

**Распределенная информационно-телекоммуникационная  
система мониторинга и диспетчерского управления  
транспортом  
("ИТС «Мониторинг транспорта-2»")  
Руководство администратора**

**2023**

## СОДЕРЖАНИЕ

1	Состав.....	3
2	Назначение.....	4
3	Запуск ПО .....	5
	3.1 Запуск ПО в Windows.....	5
	3.2 Запуск ПО в AstraLinux.....	5
4	Остановка ПО.....	8
5	Конфигурирование параметров соединения с БД.....	9
6	Настройка связи между модулями .....	10

# 1 Состав

Распределенная информационно-телекоммуникационная система мониторинга и диспетчерского управления транспортом (далее ПО) состоит из следующих основных модулей:

- сервер бизнес-логики (далее Сервер);
- сервер публикации Web-интерфейса (далее Web-сервер).

## **2 Назначение**

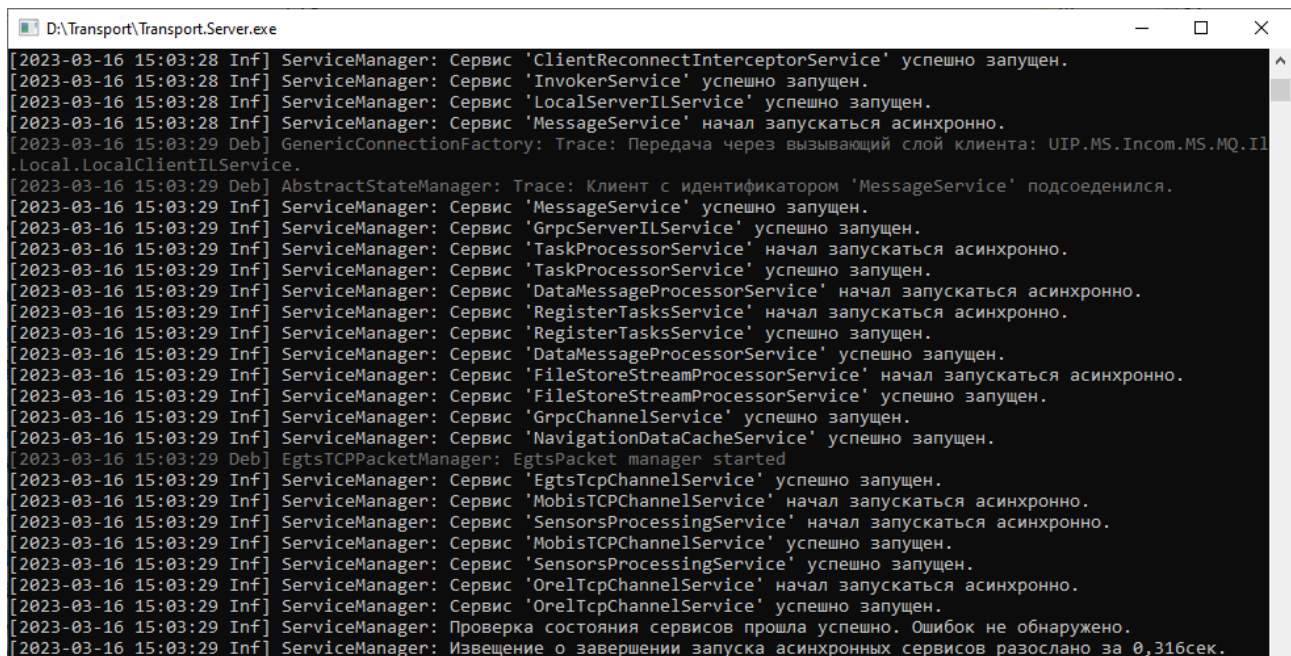
Распределенная информационно-телекоммуникационная система мониторинга и диспетчерского управления транспортом предназначена для наблюдения и контроля передвижения транспортных средств, а также просмотра их трека движения за предыдущие периоды.

ИТС «Мониторинг транспорта-2» — это комплекс сервисных программ и модулей, доступных разработчику. Сервер предоставляет интегрированную среду для развертывания и выполнения высокопроизводительных серверных бизнес-приложений и управления ими. Эти приложения могут обслуживать запросы, принимаемые от удаленных клиентских систем, в том числе, подключающихся из Интернета, корпоративной сети или интрасети. Предоставляет разработчикам упрощенную модель программирования сетевых серверных приложений. Разработчики могут использовать встроенные библиотеки для реализации в приложениях множества функций, таких как ввод-вывод, обработка численных данных и текста, доступ к базам данных, управление транзакциями, бизнес-правила и веб-службы.

## 3 Запуск ПО

### 3.1 Запуск ПО в Windows

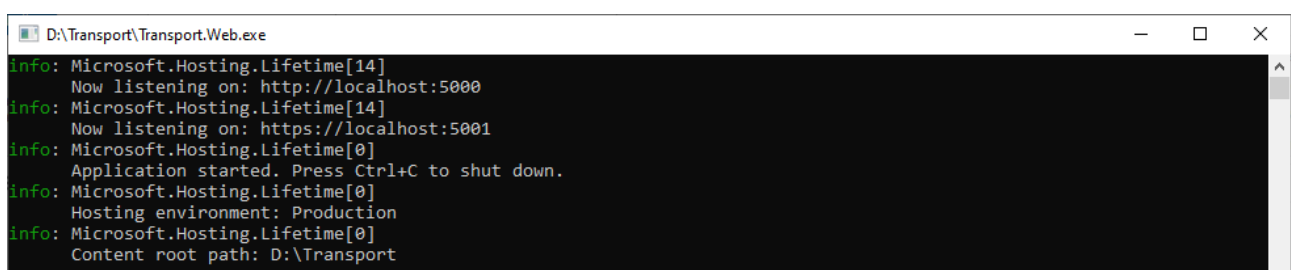
Для запуска Сервера в соответствующей папке нужно запустить файл Transport.Server.exe, откроется консольное окно, отображающее журнал запуска.



```
D:\Transport\Transport.Server.exe
[2023-03-16 15:03:28 Inf] ServiceManager: Сервис 'ClientReconnectInterceptorService' успешно запущен.
[2023-03-16 15:03:28 Inf] ServiceManager: Сервис 'InvokerService' успешно запущен.
[2023-03-16 15:03:28 Inf] ServiceManager: Сервис 'LocalServerILService' успешно запущен.
[2023-03-16 15:03:28 Inf] ServiceManager: Сервис 'MessageService' начал запускаться асинхронно.
[2023-03-16 15:03:29 Deb] GenericConnectionFactory: Trace: Передача через вызывающий слой клиента: UIP.MS.Incom.MS.MQ.II
.Local.LocalClientILService.
[2023-03-16 15:03:29 Deb] AbstractStateManager: Trace: Клиент с идентификатором 'MessageService' подключился.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'MessageService' успешно запущен.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'GrpcServerILService' успешно запущен.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'TaskProcessorService' начал запускаться асинхронно.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'TaskProcessorService' успешно запущен.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'DataMessageProcessorService' начал запускаться асинхронно.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'RegisterTasksService' начал запускаться асинхронно.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'RegisterTasksService' успешно запущен.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'DataMessageProcessorService' успешно запущен.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'FileStoreStreamProcessorService' начал запускаться асинхронно.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'FileStoreStreamProcessorService' успешно запущен.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'GrpcChannelService' успешно запущен.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'NavigationDataCacheService' успешно запущен.
[2023-03-16 15:03:29 Deb] EgtsTCPpacketManager: EgtsPacket manager started
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'EgtsTcpChannelService' успешно запущен.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'MobistTCPChannelService' начал запускаться асинхронно.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'SensorsProcessingService' начал запускаться асинхронно.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'MobistTCPChannelService' успешно запущен.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'SensorsProcessingService' успешно запущен.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'OrelTcpChannelService' начал запускаться асинхронно.
[2023-03-16 15:03:29 Inf] ServiceManager: Сервис 'OrelTcpChannelService' успешно запущен.
[2023-03-16 15:03:29 Inf] ServiceManager: Проверка состояния сервисов прошла успешно. Ошибок не обнаружено.
[2023-03-16 15:03:29 Inf] ServiceManager: Извещение о завершении запуска асинхронных сервисов разослано за 0,316сек.
```

Если всё установлено и сконфигурировано корректно, то в последних строках журнала можно увидеть строку «ServiceManager: Проверка состояния сервисов прошла успешно. Ошибок не обнаружено.»

Для запуска Web-сервера в соответствующей папке нужно запустить файл Transport.Web.exe, откроется консольное окно, отображающее журнал запуска.



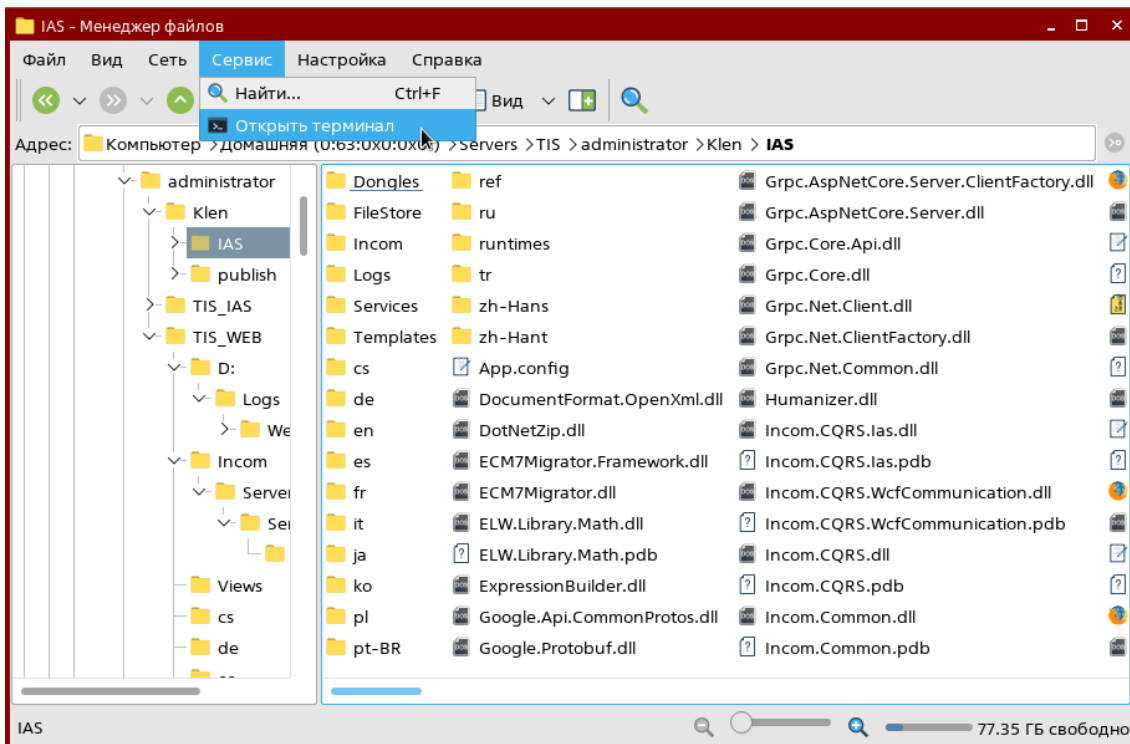
```
D:\Transport\Transport.Web.exe
info: Microsoft.Hosting.Lifetime[14]
  Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[14]
  Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
  Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
  Hosting environment: Production
info: Microsoft.Hosting.Lifetime[0]
  Content root path: D:\Transport
```

Если всё установлено и сконфигурировано корректно, то в открытом консольном будет присутствовать строка «Application started. Press Ctrl+C to shut down.»

### 3.2 Запуск ПО в AstraLinux

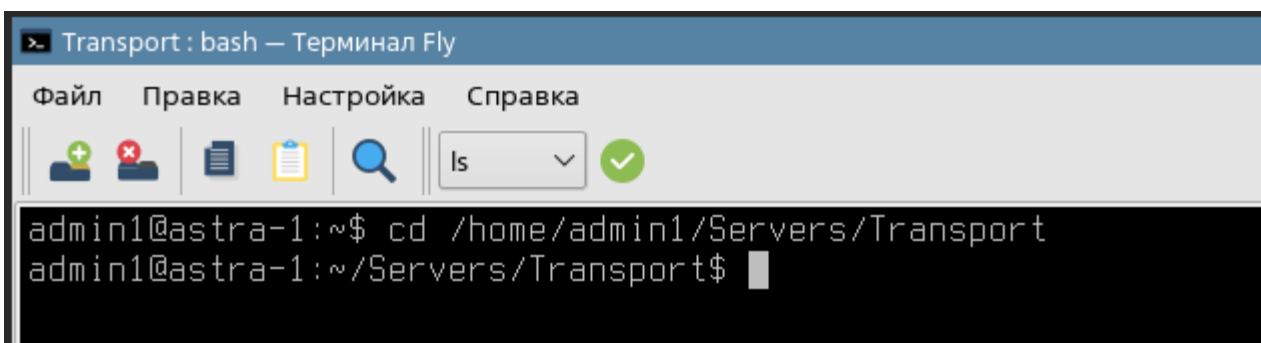
Для запуска ПО под AstraLinux нужно открыть по окну терминала для каждой папки куда разархивировано ПО. Для этого в менеджере файлов

перейдите в соответствующую папку и выберите в меню «Сервис» пункт «Открыть терминал».

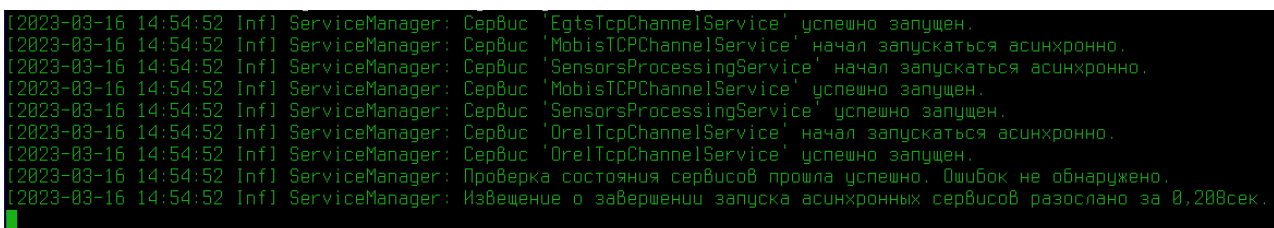


Также можно открыть терминал из меню приложений системы и перейти в нужную папку с помощью команды:

```
cd [путь к папке ПО]
```



Для запуска Сервера в соответствующем терминале нужно выполнить команду «dotnet Transport.Server.dll»



Если всё установлено и сконфигурировано корректно, то в последних строках журнала можно увидеть строку «ServiceManager: Проверка состояния сервисов прошла успешно. Ошибок не обнаружено»

Для запуска Сервера в соответствующем терминале нужно выполнить команду «dotnet Transport.Web.dll»

```
admin1@astra-1:~/Servers/Transport/web$ sudo /home/admin1/dotnet/dotnet Transport.Web.dll
[sudo] пароль для admin1:
Unknown device "/dev/sda": Permission denied
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Production
info: Microsoft.Hosting.Lifetime[0]
      Content root path: /home/admin1/Servers/Transport/web
```

Если всё установлено и сконфигурировано корректно, то в открытом консольном будет присутствовать информация о запущенном приложении (как на скриншоте) и отсутствие каких-либо ошибок.

После запуска приложений в браузере можно будет открыть сайт по адресу <http://localhost:5000> и увидеть ПО в работе.

## 4 Остановка ПО

Вне зависимости от ОС в которой запущено ПО, целесообразно сначала останавливать Web-сервер а затем Сервер. Поскольку в обоих случаях приложения работают в консолях, оба останавливаются посредством комбинации клавиш «Ctrl» + «С».

После остановки Web-сервер выведет сообщение о скором закрытии, после чего в Windows консольное окно закроется, а в AstraLinux вернётся строка ввода в терминале.

```
info: Microsoft.Hosting.Lifetime[0]
      Application is shutting down...
```

После остановки Сервер, запустит процесс остановки всех запущенных сервисов, и в конце будет строка, сообщающая о завершении остановки.

```
{2022-12-20 14:49:47 Inf} ServiceManager: Для сервиса 'IndicatorCalculationService' освобождены ресурсы.
{2022-12-20 14:49:47 Inf} ServiceManager: Сервис 'IndicatorLastValueObserverService' деинициализирован.
{2022-12-20 14:49:47 Inf} ServiceManager: Для сервиса 'IndicatorLastValueObserverService' освобождены ресурсы.
{2022-12-20 14:49:47 Inf} ServiceManager: Сервис 'MonitoringObjectsTreeService' деинициализирован.
{2022-12-20 14:49:47 Inf} ServiceManager: Для сервиса 'MonitoringObjectsTreeService' освобождены ресурсы.
{2022-12-20 14:49:47 Inf} ServiceManager: Сервис 'ExcelManagerService' деинициализирован.
{2022-12-20 14:49:47 Inf} ServiceManager: Сервис 'PathManagerService' деинициализирован.
{2022-12-20 14:49:47 Inf} ServiceManager: Сервис 'WordManagerService' деинициализирован.
{2022-12-20 14:49:47 Inf} ApplicationServer: Сервер остановлен.
Server stopped, press any key to close
```

В отличие от Web-сервера, Сервер после остановки сервисов, уже не может использоваться для работы ПО, но остаётся открытым, и повторное нажатие сочетания клавиш «Ctrl» + «С» останавливает его окончательно, также либо закрывая консольное окно, либо возвращая в строку ввода терминала.



## 5 Конфигурирование параметров соединения с БД

В папке с файлами Сервера найдите файл `appsettings.json` и откройте его в любом текстовом редакторе. В файле содержится единственный параметр — строка подключения к БД.

```
{  
  "ConnectionStrings": {  
    "Default": "host=localhost;port=5435;database=Transport;username=postgres;password=123456;"  
  }  
}
```

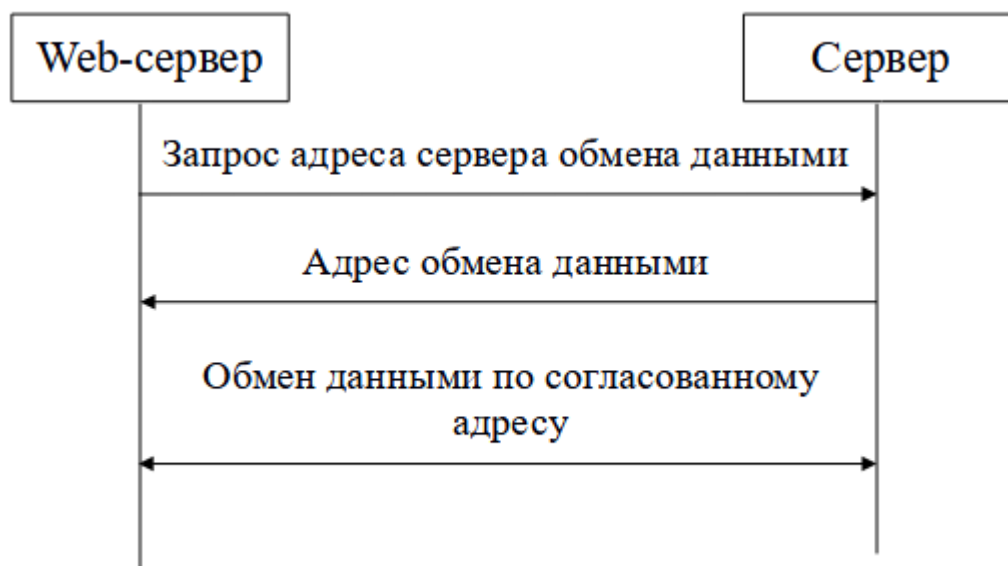
Если вы устанавливали PostgreSQL на этом же компьютере тогда в строке подключения нужно заменить параметры `database={название созданной БД}` и пароль пользователя `password={заданный при установке пароль}`.

Если вы решили использовать другой сервер PostgreSQL то укажите его параметры.

## 6 Настройка связи между модулями

Если Сервер и Web-сервер запускаются на одном компьютере, то необходимости настраивать что-то дополнительно нет. Однако, само по себе разделение ПО на модули, подразумевает возможность установки их на разные ЭВМ ради безопасности Сервера и данных в БД. В этом случае следует согласовать конфигурацию сервера и сетевой безопасности.

В общем виде взаимодействие Сервера и Web-сервера выглядит так:



Со стороны Сервера конфигурация удалённого доступа настраивается в файле «{Папка Сервера}\Services\System\GrpcHostManager.service.config», который выглядит так:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <service name="GrpcHostManager" assemblyName="UIP.GrpcChannel.Server"
    className="UIP.GrpcChannel.Server.Service.GrpcHostManagerService">
    <parameter name="ServiceUrl" value="localhost"/>
    <parameter name="GrpcPorts" value="2666,2669"/>
    <parameter name="HttpPort" value="3222"/>
    <dependOn name="LoggerService"/>
    <dependOn name="PreRequirementsService"/>
  </service>
  <service name="HandshakeService" assemblyName="UIP.GrpcChannel.Server"
    className="UIP.GrpcChannel.Server.Service.HandshakeService">
    <parameter name="Address" value="localhost"/>
  </service>
</configuration>
```

```

<parameter name="Port" value="2667"/>
<parameter name="GrpcHostManagerName" value="GrpcHostManager"/>
<parameter name="ExternalAddresses" value=""/>
<dependOn name="GrpcHostManager"/>
<dependOn name="LoggerService"/>
<dependOn name="PreRequirementsService"/>
</service>
</configuration>

```

Выше описана конфигурация двух сервисов внутри Сервера:

- GrpcHostManager – сервис определяющий параметры соединения к Серверу для подключения и обеспечивает взаимодействие с Web-сервером
- HandshakeService – сервис «рукопожатия» предоставляет публичный адрес на который соединяется Web-сервер, после чего получает уже адрес обмена данными

Актуальные параметры GrpcHostManager:

- ServiceUrl – адрес по которому может доступен Сервер для обмена данными
- GrpcPorts – порты, через запятую, которые будут добавляться к адресу в ServiceUrl при отправке данных о подключении после рукопожатия

Актуальные параметры HandshakeService:

- Address – сетевой адрес компьютера для публикации, актуально заполнять если на ЭВМ с запускаемым Сервером более одного сетевого интерфейса
- Port – порт по которому Web-сервер сможет запросить адрес для обмена данными

Конфигурация Web-сервера находится в файле «{папка Web-сервера\appsettings.json}», который выглядит так:

```

{
  "ServerChannel": {
    "handshakeAddress": "localhost",
    "handshakePort": 2667
  },

```

```

"SystemService": {
  "systemServiceLogin": "systemService",
  "systemServicePassword": "tsAZMBTrd"
},
"SiteName": "Мониторинг ТС",
"Version": "2022.12.1",
"YandexSearch": {
  "YandexApiKey": ""
},
"TempFile": {
  "Path": "tempFiles"
},
"NLog": {
  "Targets": [
    {
      "Name": "AllLog",
      "FileName": "C:/Transport/Logs/WebLogs/Transport/${shortdate}.txt",
      "Layout": "${date:format=HH\\:mm\\:ss\\.fff} ${message}
${exception:format=tostring}"
    }
  ],
  "Rules": [
    {
      "Name": "nlog",
      "Level": "Error",
      "WriteTo": "AllLog"
    },
    {
      "Name": "nlog",
      "Level": "Warn",
      "WriteTo": "AllLog"
    },
    {
      "Name": "nlog",
      "Level": "Info",
      "WriteTo": "AllLog"
    },
    {
      "Name": "nlog",
      "Level": "Debug",

```

```

    "WriteTo": "AllLog"
  }
]
},
"Logging": {
  "LogLevel": {
    "Default": "Information",
    "Microsoft": "Warning",
    "Microsoft.Hosting.Lifetime": "Information"
  }
},
"MassMedia": {
  "ReportsFolder": "C:\\Transport\\PublicReports\\MassMedia"
},
"GeoProxy": {
  "BasicPassword": ""
},
"AllowedHosts": "*"
}

```

Параметры соединения с Сервером содержатся в секции «ServiceChannel»:

- handshakeAddress – адрес Сервера
- handshakePort – порт HandshakeService Сервера

Следует также учитывать параметр «FileName» секции «Nlog», в нём указывается путь к файлу(ам) журналов Web-Сервера. Нужно указать актуальный путь с учётом используемой ОС и наличия дисков. Если указать путь к конкретному файлу, будет записываться один файл журналов, если заменить только часть «C:/Transport/Logs/WebLogs/» то на каждый день работы Web-сервера будет формироваться отдельный файл.