnectionStrings.config, который находится в корневой папке сайта Creatio, содержатся параметры подключения вашего приложения к базе данных и внешним сервисам.

Настроить ConnectionStrings.config

1. Перейдите в корневой каталог сайта Creatio ~\WebAppRoot\Creatio.
2. Откройте в текстовом редакторе файл ConnectionStrings.config.
3. Укажите параметры подключения (**connectionStrings**).

Пример настроенного файла ConnectionStrings.config

```
<?xml version="1.0" encoding="utf-8"?>
<connectionStrings>
  <add name="db" connectionString="Server=[ Имя сервера базы данных ];Port=[ Порт сервера базы да
  <add name="redis" connectionString="host=[ Имя компьютера ];db=[ Номер БД Redis ];port=6379;maxB
  <add name="redisSentinel" connectionString="sentinelHosts=localhost:26380,localhost:26381,localh
  <add name="defPackagesWorkingCopyPath" connectionString="%TEMP%\%APPLICATION%\%APPPPOOLIDENTITY%\
  <add name="tempDirectoryPath" connectionString="%TEMP%\%APPLICATION%\%APPPPOOLIDENTITY%\%WORKSPAC
  <add name="sourceControlAuthPath" connectionString="%TEMP%\%APPLICATION%\%APPPPOOLIDENTITY%\%WORK
  <add name="elasticsearchCredentials" connectionString="User=[ Имя пользователя Elasticsearch ];
  <add name="influx" connectionString="url=http://10.0.7.161:30359; user=; password=; batchInterva
</connectionStrings>
```

Обязательные настройки ConnectionStrings.config

Параметры подключения к базе данных и серверу кэширования необходимы для работы приложения Creatio.

1. К восстановленной базе данных (**name="db"**):

```
<add name="db" connectionString="Server=[ Имя сервера базы данных ];Port=[ Порт сервера базы
```

2. К Redis Server (**name="redis"**):

```
<add name="redis" connectionString="host=[ Имя компьютера ];db=[ Номер БД Redis ];port=6379;m
```

Необязательные настройки ConnectionStrings.config

Параметры подключения к внешним сервисам опциональны, они заполняются только в том случае, когда необходимо, например, при подключении системы контроля версий.

- **tempDirectoryPath** — путь к временной директории, используемой механизмом установки пакетов:

```
<add name="tempDirectoryPath" connectionString="[ Путь к временной директории, используемой механизмом установки пакетов ]";
```

- **defPackagesWorkingCopyPath** — путь к рабочей копии пользовательских пакетов приложения (только при использовании системы контроля версий SVN). Содержимое рабочей копии представляет собой пользовательские пакеты в виде набора папок и файлов. Оно синхронизируется с репозиторием системы контроля версий SVN при помощи встроенного в Creatio клиента SVN. Данный параметр настраивается при подключении системы контроля версий и используется только в режиме разработки по умолчанию, без включения режима разработки в файловой системе. Значение по умолчанию указывает на временный каталог, который может быть очищен операционной системой. Рекомендуется установить значение, указывающее на пользовательский каталог, так как использование каталогов приложения, например, `.\Terrasoft.WebApp\Terrasoft.Configuration\Pkg`, может привести к ошибкам компиляции.

```
<add name="defPackagesWorkingCopyPath" connectionString="[ Путь к рабочей копии пользовательских пакетов ]";
```

- **sourceControlAuthPath** — путь к данным авторизации для встроенного клиента системы контроля версий SVN (если используется). Значение по умолчанию указывает на временный каталог, который может быть очищен операционной системой. При использовании системы контроля версий рекомендуем указать в данном параметре путь к фиксированному каталогу.

```
<add name="sourceControlAuthPath" connectionString="[ Путь к данным авторизации системы контроля версий SVN ]";
```

- **influx** — отвечает за взаимодействие с сервисом сбора метрик сайта. Данный параметр заполняется только в том случае, когда необходимо собрать статистику использования функциональности для последующей отладки.

```
<add name="influx" connectionString="url=[ Адрес сервиса сбора метрик сайта ]; user=[ Пользователь ]; password=[ Пароль ]";
```

- **clientPerformanceLoggerServiceUri** — отвечает за взаимодействие с сервисом логгирования. Данный параметр заполняется только в том случае, когда необходимо собрать информацию о загрузке страниц приложения.

```
<add name="clientPerformanceLoggerServiceUri" connectionString="[ Адрес сервиса логгирования ]";
```

- **messageBroker** — отвечает за взаимодействие с сервисом RabbitMQ. Данный параметр заполняется только в том случае, когда необходимо настроить горизонтальное масштабирование нагрузки с помощью RabbitMQ.

```
<add name="messageBroker" connectionString="amqp://[ Имя пользователя MessageBroker ]:[ Парол
```

Настроить Web.config

ПРОДУКТЫ: [ВСЕ ПРОДУКТЫ](#)

Необходимо сгенерировать для вашего приложения уникальное значение machineKey.

1. Скачайте powershell-скрипт. [Скачать скрипт](#).
2. Запустите powershell-терминал от имени администратора.
3. В powershell-терминале укажите путь к корневой папке приложения и выполните скрипт. Пример запуска скрипта:

```
.\UpdateMachineKey.ps1 "путь\к\корневой\папке\Creatio"
```

В результате в файлах Web.config, которые находятся в корневой папке приложения и в папке Terrasoft.WebApp, будет сгенерировано уникальное значение machineKey.

Дополнительная настройка Web.config (только для Oracle)

После настройки параметров соединения с базой данных в файле ConnectionStrings.config для Oracle Database также необходимо настроить конфигурационный файл Web.config, в который вынесены некоторые параметры конфигурации приложения, необходимые для его работы.

Чтобы настроить Web.config:

1. Перейдите в корневую папку приложения **~\WebAppRoot\Creatio**.
2. Откройте файл файл Web.config в текстовом редакторе.
3. В параметре general блока <db> измените значение атрибута currentSchemaName на имя схемы, которая указана в ConnectionStrings.config. Это позволит сайту приложения увидеть записанные в файле параметры конфигурации.

```
<db>
  <general securityEngineType="Terrasoft.DB.Oracle.OracleSecurityEngine, Terrasoft.DB.Oracle"
  executorType="Terrasoft.DB.Oracle.OracleExecutor, Terrasoft.DB.Oracle"
  isCaseInsensitive="true" maxAnsiJoinCount="0"
  engineType="Terrasoft.DB.Oracle.OracleEngine, Terrasoft.DB.Oracle"
  metaEngineType="Terrasoft.DB.Oracle.OracleMetaEngine,
  Terrasoft.DB.Oracle"
  metaScriptType="Terrasoft.DB.Oracle.OracleMetaScript,
  Terrasoft.DB.Oracle" typeConverterType="Terrasoft.DB.Oracle.OracleTypeConverter,
  Terrasoft.DB.Oracle" connectionStringName="db"
  binaryPackageSize="1048576" useOrderNullsPosition="true"
```



```
currentSchemaName="[ Имя схемы базы данных Oracle ]" maxEntitySchemaNameLength="30" />
</db>
```

4. Проверьте, что в блоке <DbProviderFactories> указана версия библиотеки 4.112.1.2. Этот параметр определяет коннектор, через который происходит подключение сайта к базе данных.

```
<system.data>
  <DbProviderFactories>
    <remove invariant="Oracle.DataAccess.Client" />
    <add name="Oracle Data Provider for .NET"
invariant="Oracle.DataAccess.Client" description="Oracle Data
Provider for .NET" type="Oracle.DataAccess.Client.OracleClientFactory, Oracle.DataAccess,
Version=4.112.1.2, Culture=neutral, PublicKeyToken=89b483f429c47342"/>
  </DbProviderFactories>
</system.data>
```

5. В блоке <quartz> укажите значение ключа **quartz.dataSource.SchedulerDb.provider**. Этот параметр необходим для того, чтобы планировщик мог взаимодействовать с таблицами в базе данных.

```
<add key="quartz.dataSource.SchedulerDb.provider" value="OracleODP-1123-40" />
```

6. Сохраните изменения.

Настроить сервер приложения на IIS

ПРОДУКТЫ: [ВСЕ ПРОДУКТЫ](#)

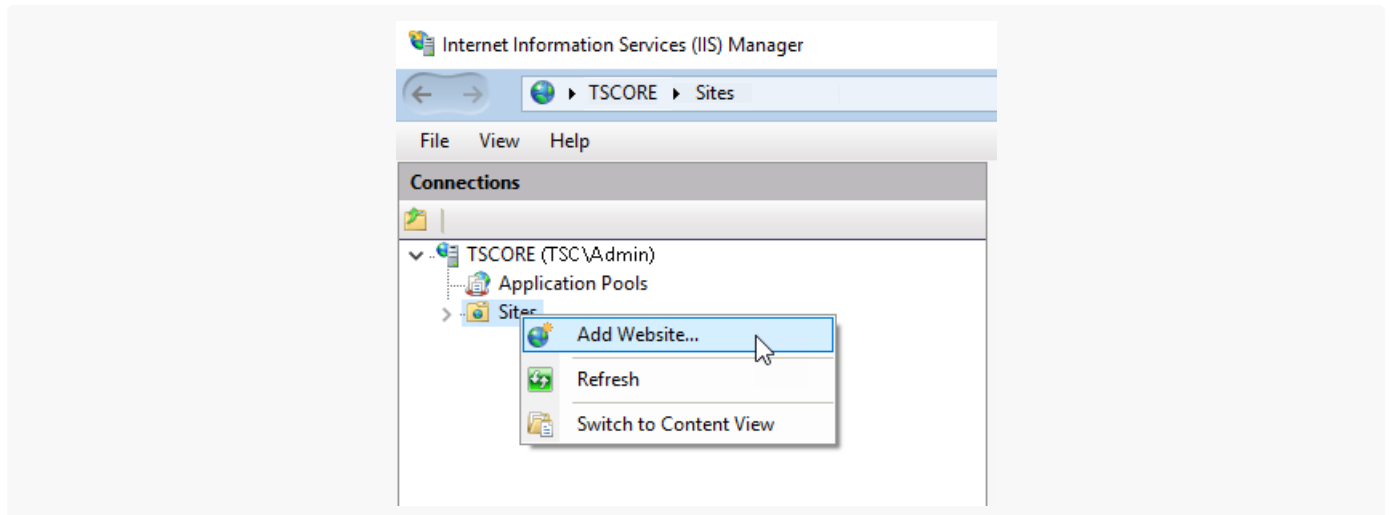
Настройка сервера приложений Creatio (веб-сервера) на IIS включает в себя настройку веб-сайта приложения в IIS и добавление пула приложения.

Настроить сайт приложения в IIS

Для создания и настройки web-сайта:

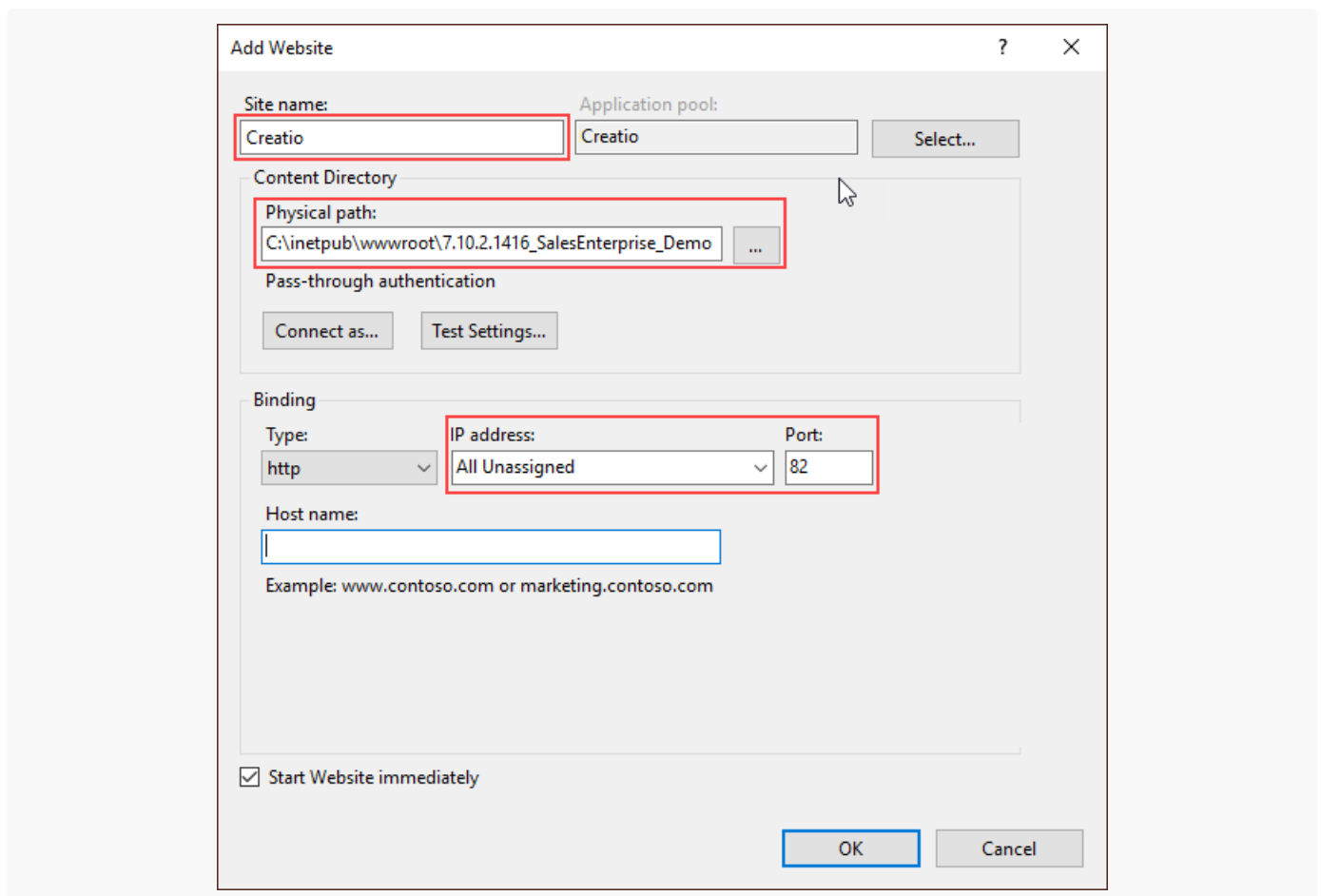
1. В окне управления IIS щелкните правой клавишей мыши по [*Sites*] и выберите [*Add Website*] в контекстном меню (Рис. 1).

Рис. 1 — Создание сайта для приложения Creatio



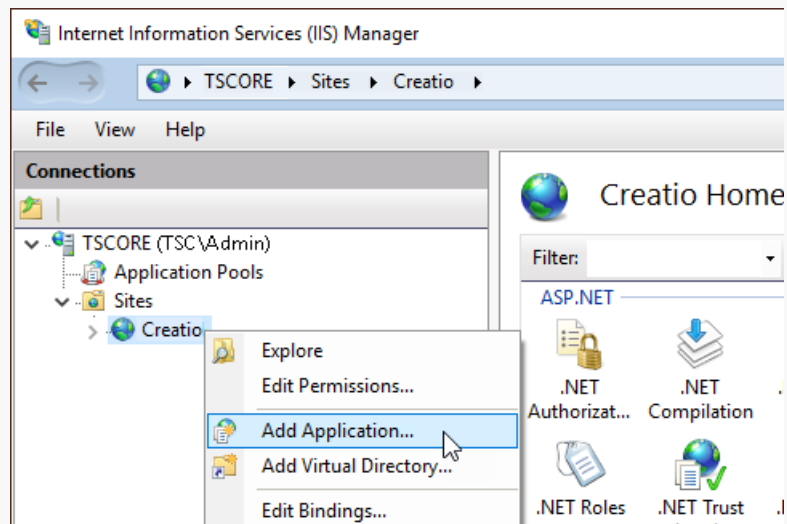
- Укажите название сайта, путь к корневому каталогу с файлами Creatio, используемые IP-адреса и порт сайта (Рис. 2). Путем по умолчанию для web-сайтов является путь C:\inetpub\wwwroot. В случае необходимости можно указать свой адрес.

Рис. 2 — Окно параметров нового сайта



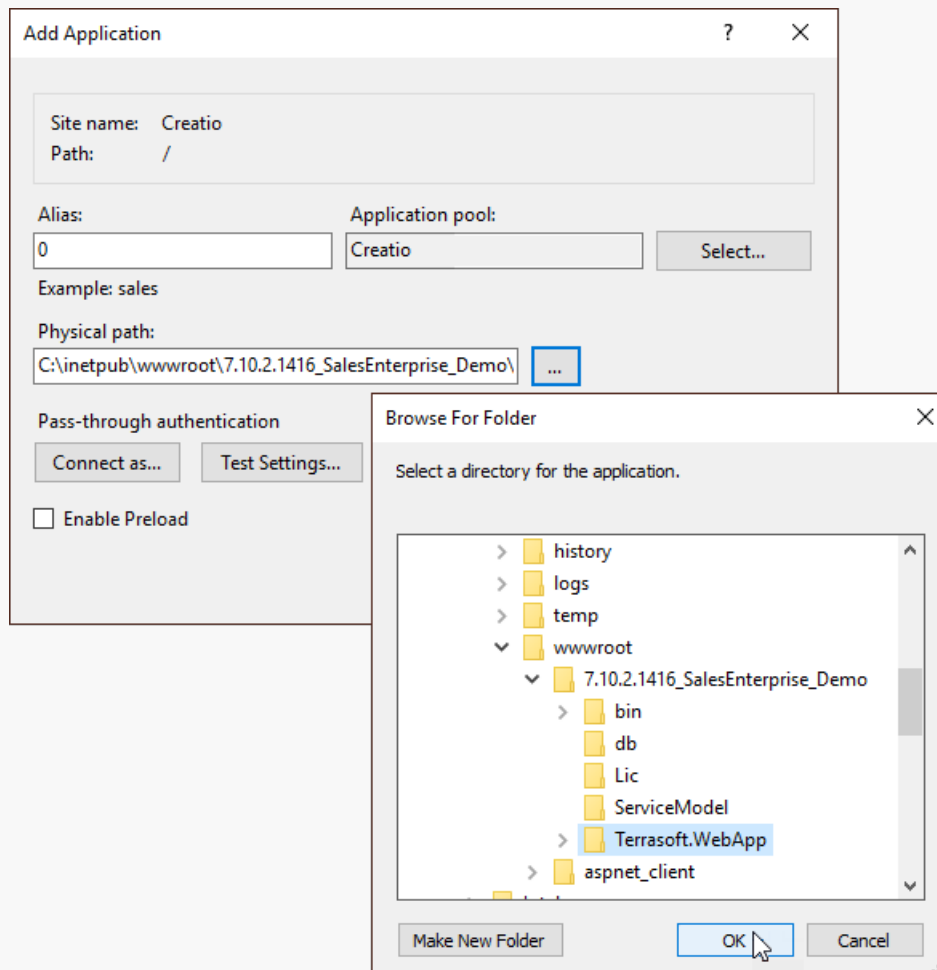
- В области [*Connections*] щелкните правой клавишей мыши по созданному сайту и выберите команду [*Add Application*] контекстного меню (Рис. 3).

Рис. 3 — Выбор команды добавления приложения в IIS



4. Введите имя приложения “0” в поле [*Alias*]. Укажите директорию “Terrasoft.WebApp” (Рис. 4).

Рис. 4 — Окно выбора параметров приложения

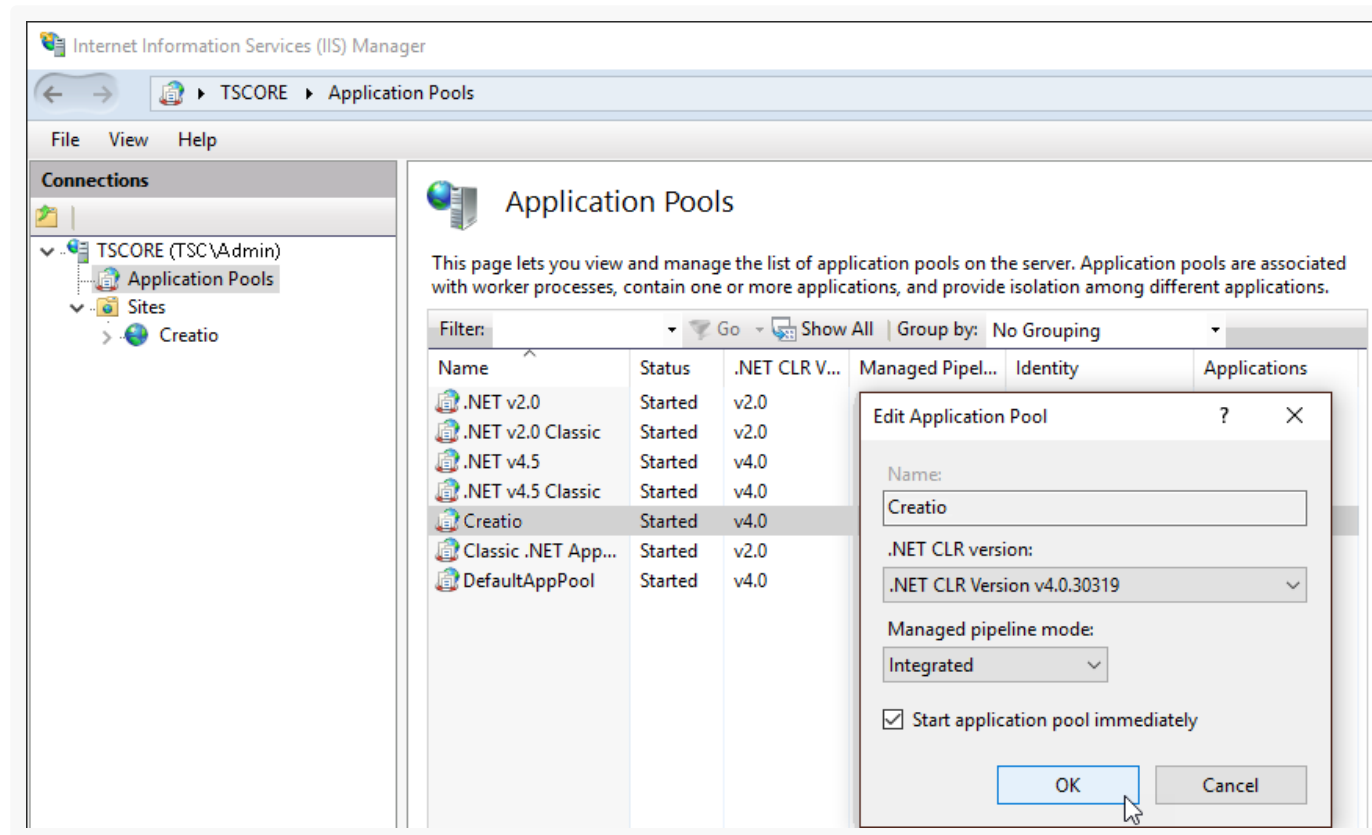


Добавить пул приложения

Для добавления пула приложения:

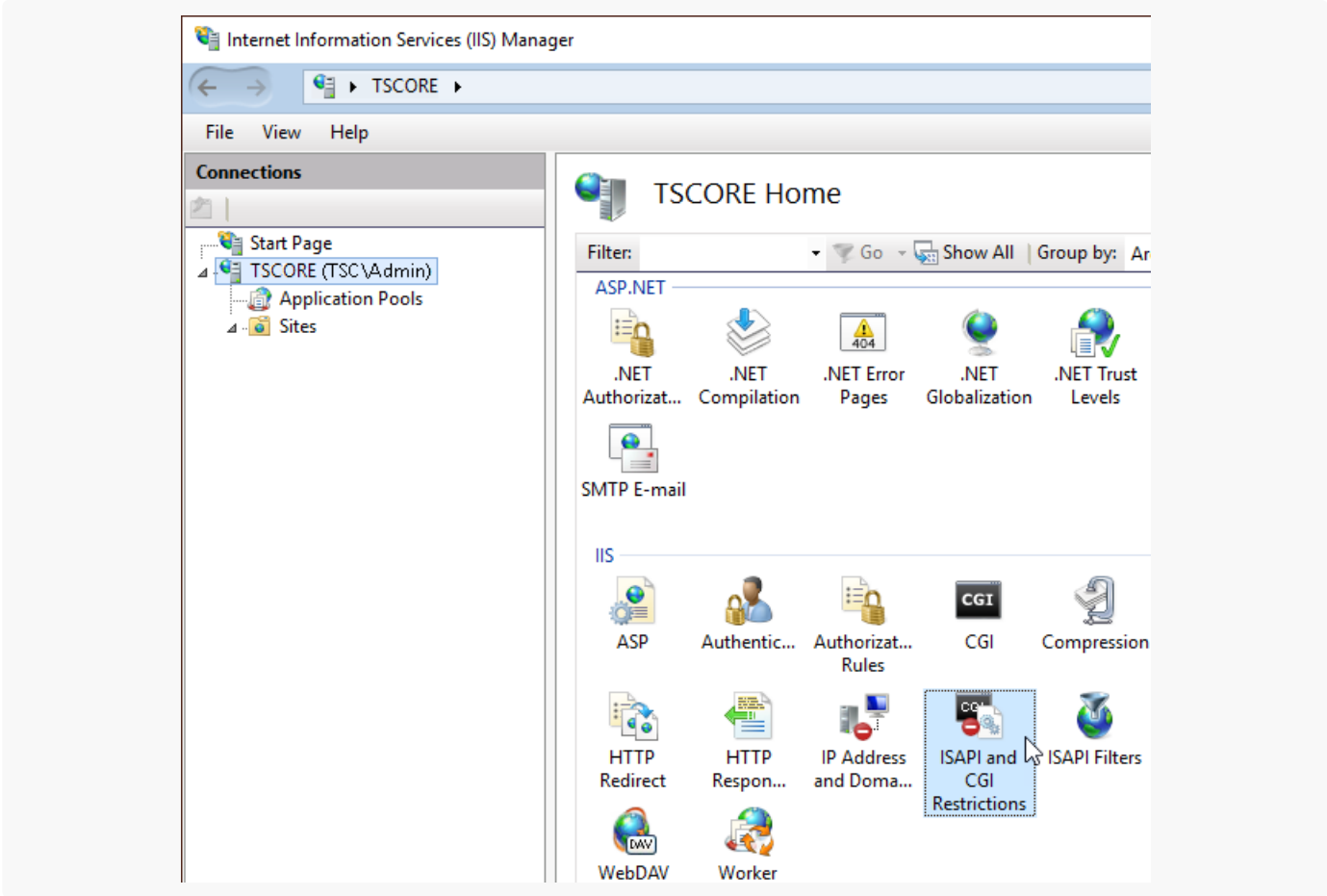
1. В области [*Connections*] окна управления IIS перейдите в секцию [*Applications Pools*].
2. Выберите пул [*Creatio*].
3. В поле [*Managed pipeline mode*] выберите режим [*Integrated*].
4. В поле [*.Net Framework version*] укажите версию Asp.Net - 4.0.30319 (Рис. 5).

Рис. 5 — Окно ввода параметров Applications Pools



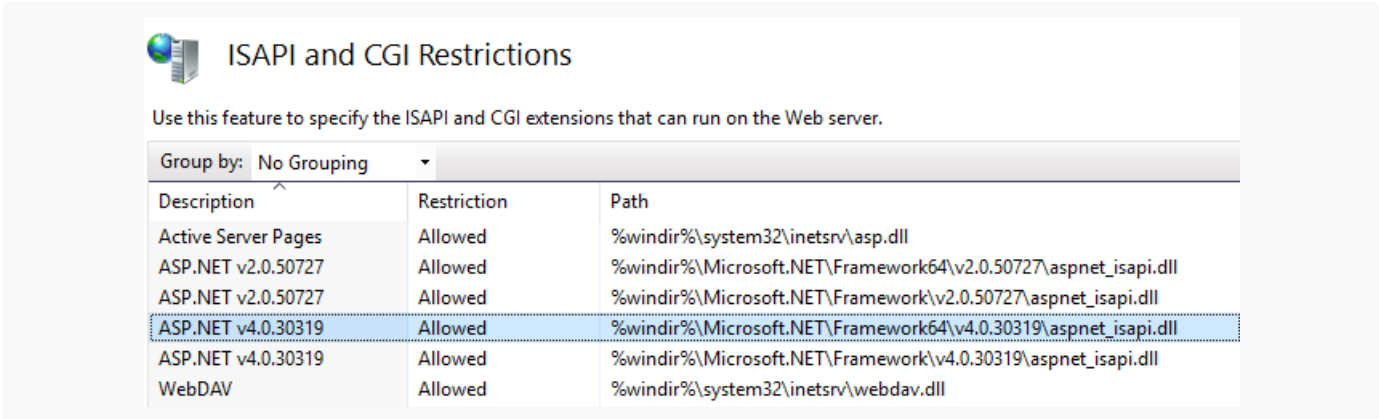
5. Проверьте, что введенная версия ASP.Net разрешена. Для этого перейдите в ISAPI and CGI Restrictions на уровне web-сервера (Рис. 6).

Рис. 6 — Переход в ISAPI and CGI Restrictions



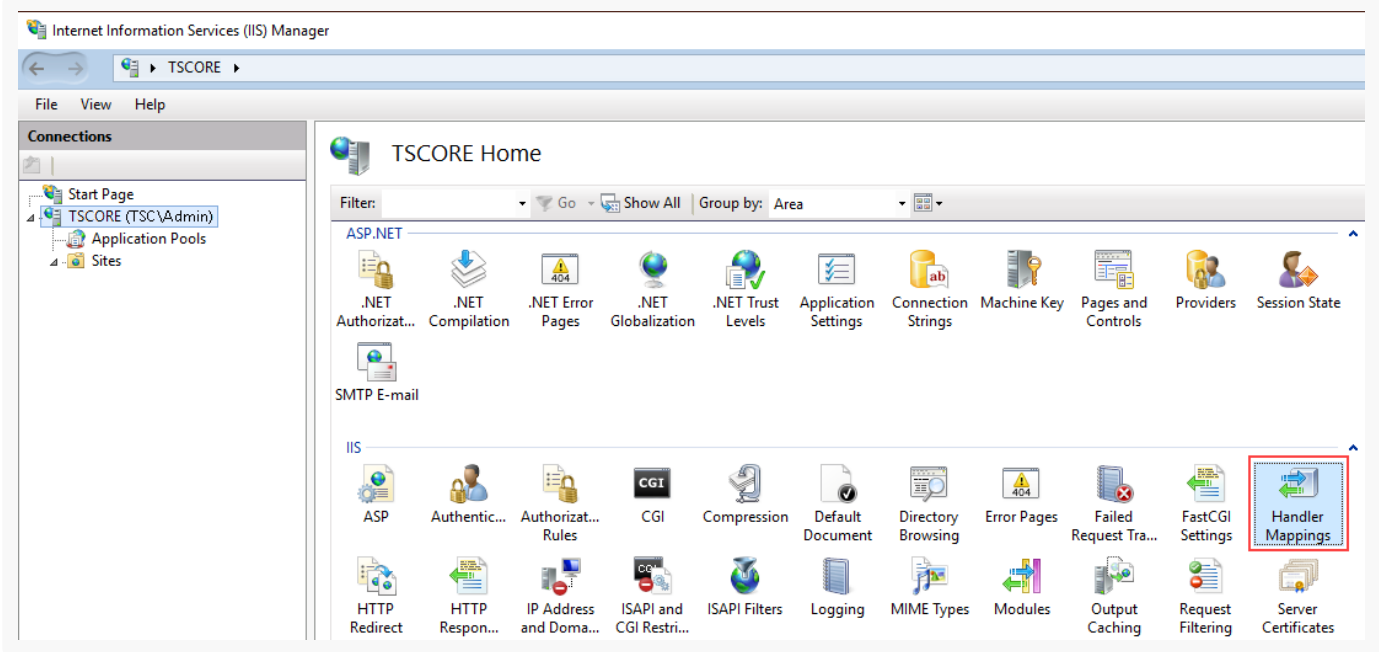
6. Убедитесь что для используемой версии ASP.Net в поле [*Restriction*] установлен статус [*Allowed*] (Рис. 7).

Рис. 7 — Проверка статуса используемой версии ASP.Net



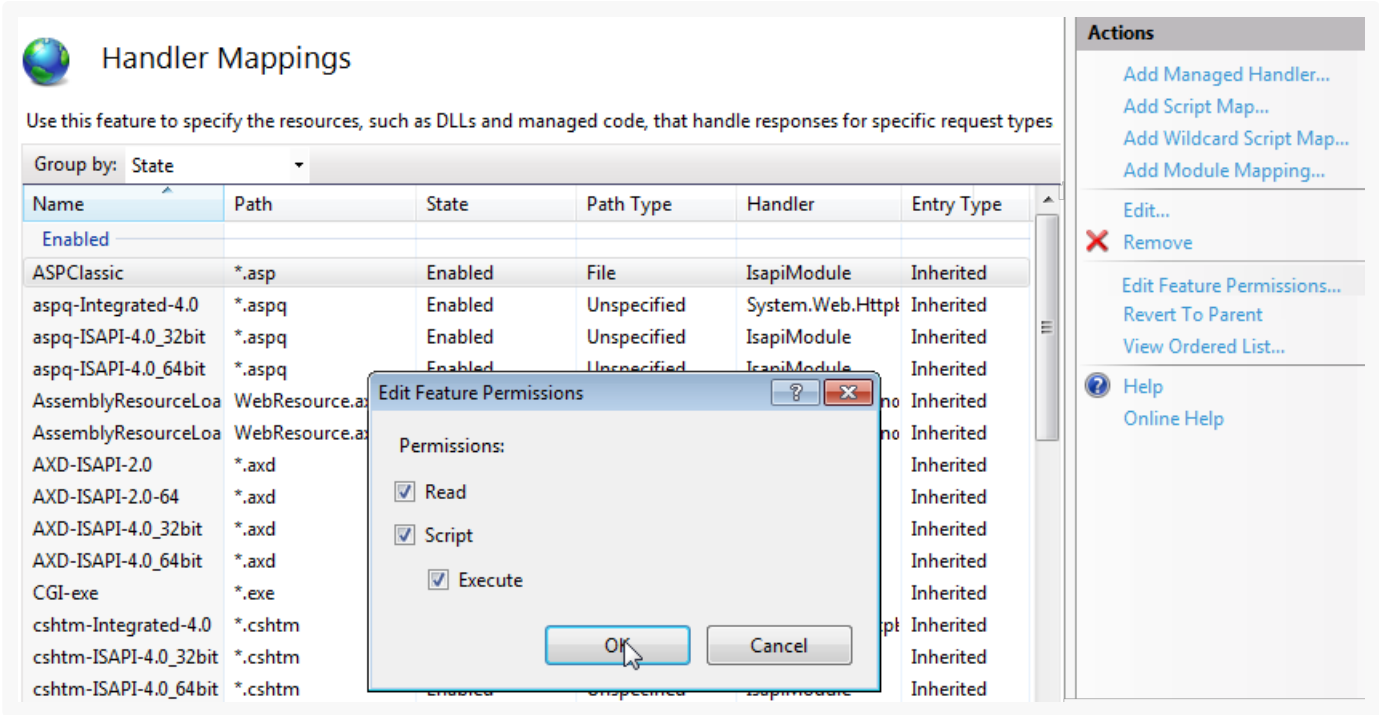
7. Убедитесь в том, что все необходимые разрешения активны. Для этого перейдите в Handler Mappings на уровне сервера (Рис. 8).

Рис. 8 — Переход в Handler Mappings



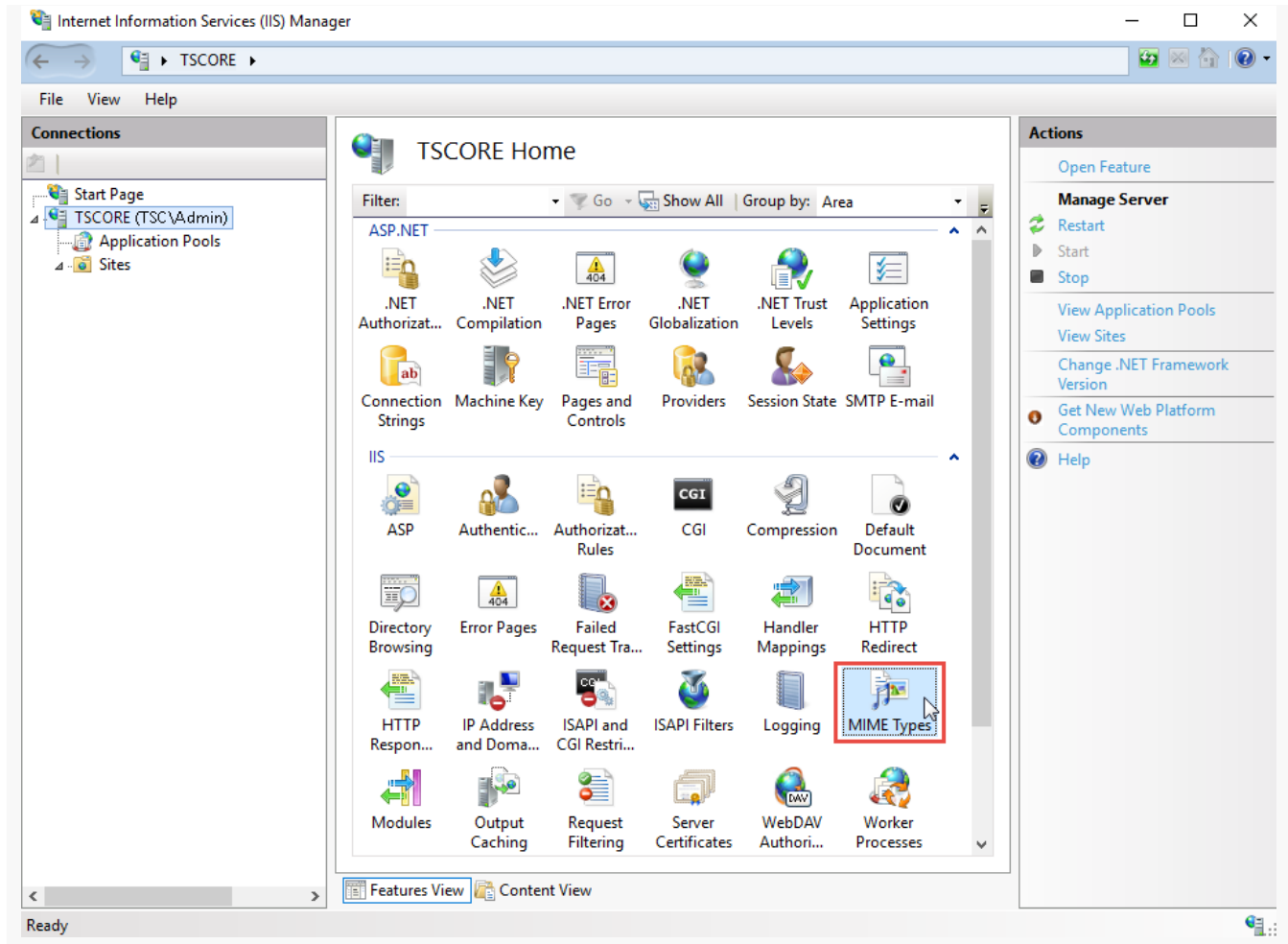
- 8. Нажмите на [*Edit Feature Permissions*] в области [*Actions*].
- 9. Убедитесь, что в окне [*Edit Feature Permissions*] установлены все признаки (Рис. 9).

Рис. 9 — Проверка необходимых параметров в окне [*Edit Feature Permissions*]



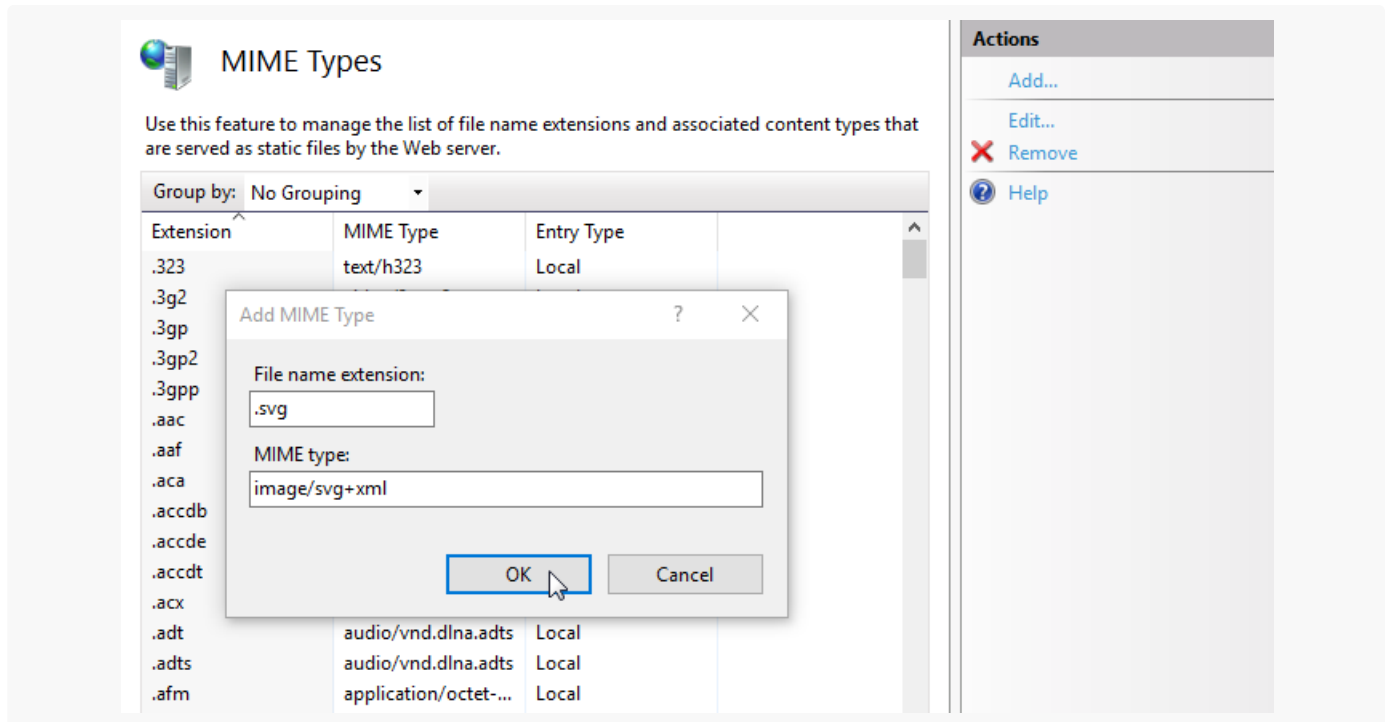
- 10. Убедитесь, что у нового приложения сконфигурирован MIME-тип для файлов с расширениями .svg и .json. Данная конфигурация может быть выполнена как на уровне сервера (в этом случае она наследуется для всех размещенных на нем приложений), так и на уровне приложения. Для проверки:
 - а. Перейдите в MIME Types на уровне сервера или на уровне приложения (Рис. 10).

Рис. 10 — Переход в MIME Types



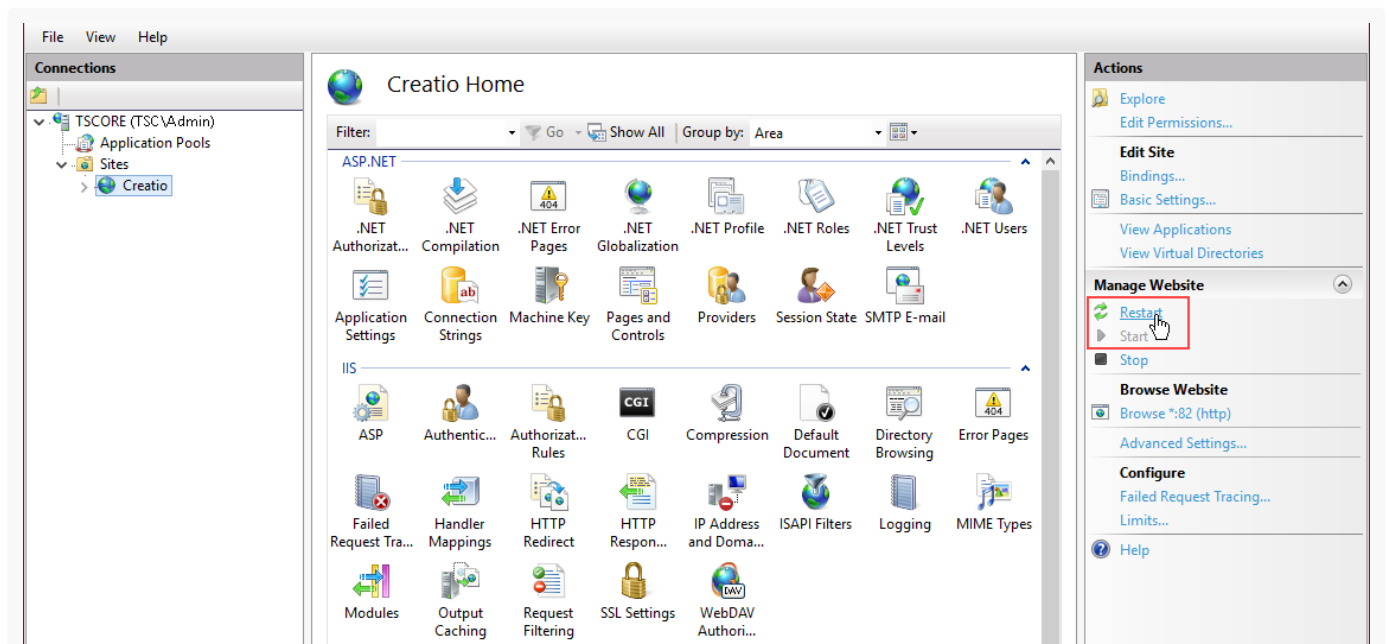
- b. Проверьте в списке наличие конфигурации для файлов с расширением .svg и .json. Если такая конфигурация есть, переходите на шаг 12.
11. Если конфигурации нет, нажмите [Add...] в области [Actions]. В открывшемся окне укажите расширение .svg и MIME-тип данных, соответствующий этому расширению (Рис. 11). Повторите те же действия для расширения .json (MIME-тип "application/json").

Рис. 11 — Настройка MIME-типа данных для файлов с расширением .svg



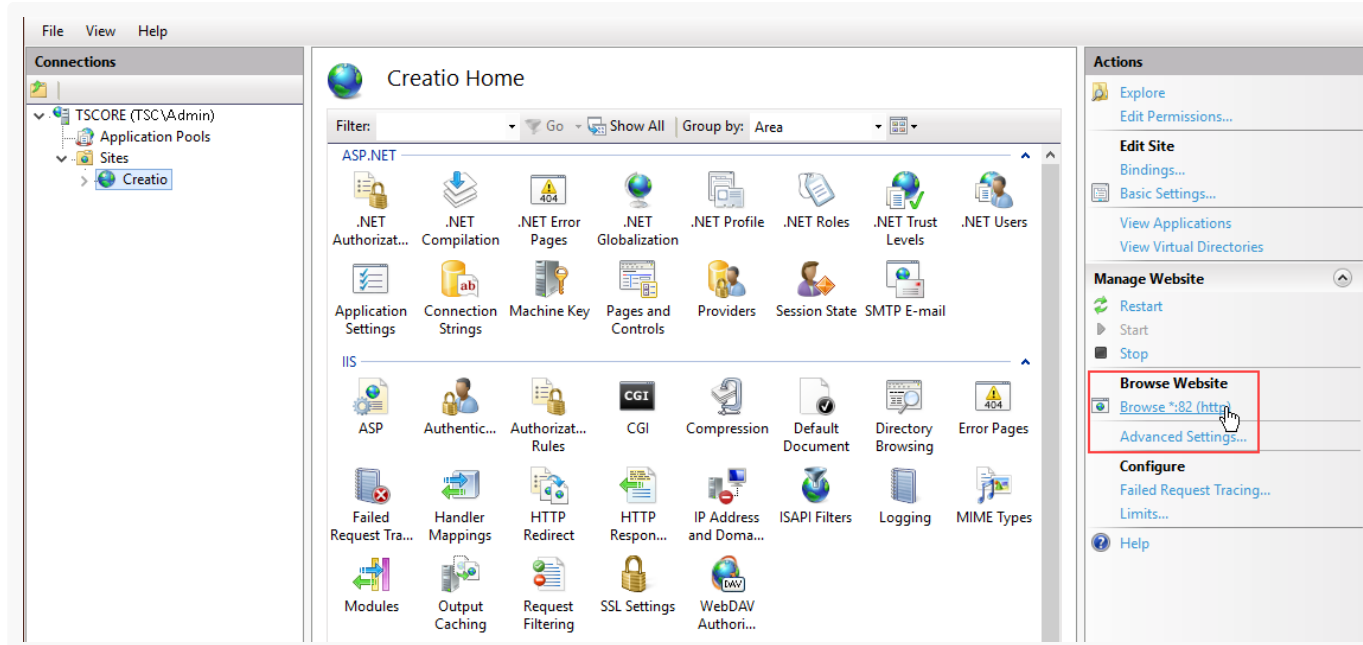
12. Перезапустите web-сайт при помощи команды [*Restart*] области [*Manage Website*] (Рис. 12).

Рис. 12 — Выбор команды [*Restart*] в области [*Manage Websites*]



13. Откройте сайт, перейдя по адресу или при помощи команды [*Browse*] (Рис. 13). Убедитесь, что открывается страница авторизации в приложении.

Рис. 13 — Выбор команды [*Browse*] в области [*Actions*] веб-сайта



На заметку. Первый вход в систему после настройки приложения осуществляется под учетной записью Supervisor (логин — Supervisor; пароль — Supervisor). Рекомендуем после первого входа изменить пароль этой учетной записи.

14. Для включения русского языка:

- Из дизайнера системы перейдите в раздел [*Языки*].
- Выберите русский язык и нажмите [*Открыть*].
- На открывшейся странице установите признаки [*Активен*] и [*Использовать по умолчанию*]. Сохраните изменения.

Для включения языка необходимо, чтобы пользователю, от имени которого в IIS запущен пул приложения, были предоставлены права на чтение, запись и удаление файлов и вложенных каталогов статического контента приложения (каталог .\Terrasoft.WebApp(conf)).

15. Из дизайнера системы перейдите в раздел [*Системные настройки*] и измените значение системной настройки “Порядок ФИО” на “Last name, First name [*Middle name*]”. Это необходимо для корректной разбивки ФИО контактов на отдельные колонки [*Фамилия*], [*Имя*], [*Отчество*].

Настроить вебсокеты

ПРОДУКТЫ: **ВСЕ ПРОДУКТЫ**

Протокол вебсокетов (WebSockets) используется в Creatio для запуска пользовательских процессов, работы уведомлений, а также интеграции с телефонией.

Для корректной работы всех функций системы необходимо, чтобы вебсокеты были включены и настроены на сервере приложения.

На сервере приложений должна быть развернута **Windows Server 2012, Windows Server 2016** или **Windows 8/10** с установленными Internet Information Services (IIS) не ниже версии 8. Конфигурационные

файлы в базовой версии Creatio по умолчанию настроены и необходимо только выполнить настройки со стороны сервера. Однако если вы разворачиваете и настраиваете Creatio впервые, рекомендуем проверить настройки конфигурационных файлов и убедиться, что веб-сокеты работают корректно.

Для использования защищенного соединения HTTPS требуется дополнительная настройка.

Важно. Если в вашей локальной сети используется прокси-сервер, то для него необходимо настроить протокол работы вебсокетов. Как правило, описание настройки данного протокола можно найти в документации прокси-сервера.

В данной статье описан **процесс настройки конфигурационных файлов Creatio** для корректной работы вебсокетов.

На заметку. Установка на сервер компонентов, обеспечивающих работу по протоколу WebSockets, подробно описана в статье [Проверить обязательные компоненты Windows](#).

Проверить настройки вебсокетов для Windows Server 2012 или Windows server 2016

Чтобы проверить настройку вебсокетов в Creatio, развернутом на сервере под управлением Windows Server 2012 или Windows Server 2016:

1. Убедитесь, что в файле Web.config, который находится в корневой директории сайта Creatio, отключено наследование, а также указаны ограничения на длину и время выполнения запроса.

```
<location path="." inheritInChildApplications="false">
  <system.web>
    ...
    <httpRuntime maxRequestLength="73400" executionTimeout="28800" targetFramework="4.7" />
  </system.web>
</location>
```

2. Для файла Web.config, который размещен в директории [*Путь к корневому каталогу Creatio*]\Terrasoft.WebApp\ проверьте, чтобы было указано значение типа wsService по умолчанию **“Terrasoft.Messaging.MicrosoftWSService.MicrosoftWSService, Terrasoft.Messaging.MicrosoftWSService”**, указан порт для подключения клиентов, ограничения на длину и время выполнения запроса HTTP, а также вызов дополнительных модулей.

Важно. Для Microsoft Windows Server версии 2012 и выше рекомендуется использовать значение “MicrosoftWSService”, а не “SuperWSService”.

На заметку. Значение portForClientConnection="0" означает использование порта web-приложения.

```

<wsService type="Terrasoft.Messaging.MicrosoftWSService.MicrosoftWSService,
Terrasoft.Messaging.MicrosoftWSService" encrypted="false" portForClientConnection="0" />
...
<location path="." inheritInChildApplications="false">
  <system.web>
    ...
    <httpRuntime maxRequestLength="102400" executionTimeout="28800"
targetFramework="4.6.2" />
    <httpHandlers>
      ...
      <add verb="GET" path="*ViewModule.aspx.ashx" type="Terrasoft.Messaging.MicrosoftWSSer
Terrasoft.Messaging.MicrosoftWSService" />
    ...
  </system.web>
  <system.webServer>
    ...
    <handlers>
      ...
      <add name="WSHandler" verb="*" path="*ViewModule.aspx.ashx"
type="Terrasoft.Messaging.MicrosoftWSService.WSHandler, Terrasoft.Messaging.MicrosoftWSServic

```

На заметку. Подключение по протоколу WebSockets можно проверить при помощи консоли разработчика (Developer Tools) веб-браузера. В случае успешного подключения в консоли будет содержаться запись типа: WebSocket-connection opened for url:ws://demo.creatio.com/0/Nui/ViewModule.aspx.ashx

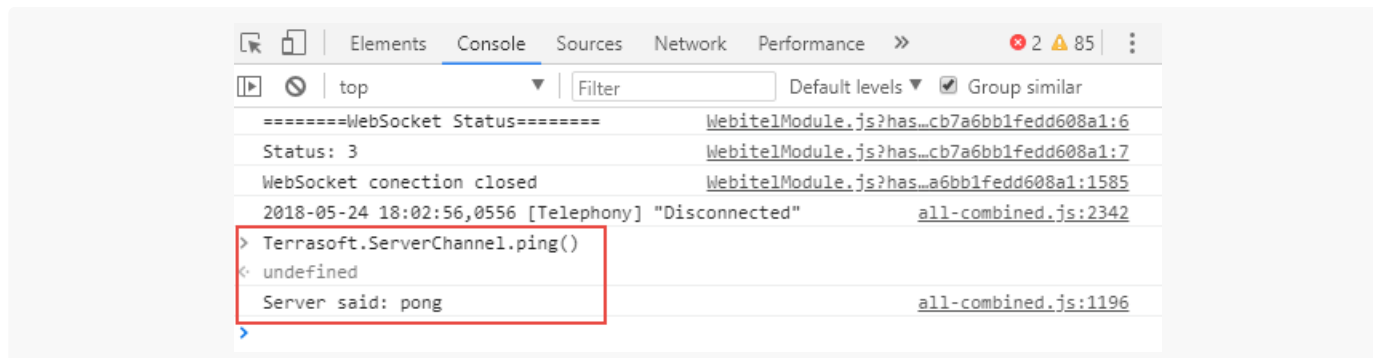
Часто задаваемые вопросы по настройке вебсокетов

Как проверить правильность настройки вебсокетов?

Существует несколько быстрых способов проверить корректность настройки вебсокетов:

- Воспользуйтесь функциональностью [импорта данных из Excel](#). Если вебсокеты настроены правильно, то импорт данных будет выполнен.
- В консоли разработчика веб-браузера введите команду `Terrasoft.ServerChannel.ping()`. При корректной настройке вебсокетов сервер ответит `pong` (Рис. 1). При любом другом ответе сервера необходимо проверить правильность настройки вебсокетов.

Рис. 1 — Проверка настройки вебсокетов при помощи консоли разработчика веб-браузера



- Запустите вручную бизнес-процесс, в котором есть стартовый таймер и автогенерируемая страница. Если автогенерируемая страница откроется, значит, вебсокеты настроены правильно.

На заметку. Элементы бизнес-процессов подробно описаны в [отдельном блоке](#).

Я настроил вебсокеты, но они не работают. Почему?

Если после настройки вебсокеты не работают, проверьте:

- Установлены ли на сервере все компоненты протокола WebSockets. [Подробнее >>>](#)
- Установлен ли протокол использования вебсокетов для прокси-сервера, если он используется в вашей локальной сети.
- Настройки антивируса и Firewall. Если нет возможности отключить на сервере эти программы, добавьте IP-адрес и порт вашего сайта Creatio в список исключений для входящих и исходящих потоков.
- Наличие расширений и надстроек web-браузера, в том числе VPN. Расширения могут блокировать работу вебсокетов.

Перевести Creatio с HTTP на HTTPS

ПРОДУКТЫ: [ВСЕ ПРОДУКТЫ](#)

Основная задача протокола HTTPS — обеспечение защищенного соединения между клиентом и веб-сервисом. Перевод Creatio с HTTP на HTTPS повысит безопасность системы и позволит использовать дополнительные сервисы, например, webitel с поддержкой WebRTC. Обратите внимание, что настройка потребуется, только если приложение развернуто on-site. Для перехода на HTTPS нужно изменить некоторые параметры веб-сайта в IIS и отредактировать конфигурационный файл Web.config. Пользователи cloud по умолчанию работают через защищенное соединение.

На заметку. Вы не сможете использовать преимущества HTTPS, если приложение Creatio развернуто на Windows Server 2008.

Настроить IIS

Прежде чем приступить к настройке HTTPS, вам необходимо:

- Получить цифровой сертификат от центра сертификации в формате PFX;

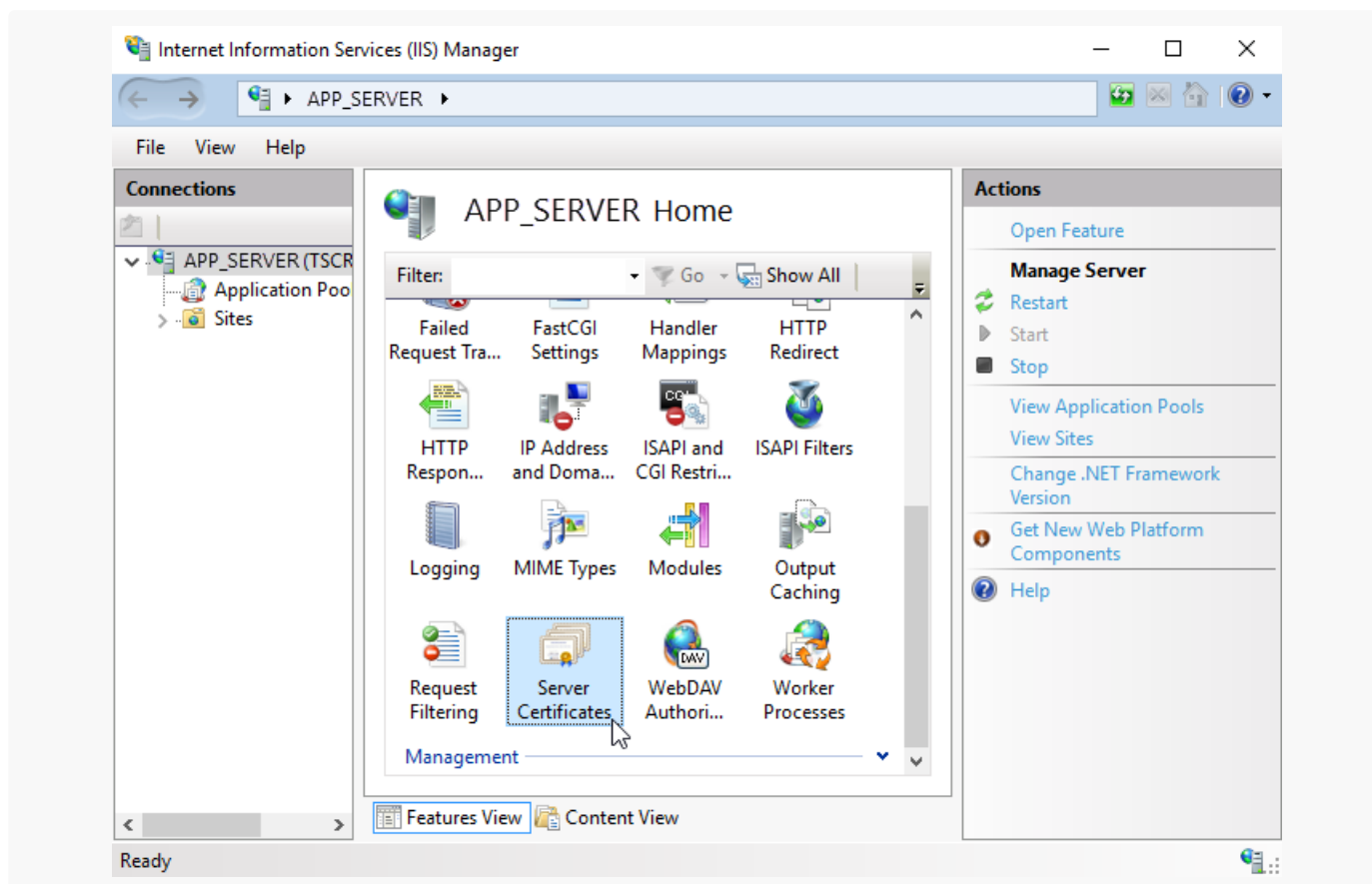
На заметку. При использовании самоподписанного сертификата подключение мобильного приложения к сайту Creatio будет недоступно из-за ограничений политики безопасности мобильных приложений.

- Произвести [настройки вебсокетов](#) для корректной работы всех компонентов системы;
- Рекомендуем также проверить список установленных [компонентов IIS](#), чтобы избежать ошибок в процессе настройки и работы с Creatio.

Полученный цифровой сертификат необходимо загрузить в хранилище сертификатов сервера:

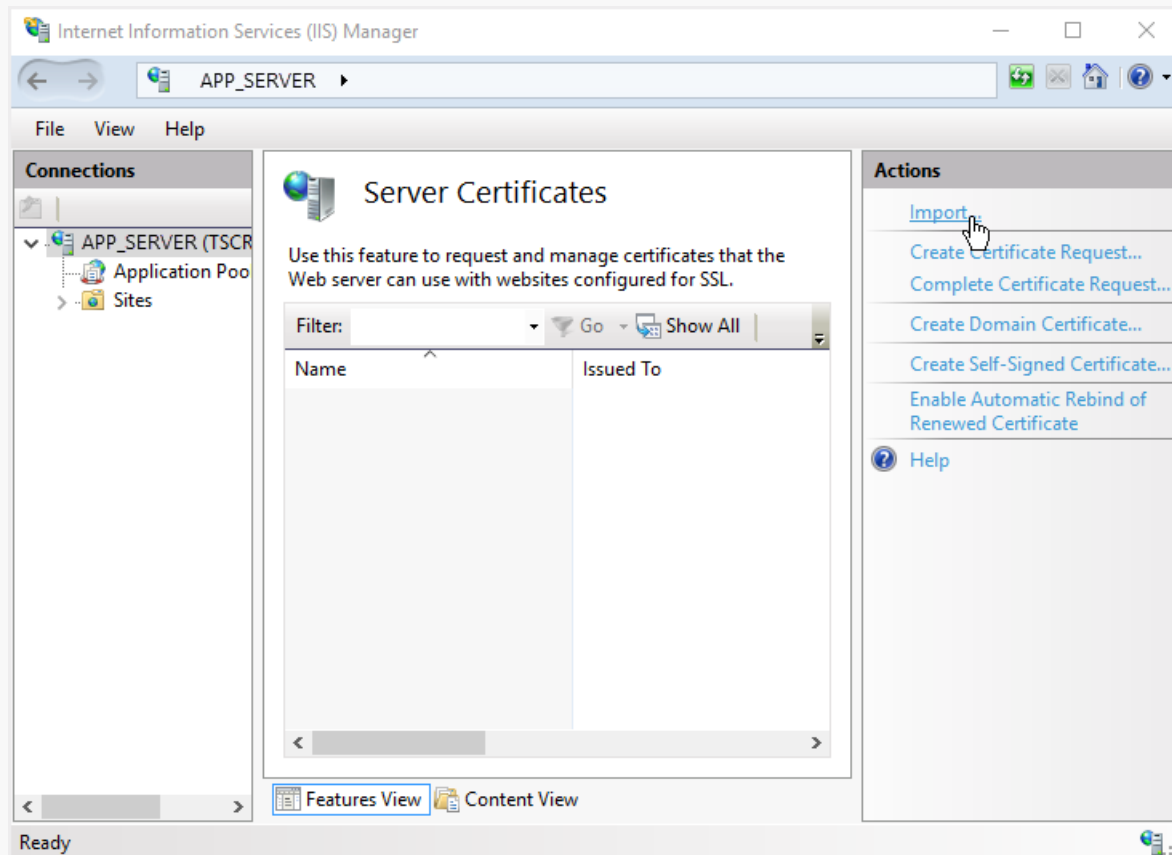
1. Откройте Internet Information Services (IIS) Manager.
2. В основном окне IIS выберите деталь [*Server Certificates*] двойным кликом мыши ([Рис. 1](#)).

Рис. 1 — Выбор детали [*Server Certificates*]



3. В окне [*Server Certificates*] нажмите на ссылку [*Import*] в меню действий справа ([Рис. 2](#)).

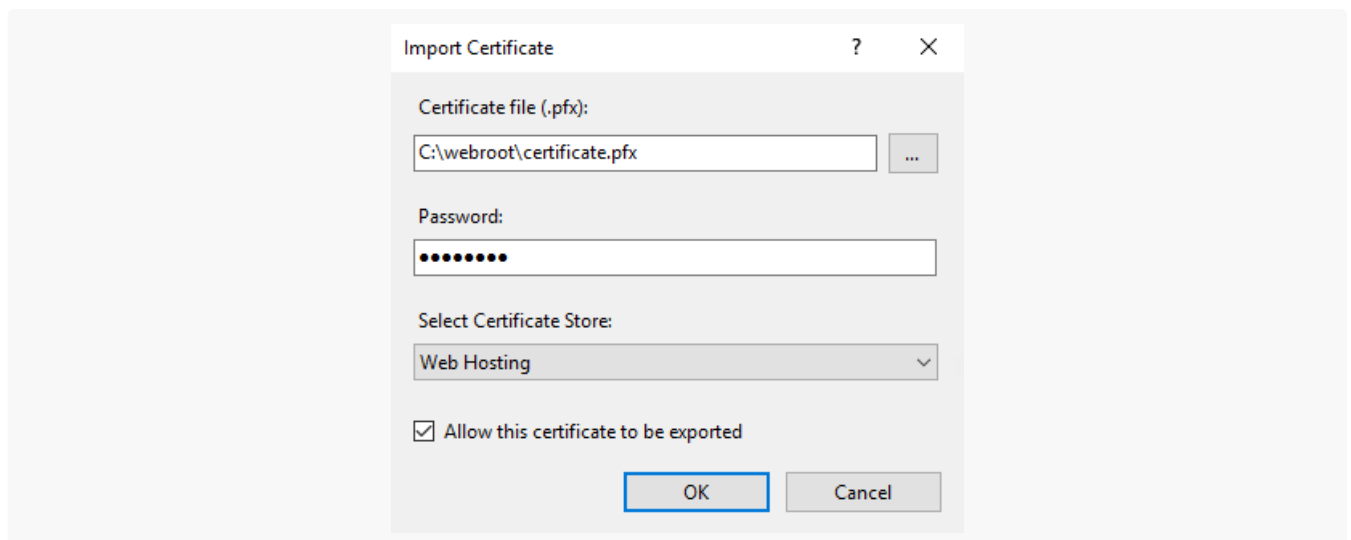
Рис. 2 — Открытие окна [*Import*]



4. В диалоговом окне импорта укажите:

- Путь к размещенному на сервере файлу импорта;
- Пароль (при необходимости);
- Хранилище сертификатов (Рис. 3).

Рис. 3 — Окно импорта сертификата

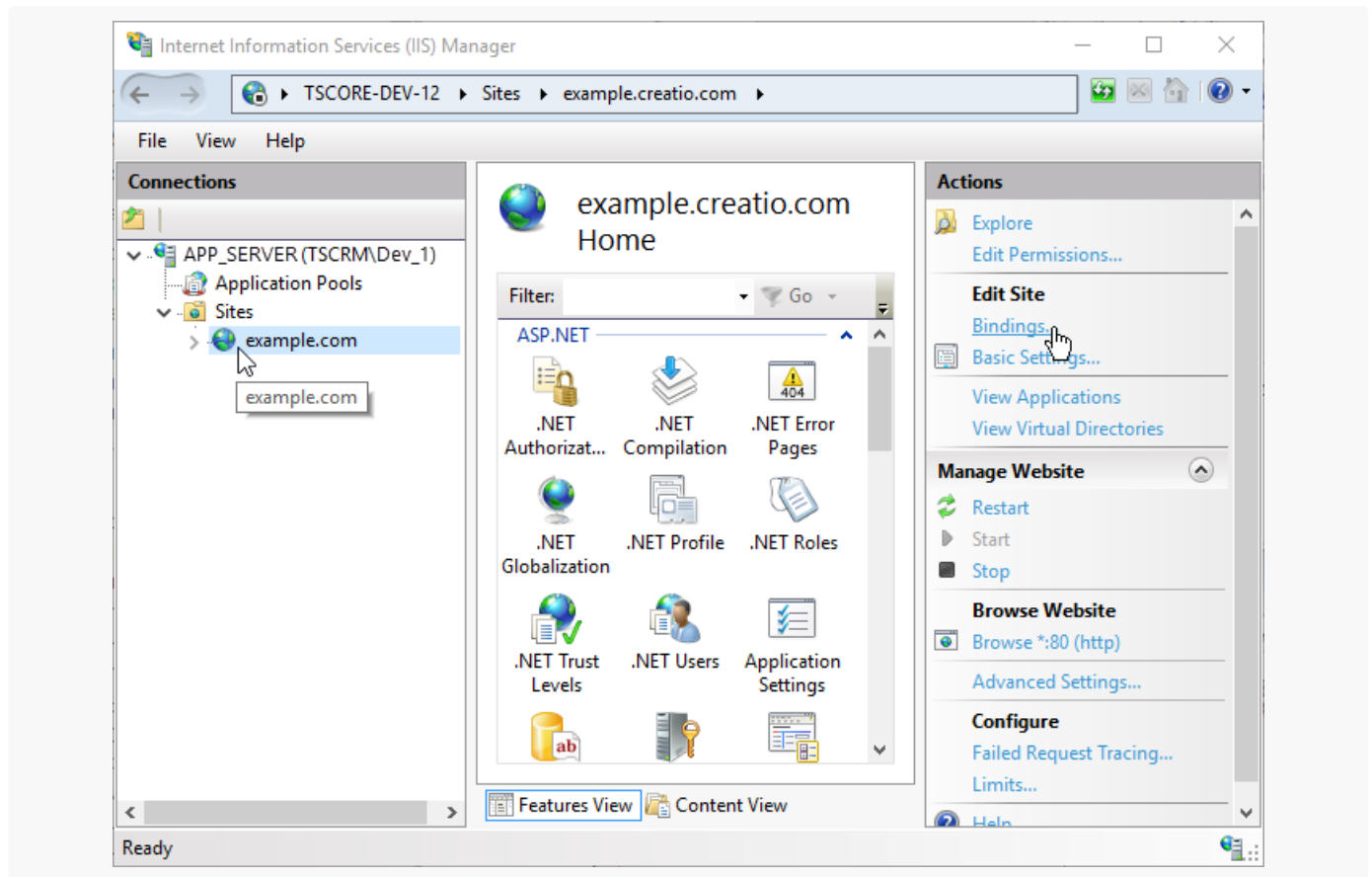


5. Нажмите на кнопку [OK], чтобы импортировать сертификат.

Далее следует привязать импортированный сертификат к приложению Creatio:

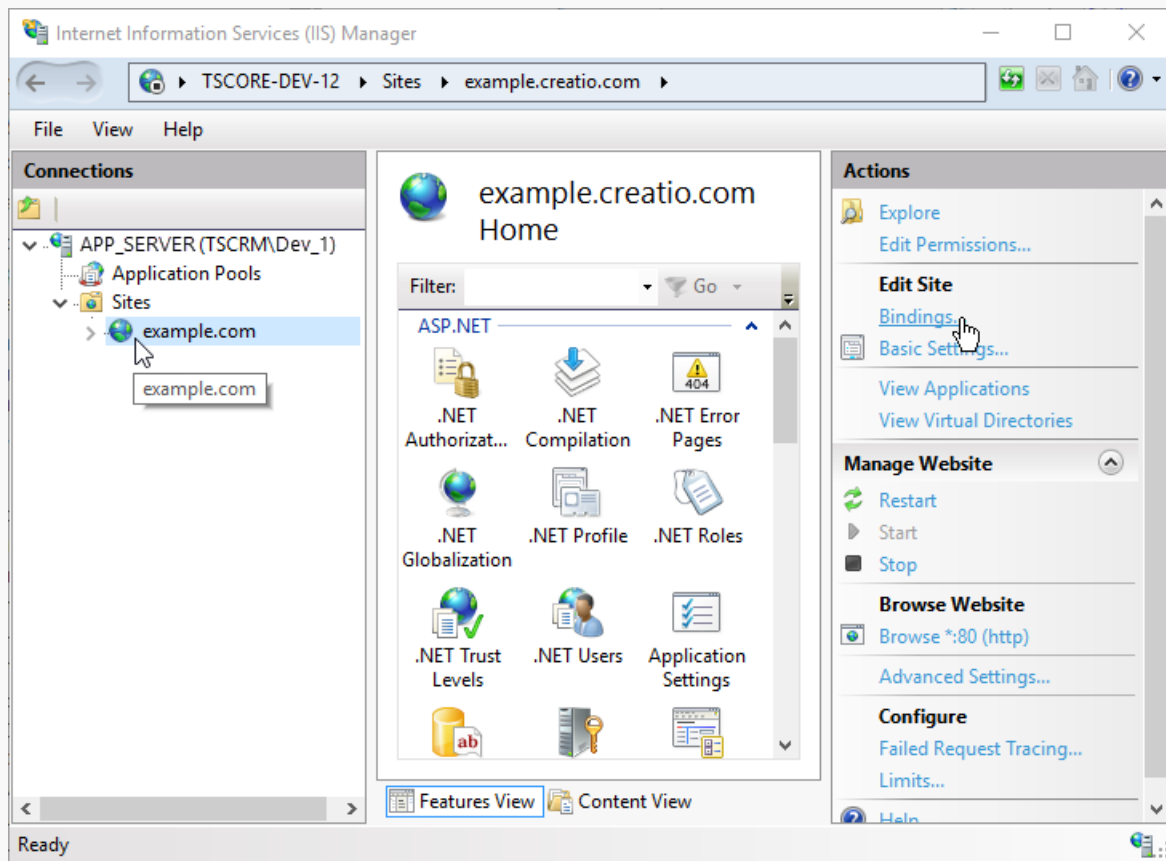
1. В окне IIS перейдите к сайту приложения, нажав на его название в левом меню [*Connections*] (Рис. 4).

Рис. 4 — Выбор сайта Creatio в окне IIS



2. Нажмите на ссылку [*Bindings*] в меню действий (Рис. 5).

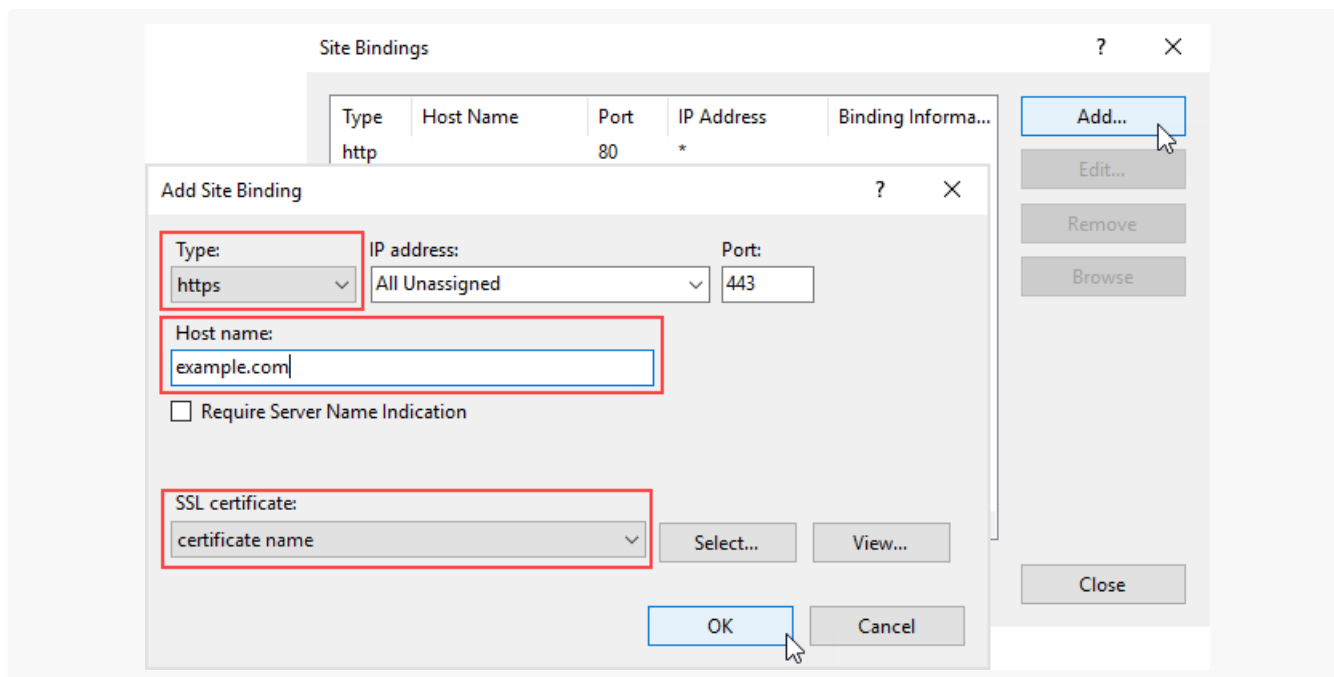
Рис. 5 — Выбор связей сайта



3. В меню связей сайта нажмите на кнопку [*Add*] и добавьте новую связь. В окне [*Add Site Binding*] укажите:

- a. Тип "https ";
- b. Адрес сайта;
- c. Электронный сертификат SSL ([Рис. 6](#)).

Рис. 6 — Привязка сертификата к сайту Creatio



4. Нажмите [OK], чтобы подтвердить настройки.

Теперь сертификат успешно привязан к веб-приложению.

Настроить Web.config

После добавления сертификата необходимо внести изменения в конфигурационный файл Web.config, который размещен в **корневой директории** сайта Creatio:

1. В конце файла найдите следующие строки:

```
<behaviors configSource="Terrasoft.WebApp\ServiceModel\http\behaviors.config" />
<bindings configSource="Terrasoft.WebApp\ServiceModel\http\bindings.config" />
```

2. Замените пути "http" на "https", чтобы получилось:

```
<behaviors configSource="Terrasoft.WebApp\ServiceModel\https\behaviors.config" />
<bindings configSource="Terrasoft.WebApp\ServiceModel\https\bindings.config" />
```

Далее необходимо отредактировать файл Web.config, который размещен в директории **Путь к корневому каталогу сайта\Terrasoft.WebApp**:

1. Задайте значение переменной encrypted="true". Настройка отличается в зависимости от операционной системы сервера с приложением Creatio.

Для Windows Server 2012 и новее конфигурационную строку нужно привести к виду:

```
<wsService
  type="Terrasoft.Messaging.MicrosoftWSService.MicrosoftWSService,
```

```
Terrasoft.Messaging.MicrosoftWSService" encrypted="true"  
portForClientConnection="443" maxConnectionNumber="100"  
clearIdleSession="false" clearIdleSessionInterval="120" />
```

2. В конце файла найдите строчку:

```
<services configSource="ServiceModel\http\services.config" />
```

3. Измените путь с "http" на "https", чтобы получилось:

```
<services configSource="ServiceModel\https\services.config" />
```

После всех изменений не забудьте сохранить файлы конфигураций.

Перезапустите приложение при помощи IIS, а затем зайдите на ваш сайт Creatio. Если все действия выполнены верно, то в строке адреса вы увидите "https://" перед веб-адресом приложения.