

ПрограмБанк.Интеграции программный комплекс

Руководство по настройке и установке

СОДЕРЖАНИЕ

1	Технические требования к серверу	3
2	Установка приложений	4
2.1	Установка Java (на примере OpenJDK от Bellsoft)	4
2.2	Установка PayaraServer	5
2.3	СУБД.....	7
2.4	Установка КриптоПроJCP.....	8
2.4.1	Подготовка дистрибутива	8
2.4.2	Установка КриптоПро JCP	9
3	Настройка сервера приложений.....	11
3.1	JVM Options	11
3.2	Источники данных	12
3.3	JMS Resources	15
4	Настройка обмена со СМЭВ	17
4.1	Настройка КриптоПро JCP	17
4.2	Настройка файла конфигурации системы «ПрограмБанк.Интеграции»	20
5	Установка программного комплекса «ПрограмБанк.Интеграции»	21
6	Проверка работоспособности сервисов	22
	Приложение 1. Перечень зависимостей (справочно).....	23
	Лист регистрации изменений.....	24

1 Технические требования к серверу

Процессор	Минимальные: Intel® Core i3 Рекомендуемые: Intel® Xeon E5 или выше или другие под соответствующую архитектуру
Оперативная память	Минимальные: 8 ГБ Рекомендуемые 16 ГБ
Жесткий диск	Объем от 200 ГБ
Сервер приложений	PayaraServer 5 (версии 5.2021.1 или выше)
Java	OpenJDK 8 (не ниже 8.282)
СУБД	PostgreSQL 12 или выше
Операционная система	Должна удовлетворять требованиям совместимости с сервером приложений и СУБД

2 Установка приложений

Для работы программного комплекса «ПрограмБанк.Интеграции» необходимо следующее стороннее программное обеспечение:

1. OpenJDK 8
2. Payara Server
3. СУБД (может быть установлена на другом сервере)
4. КриптоПро JCP

2.1 Установка Java (на примере OpenJDK от Bellsoft)

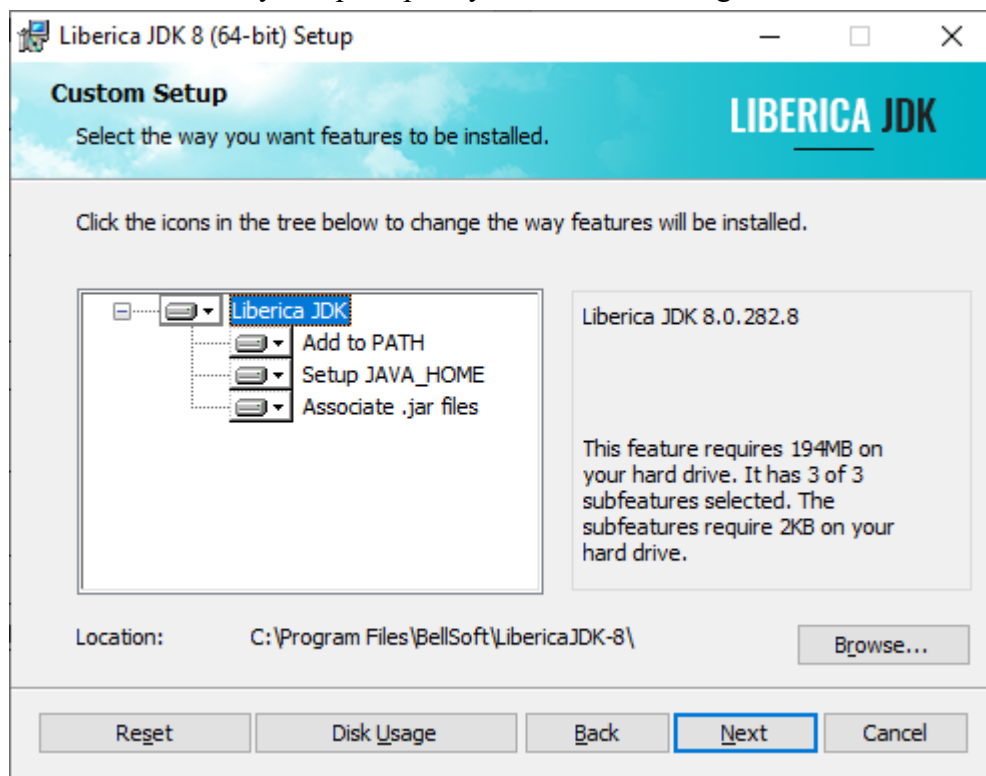
При установке Java следует:

1. Скачать дистрибутив по ссылке <https://bell-sw.com/pages/downloads/#/java-8-lts>.

ВНИМАНИЕ!

Устанавливать OpenJDK 8 версии, не ниже 8.282.

2. Установить в папку, например по умолчанию C:\Program Files\BellSoft\LibericaJDK-8\.

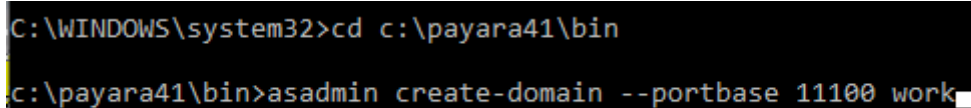


2.2 Установка PayaraServer

Для установки PayaraServer необходимо выполнить следующие действия:

1. Скачать версию PayaraServer 5 (или build выше к версии PayaraServer 5) <https://search.maven.org/artifact/fish.payara.distributions/payara/5.2021.1/zip>
2. Распаковать в папку на диске c:\. Название папки значение не имеет.
3. Если установлено несколько версий JDK, то необходимо в файле asenv.conf, который находится по пути <Папка payara>\glassfish\config\ (если ОС - Windows, то файл asenv.bat, если Linux, то asenv.conf) указать путь до OpenJDK нужной версии. Например:


```
set AS_JAVA=C:\Program Files\BellSoft\LibericaJDK-8\
```
4. Создать новый домен, например, work. Запускать команду (можно через cmd или TotalCommander) из папки <Папка payara>\bin, например, c:\payara41\bin\



```
C:\WINDOWS\system32>cd c:\payara41\bin
c:\payara41\bin>asadmin create-domain --portbase 11100 work
```

рис. 1. Создание домена

```
asadmin create-domain --portbase 11100 work
```

где:

- work: – имя нового домена,
- ключ – portbase 11100 автоматически меняет все стандартные порты домена.

Стандартные порты по умолчанию: 4848 (консоль управления), 8080 (доступ через https), 8081 (доступ через https).

В результате выполнения данной команды, все порты будут начинаться со 111 и заканчиваться двумя стандартными цифрами порта (в данном случае 11148, 11180, 11181).

Создание домена с ключом portbase необходимо в случае если стандартные порты заняты другим доменом, приложения или иным ПО.

5. Ввести логин Администратора, после ввода пароля (при нажатии на клавишу **Enter**, по умолчанию, будет admin без пароля).

ПРИМЕЧАНИЕ

Без пароля не будет удаленного доступа к панели управления PayaraServer.

Итог выполнения команды будет таков:

```
d:\pb\payara5\bin>asadmin create-domain --portbase 11100 work
Enter admin user name [Enter to accept default "admin" / no password]>admin
Enter the admin password [Enter to accept default of no password]>
Enter the admin password again>
Using port 11148 for Admin.
Using port 11180 for HTTP Instance.
Using port 11176 for JMS.
Using port 11137 for IIOP.
Using port 11181 for HTTP_SSL.
Using port 11138 for IIOP_SSL.
Using port 11139 for IIOP_MUTUALAUTH.
Using port 11186 for JMX_ADMIN.
Using port 11166 for OSGI_SHELL.
Using port 11109 for JAVA_DEBUGGER.
Using port 11149 for Hazelcast DAS.
Using port 11159 for Hazelcast Start.
Distinguished Name of the self-signed X.509 Server Certificate is:
[CN=kenbook,OU=Payara,O=Payara Foundation,L=Great Malvern,ST=Worcestershire,C=UK]
Distinguished Name of the self-signed X.509 Server Certificate is:
[CN=kenbook-instance,OU=Payara,O=Payara Foundation,L=Great Malvern,ST=Worcestershire,C=UK]
Domain work created.
Domain work admin port is 11148.
Domain work admin user is "admin".
Command create-domain executed successfully.

d:\pb\payara5\bin>
```

рис. 2. Результат выполнения команды создания домена

6. Запустить созданный домен work (запустить также из папки <Папка payara>\bin\):

```
asadmin start-domain work
```

При успешном запуске отобразится следующая запись:

```
Command start-domain executed successfully.

d:\pb\payara5\bin>
```

рис. 3. Успешное выполнение запуска домена

7. Далее зайти через веб-браузер для администрирования Payara по ссылке <http://localhost:11148> (для удаленного подключения вместо localhost необходимо указать IP адрес или DNS имя компьютера, где установлен сервер приложений Payara) и ввести в соответствующие поля **Логин** и **Пароль** администратора, указанные при создании домена:

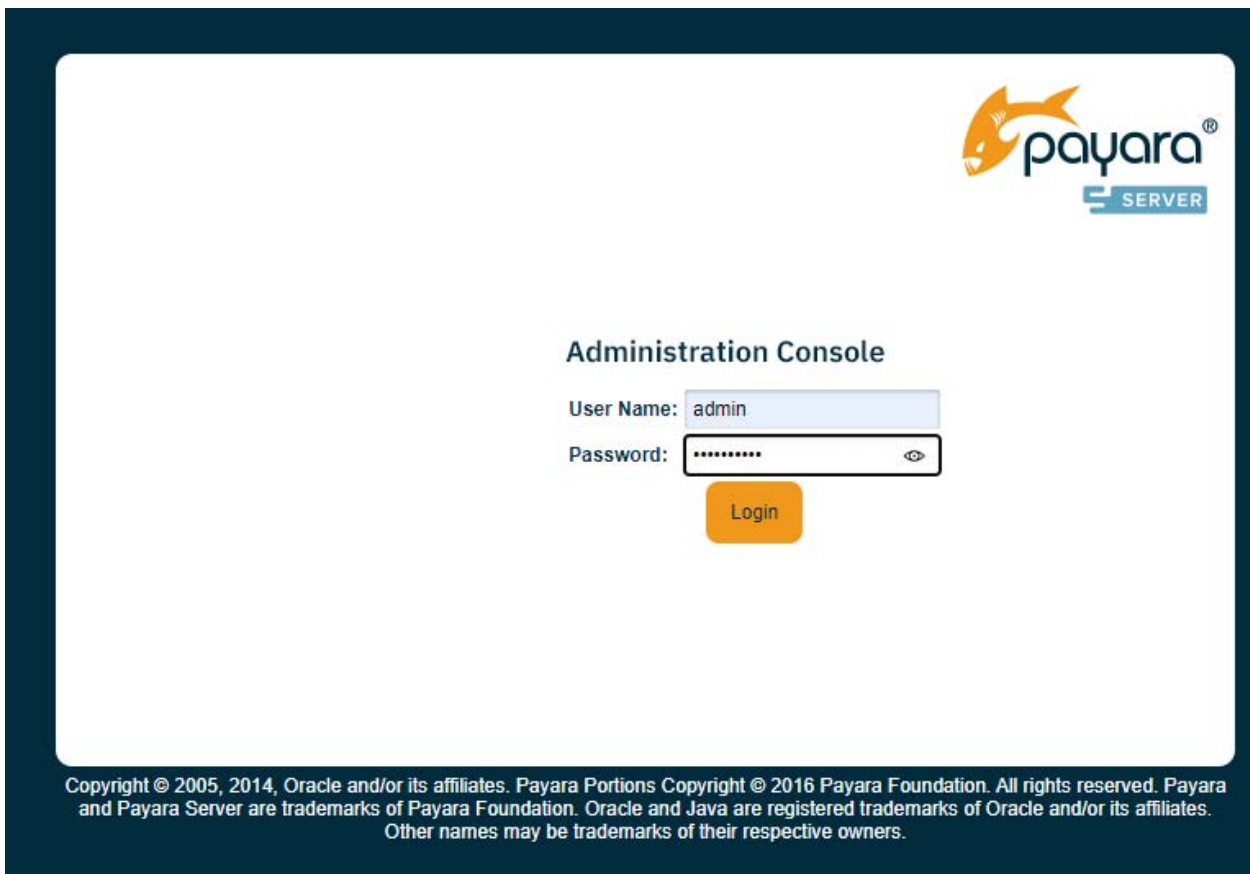


рис. 4. Вход через веб-браузер для администрирования Payara

2.3 СУБД

Для системы требуется взаимодействие с ДВУМЯ источниками данных.

В роли СУБД рекомендуется использование PostgreSQL.

Для общего случая настройки необходимо:

1. Создать ДВЕ базы данных. Одна с именем SMEV другая – DIRECT.
8. В базе данных DIRECT необходимо создать 4 пустые схемы: prbank; prbank_fias; prbank_hs; prbank_fms;

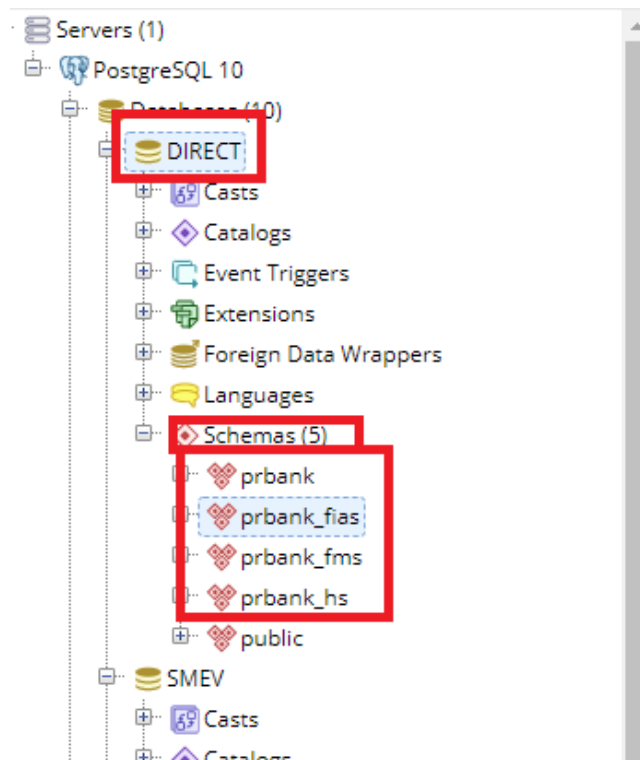


рис. 5. Созданные схемы в базе данных

9. Скачать JDBC драйвер по нижеуказанной ссылке:
<https://jdbc.postgresql.org/download.html>
10. Разархивировать скачанный драйвер в папку
<Папка Payara>\glassfish\domains\<имядомена>\lib

В итоге в папке lib должны находиться несколько пустых папок, которые формируются при создании домена и JDBC драйвер.

2.4 Установка КриптоПро JCP

2.4.1 Подготовка дистрибутива

Для подготовки дистрибутива необходимо выполнить следующее:

1. Скачать дистрибутив КриптоПро JCP с сайта:
<https://www.cryptopro.ru/products/csp/jcp/downloads>
2. Распаковать архив в папку на сервере (в данном примере C:\cryptopro).
| ВАЖНО!
| Полный путь к папке не должен содержать кириллические символы.
3. Найти JRE, которое использует сервер приложений, например
C:\Program Files\BellSoft\LibericaJDK-8\jre\lib\ext\
Далее этот путь используется неоднократно, обозначим его коротко <JRE>.
4. Скопировать библиотеки из C:\cryptopro\dependencies в каталог <JRE>\lib\ext
 - serializer.jar,
 - commons-logging.jar,
 - xalan.jar,
 - xmlsec.jar

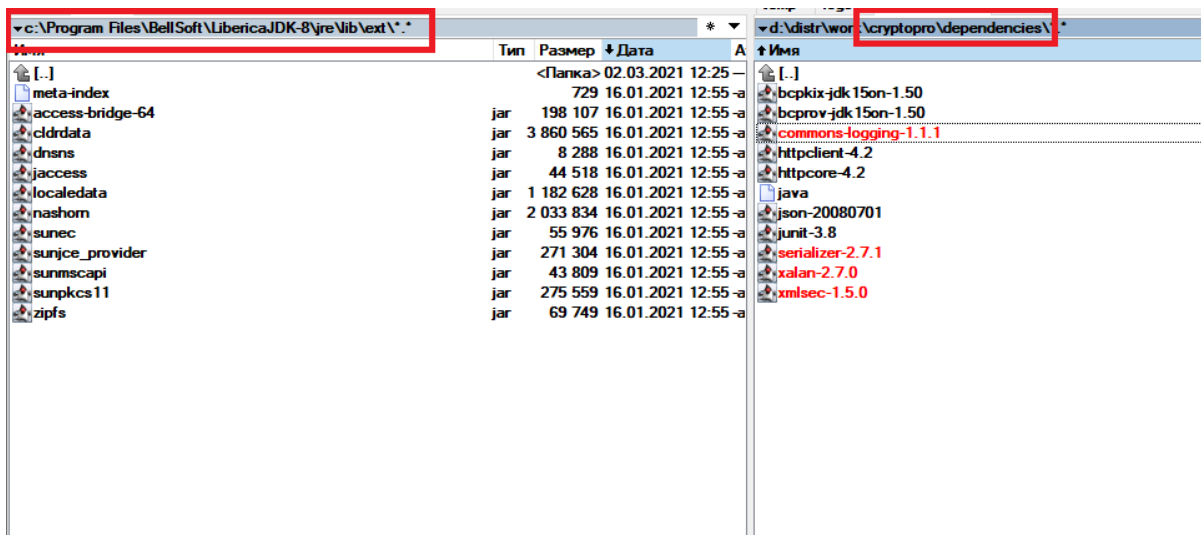


рис. 6. Копирование в каталог <JRE>\lib\ext

5. Выполнить инструкции по подготовке java к установке КриптоПро JCP: <https://support.cryptopro.ru/index.php?/Knowledgebase/Article/View/44/6/snjatie-ehkspornykh-ogrnichenijj>

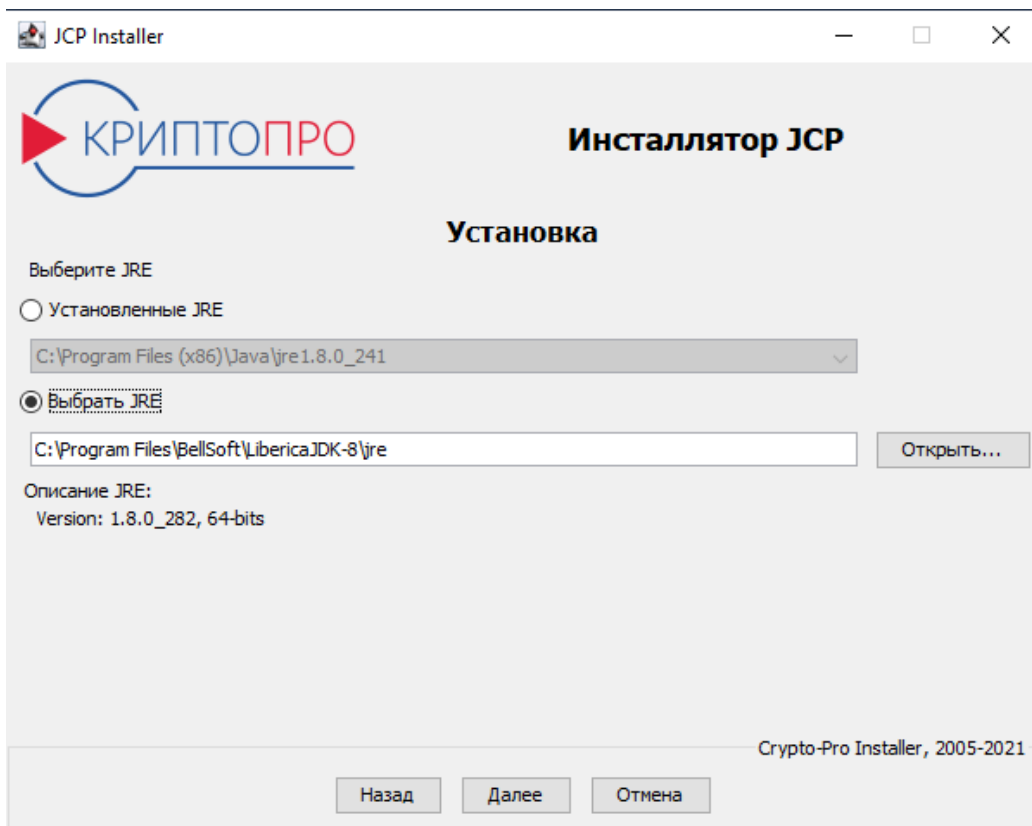
2.4.2 Установка КриптоПро JCP

ВНИМАНИЕ!

Установка должна производиться администратором.

Запустить приложения setup.exe от имени администратора. После выбора языка и действия (установка/удаление) будет предложено указать, в какой JRE будет производиться настройка, какие модули следует установить. Необходимо выбрать

C:\Program Files\BellSoft\LibericaJDK-8\jre:



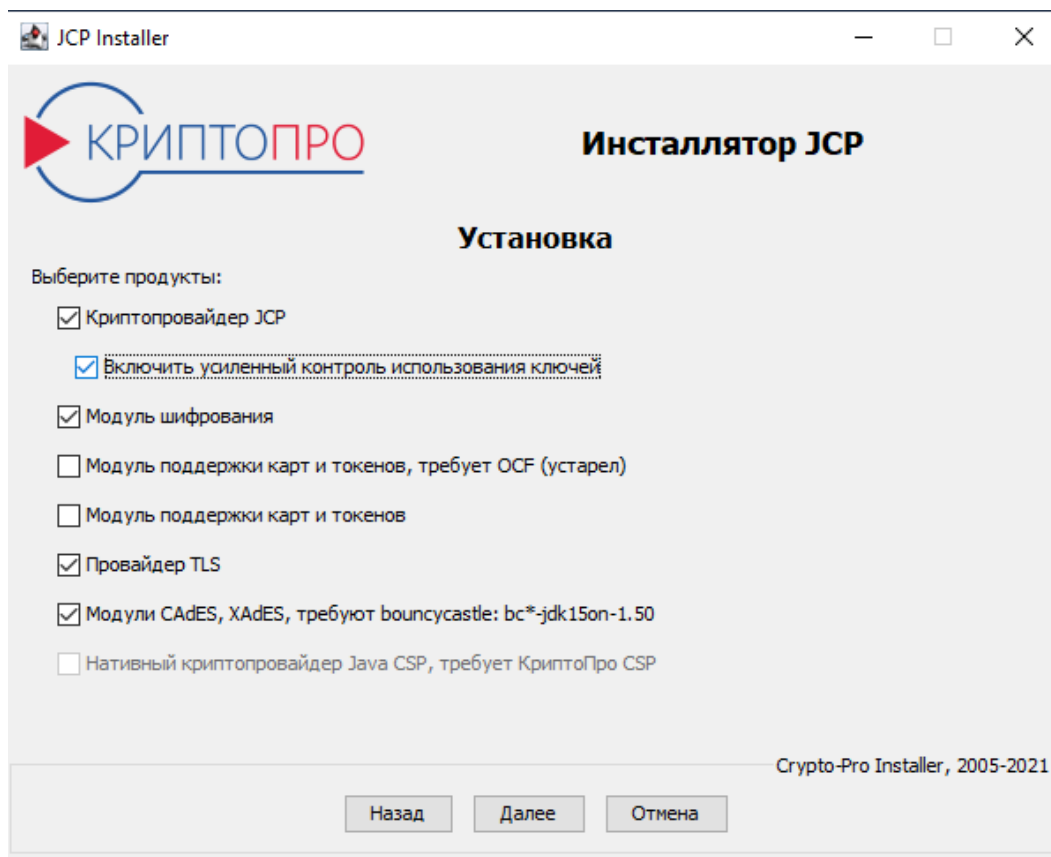


рис. 7. Выбор JRE и выбор модулей для установки

В случае успешного выполнения установки автоматически сформируется сообщение: «Установка пакетов завершена успешно».

После установки рекомендуется перезагрузка сервера. В соответствии с лицензионными ограничениями компании КриптоПро, на пакет КриптоПро JCP дается демонстрационный доступ длительностью 3 месяца с даты первой установки.

3 Настройка сервера приложений

Настройки в Payara для системы «ПрограмБанк.Интеграции»

В файле <JRE>/lib/security/java.security удалить 2 строки:

```
ssl.SocketFactory.provider=ru.CryptoPro.ssl.SSLSocketFactoryImpl  
  
ssl.ServerSocketFactory.provider=ru.CryptoPro.ssl.SSLServerSocketFactoryImpl
```

В следующих строках:

```
ssl.KeyManagerFactory.algorithm=GostX509  
ssl.TrustManagerFactory.algorithm=GostX509
```

изменить предоставленные КриптоПро JCP значения на значения по умолчанию:

```
ssl.KeyManagerFactory.algorithm= SunX509  
ssl.TrustManagerFactory.algorithm= PKIX
```

Далее, из командой строки необходимо осуществить старт домена сервера приложений и выполнить вход через веб браузер в интерфейс администратора.

Выполнить следующие настройки:

- JVM Options
- Источники данных
- JMS Resources

3.1 JVM Options

На вкладке **JVM Options** (последовательно выбирается *Configurations* → *server-config* → *JVM Settings* → *JVM Options*) необходимо изменить ключи, которые приводятся на рисунке ниже:

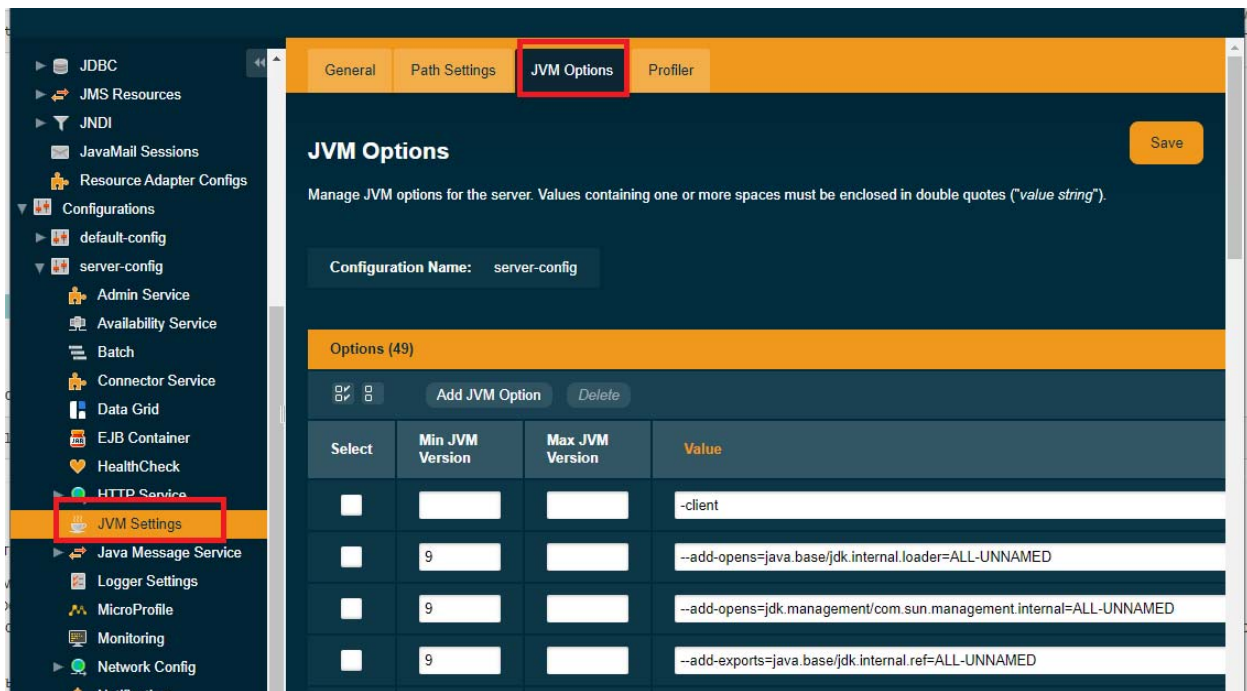


рис. 8. Вкладка JVM Options

Если ключа нет, то его следует добавить по кнопке **Add JVM Option** и в поле **value** его прописать.

- Увеличить максимальный размер памяти доступный серверу приложений, не менее 2Гб:

```
-Xmx2048m
```

- Для работы КриптоПро JCP:

```
-DUseSunHttpHandler=true
```

- Для обеспечения безопасности и сохранения значений некоторых ключей при обновлении приложения можно создать альтернативный файл конфигурации, чтобы не корректировать перед каждым обновлением тот, что лежит в jsmev*.ear. Добавить в системном свойстве **server-config-JVM Settings** на вкладке **JVM Options**:

```
-Dru.prbank.jsmev.config.location=<имя Вашего файла конфигурации (полное, с путем)>
```

- Добавить свойство `classloading.delegate=false`. Настраивается в **JVM-settings** на вкладке **JVM Options**:

```
-Dfish.payara.classloading.delegate=false
```

- Удалить ключ:

```
-XX:+UseOpenJSSE
```

3.2 Источники данных

Необходимо настроить два JDBC ConnectionPools в портале управления домена (пункт меню **Resources** → **JDBC** → **JDBC ConnectionPools**).

Прежде всего, следует создать пул с идентификатором **jsmev**:

В качестве **Resource Type** указать `java.sql.ConnectionPoolDataSource`.

В поле **DatabaseDriverVendor**, в качестве вендора указать PostgreSQL, далее нажать на кнопку **NEXT**. Значение в поле **DatasourceClassname** определится автоматически:

New JDBC Connection Pool (Step 2 of 2) Previous Finish Cancel

Identify the general settings for the connection pool. Datasource Classname or Driver Classname must be specified for the connection pool. * Indicates required field

General Settings

Pool Name: jsmev

Resource Type: javax.sql.ConnectionPoolDataSource

Database Driver Vendor: Postgresql

Datasource Classname: org.postgresql.ds.PGConnectionPoolDataSource

Driver Classname:

Ping: Enabled
When enabled, the pool is pinged during creation or reconfiguration to identify and warn of any erroneous values for its attributes

Description:

рис. 9. Настройка JDBC Connection Pools

На вкладке **General** в разделе **PoolSettings** указать параметр *IdleTimeout = 3000*.

Pool Settings

Initial and Minimum Pool Size: 8 Connections
Minimum and initial number of connections maintained in the pool

Maximum Pool Size: 32 Connections
Maximum number of connections that can be created to satisfy client requests

Pool Resize Quantity: 2 Connections
Number of connections to be removed when pool idle timeout expires

Idle Timeout: 3000 Seconds
Maximum time that connection can remain idle in the pool

Max Wait Time: 60000 Milliseconds
Amount of time caller waits before connection timeout is sent

рис. 10. Раздел Pool Settings, параметр IdleTimeout = 3000

В разделе **Additional Properties** необходимо оставить только следующие свойства:

Additional Properties (5)			
Select	Name	Value	Description
<input type="checkbox"/>	DatabaseName	SMEV1	
<input type="checkbox"/>	PortNumber	5432	
<input type="checkbox"/>	User	postgres	
<input type="checkbox"/>	Password	1	
<input type="checkbox"/>	ServerName	localhost	

рис. 11. Пример настройки пула соединения с именем jsmev на основе PostgreSQL

- ServerName – адрес сервера
- PortNumber – порт
- Password – пароль к СУБД
- User – пользователь СУБД
- DatabaseName – Имя БД, созданной в п. «СУБД»SMEV

Нажать на кнопку **Finish**.

На вкладке **Advanced** в разделе **ConnectionValidation** установить признак **Required**, а в поле **ValidationMethod** указать **auto-commit**.

Далее перейти в свойства созданного пула и нажать на **Ping** для проверки соединения с СУБД (рис. 12).

Пример успешного прохождения Ping приводится на следующем рисунке:

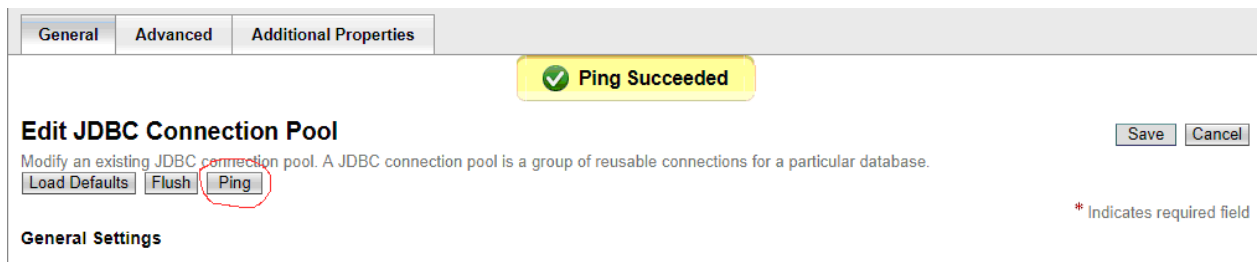


рис. 12. Успешное прохождение Ping

В иных случаях будет означать, что неверно настроено соединение с БД.

Аналогично, следует создать второй пул с идентификатором **directories**.

ВНИМАНИЕ!

Для пула **directories** в поле **DatabaseName** нужно указывать вторую БД из п «СУБД». DIRECT.

Необходимо настроить два JDBC Resources в портале управления домена (пункт меню **Resources** → **JDBC** → **JDBC Resources**).

Ресурсы создаются с именами без изменений.

1. Имя первого ресурса **jdbc/pbJSmev**, указать в качестве пула соединение, созданное на предшествующем шаге с именем **jsmev**. Пример:

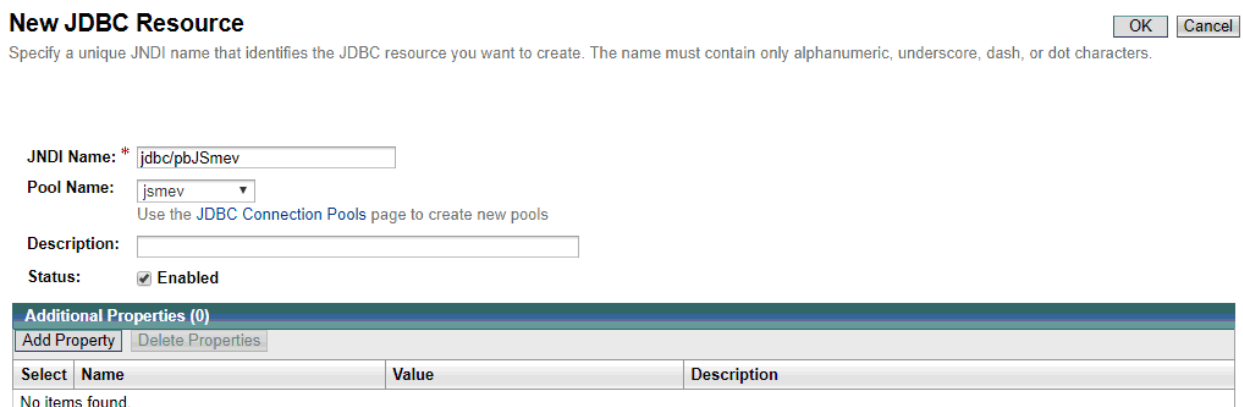


рис. 13. Ресурс jdbc/pbJSmev с именем пула jsmev

2. Имя второго ресурса **jdbc/directories**, в списке **PoolName** выбрать созданный пул с именем **directories**.

3.3 JMS Resources

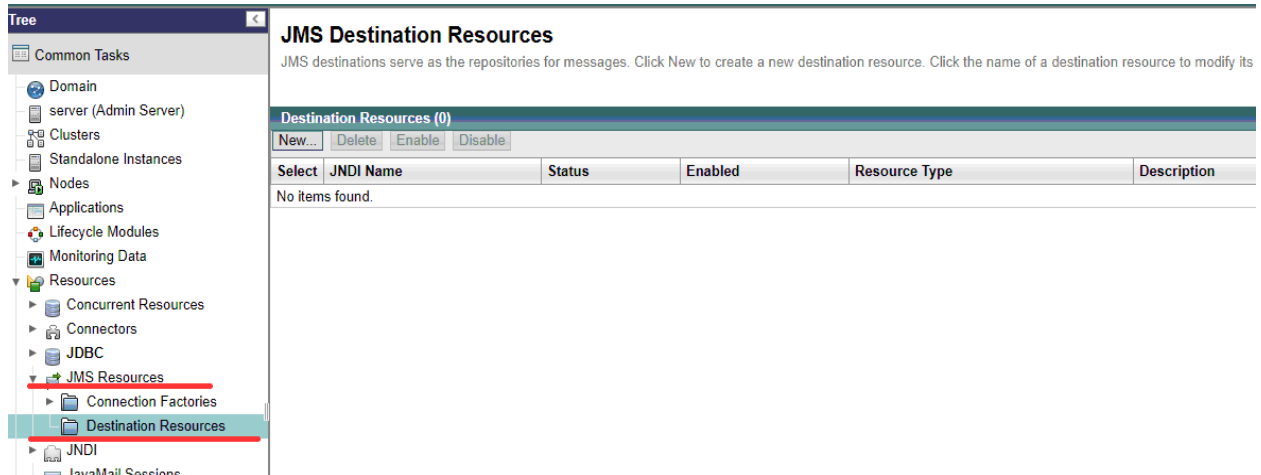


рис. 14. Настройка JMS Destination Resources

Для настройки **JMS Resources** необходимо следующее:

1. Настроить следующие JMS Destination Resources:

JNDI Name	Physical Destination Name	Resource Type
jms/GisGkhErrorQueue	GisGkhErrorQueue	javax.jms.Queue
jms/GisGkhErrorQueue2	GisGkhErrorQueue2	javax.jms.Queue
jms/GisGkhQueue	GisGkhQueue	javax.jms.Queue
jms/GisGkhQueue2	GisGkhQueue2	javax.jms.Queue
jms/UfebsFilesErrorQueue	UfebsFilesErrorQueue	javax.jms.Queue
jms/UfebsFilesQueue	UfebsFilesQueue	javax.jms.Queue
mq.sys.dmq	dmq	javax.jms.Queue
jms/Smev3Queue	Smev3Queue	javax.jms.Queue
jms/Smev3ErrorQueue	Smev3ErrorQueue	javax.jms.Queue

2. Настроить следующие JMS Connection Factories:

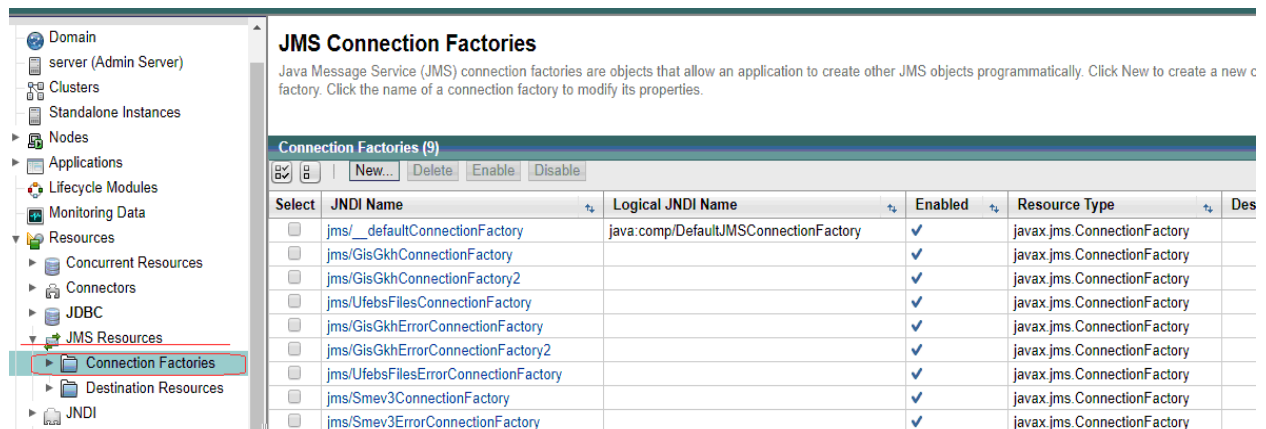


рис. 15. Настройка JMS Connection Factories

ПрограмБанк.Интеграции

Pool Name	Resource Type
jms/GisGkhConnectionFactory	javax.jms.ConnectionFactory
jms/GisGkhConnectionFactory2	javax.jms.ConnectionFactory
jms/UfebsFilesConnectionFactory	javax.jms.ConnectionFactory
jms/GisGkhErrorConnectionFactory	javax.jms.ConnectionFactory
jms/GisGkhErrorConnectionFactory2	javax.jms.ConnectionFactory
jms/UfebsFilesErrorConnectionFactory	javax.jms.ConnectionFactory
jms/Smev3ConnectionFactory	javax.jms.ConnectionFactory
jms/Smev3ErrorConnectionFactory	javax.jms.ConnectionFactory

3. После произведенных изменений требуется перезагрузить сервер приложений.

Из командной строки из папки <Папка payara>\bin:

```
asadmin stop-domain <имя домена>
```

После того как произойдет остановка домена выполнить запуск домена:

```
asadmin start-domain <имя домена>
```

или перезагрузить командой:

```
restart-domain <имя домена>
```


4 Настройка обмена со СМЭВ

4.1 Настройка КриптоПро JCP

Запуск панели КриптоПро JCP осуществляется через командную строку от имени администратора:

```
<JRE>java.exe -Dfile.encoding=cp866 ru.CryptoPro.JCP.ControlPane.MainControlPane
```

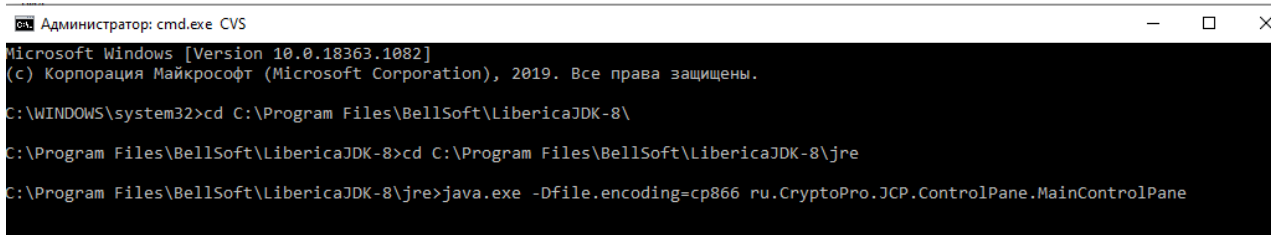


рис. 16. Запуск панели управления КриптоПроJCP

Настройка параметров

ВНИМАНИЕ!

Параметры КриптоПро JCP хранятся в разрезе пользователей операционной системы, поэтому указанные в этом пункте действия следует выполнить под каждым пользователем, от имени которого будет осуществляться запуск сервера приложения.

На вкладке **Настройки сервера** (TLSSettings) снять флажки в полях:

- Проверять сертификаты цепочки на отзыв (Checkrevocationstatus of certificate chain)
- Загружать CRL по CRLDP сертификата (Load CRL by CRLDP)
- Загружать CRL из папки (Load CRL from directory)

Установить переключатель в положение:

- Разрешить небезопасное возобновление сессии и несоответствие RFC 5746(Allow unsafe regeneration and legacy hello messages):

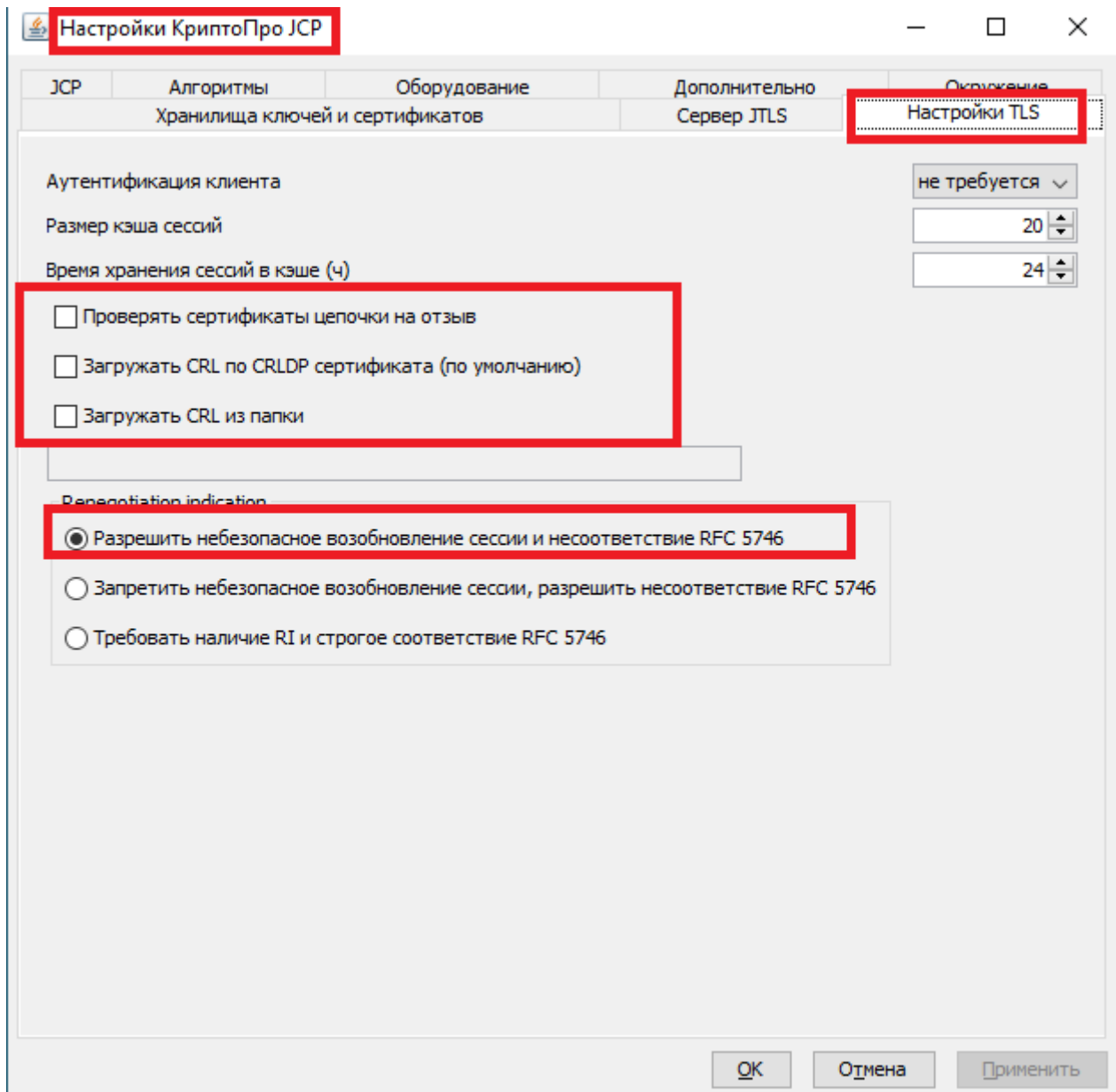


рис. 17 Настройка КриптоПро JCP

Дополнительно следует проверить, что у **каждого** пользователя, от имени которого будет осуществляться запуск сервера приложения, есть доступ к папке `JRE\lib\ext` и находящимся в ней JAR файлами КриптоПро JCP.

Директория для сохранения контейнеров ключей задается на вкладке **Оборудование** (Hardware).

ЗАМЕЧАНИЕ

Важно, чтобы полный путь к папке не содержал кириллические символы.

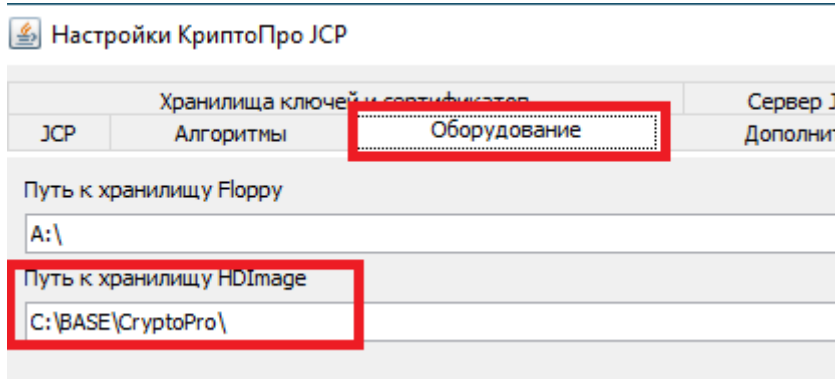


рис. 18. Вкладка «Оборудование»

Чтобы загрузить в хранилище контейнеров личную ключевую пару, необходимо в директорию, указанную, как путь к хранилищам HDImageStore, скопировать ключевой контейнер (папка *.000 с файлами *.key). Затем нужно обновить список по кнопке **Обновить**, расположенной внизу:

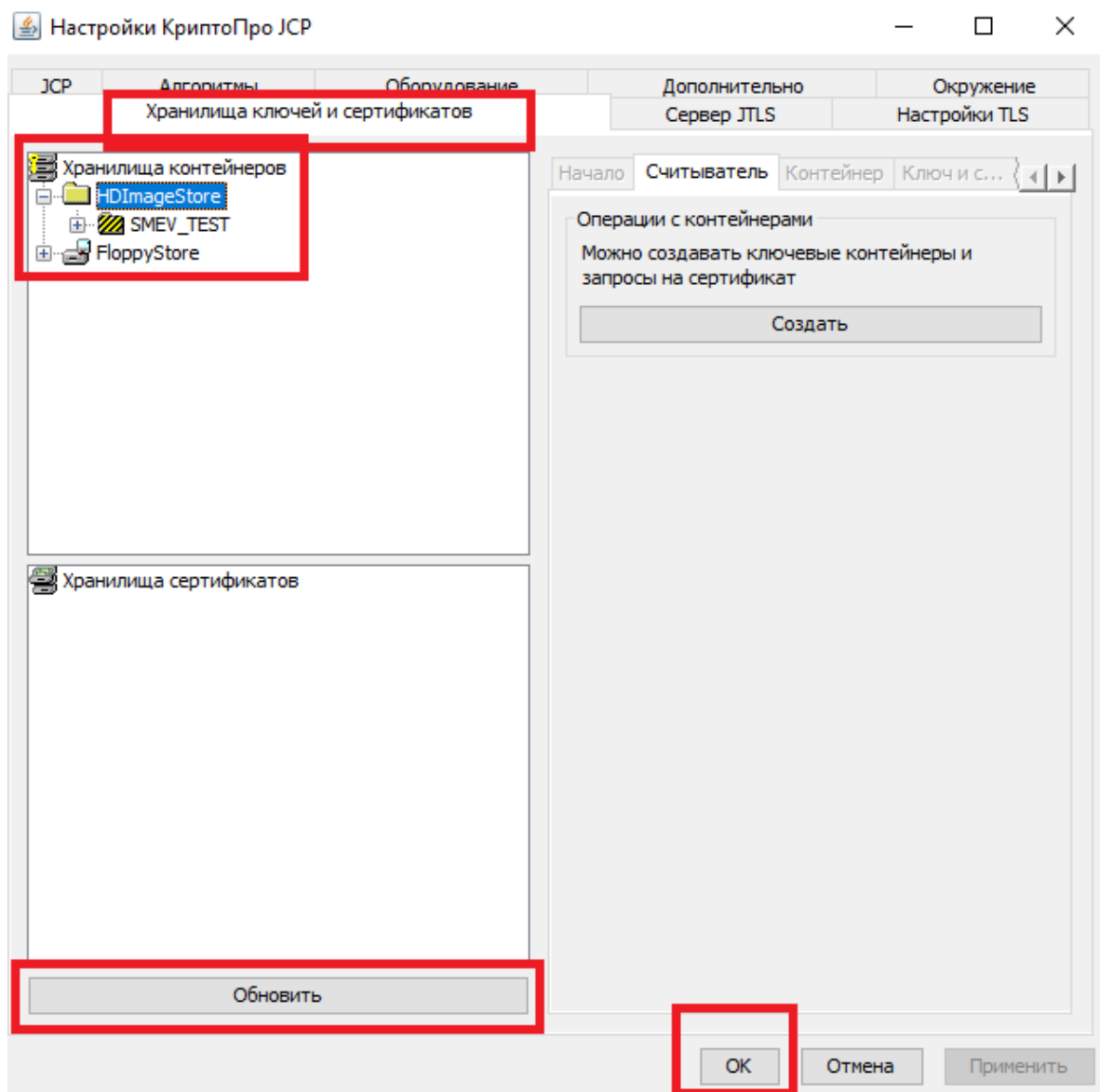


рис. 19 Хранилище ключей КриптоПро JCP

4.2 Настройка файла конфигурации системы «ПрограмБанк.Интеграции»

Корректировка файла конфигурации .config.

1. Файл конфигурации из дистрибутива необходимо настроить и скопировать в папку, путь к которому указан в настройке в пункте «JVM Options».

Если используется прокси для соединения с интернетом, то необходимо, изменить условие:

! Прокси

smev.proxy = *true*

smev.proxy.url = **Ошибка! Недопустимый объект гиперссылки.**

или:

11. Создать папку `c:/SMEV/Logs/`,

или

при необходимости изменить путь в настройке до папки с логами:

smev.jrn.log.path = *c:/SMEV/Logs/*

12. Задать имя контейнера с ключами и пароль (по умолчанию прописаны данные тестового контейнера и пароль к нему).

! имя контейнера КриптоПроJCP и пароль приватного ключа ЭЦП

Например:

abonent.gefest.eds.container =test

abonent.gefest.eds.password =1234567890

5 Установка программного комплекса «ПрограмБанк.Интеграции»

Для правильной работы программного комплекса перед выполнением команды **deploy** необходимо выполнить настройку конфигурации в файле **.config**.

Рекомендуется сделать свой собственный «альтернативный» файл конфигурации, который будет находиться в файловой системе ОС, и задать путь к нему в системной переменной - **Dru.prbank.jsmev.config.location** (см. п. «JVM Options»). Файл-пример находится в архиве **jsmev*.ear** в каталоге **jsmev-config-ejb-2.0-SNAPSHOT_jar\ru\prbank\jsmev**.

В «альтернативном» файле конфигурации можно указывать не все параметры. В этом случае, значения параметров, которые Вы не перекрыли в «своем» файле конфигурации будут зачитаны из файла внутри архива **jsmev*.ear**.

Для разворачивания программного комплекса в разделе **Applications** по кнопке **Deploy** выбирается каталог, где лежит файл **jsmev*.ear**. Далее используется кнопка **OK**:

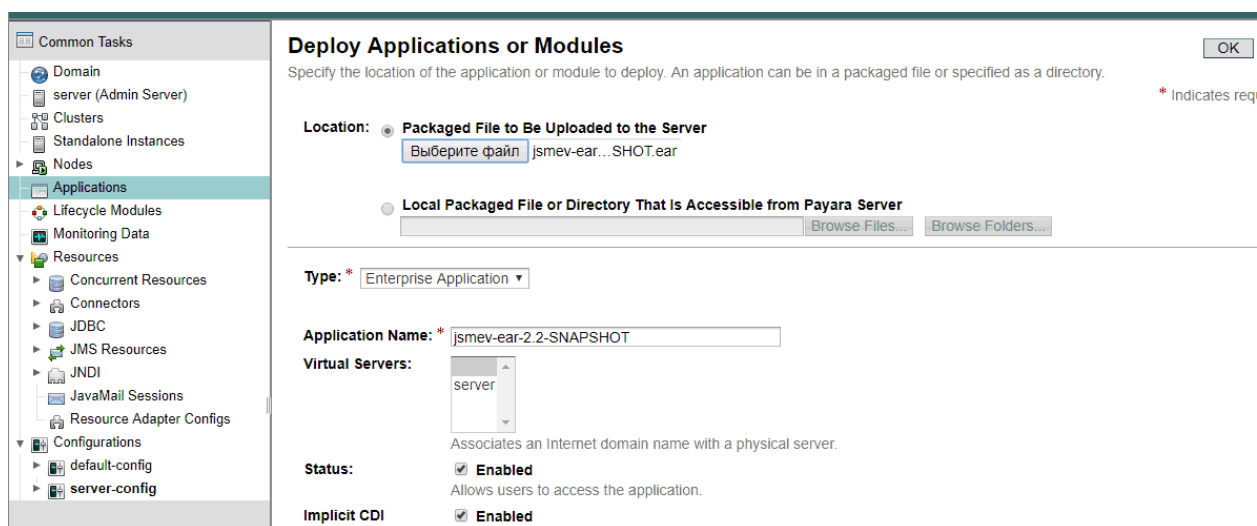


рис. 20. Выбор каталога

Имя установленного приложения появится в окне после успешного выполнения команды **Deploy**:

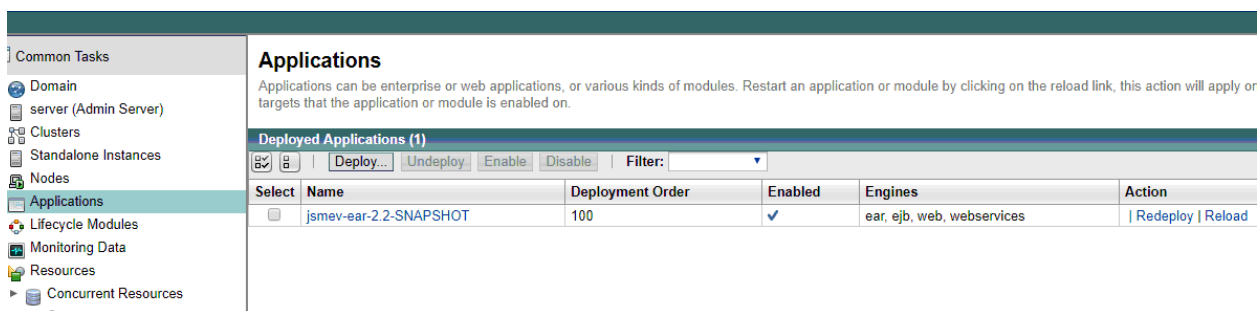


рис. 21. Результат установки приложения на сервере приложений

6 Проверка работоспособности сервисов

Проверить работоспособность сервиса: GET /gis-gkh/ping

Можно проверить, через веб-браузер, обратившись по адресу:

Ошибка! Недопустимый объект гиперссылки.

Здесь ip:port – адрес по которому установлена система «ПрограмБанк.Интеграции»/

Возвращает текущее время в формате уууу-ММ-ddНН:мм:ss.SSSSSS

Выглядит это так:

2020-11-06 17:56:19.000316

Если система запущена на текущем компьютере и использован порт 11180, то строка запуска будет следующей:

<http://localhost:11180/smev/rest/gis-gkh/ping>

Проверить работоспособность сервиса: GET /gis-gmp/ping

Аналогичным способом проверяется через веб-браузер

Ошибка! Недопустимый объект гиперссылки.

Возвращает текущее время в формате уууу-ММ-ddНН:мм:ss.SSSSSS

Если система запущена на текущем компьютере и использован порт 11180, то строка запуска будет следующей:

<http://localhost:11180/smev/rest/gis-gmp/ping>

Проверить работоспособность сервиса: GET /gis-gmp-3/ping

Аналогично

Ошибка! Недопустимый объект гиперссылки.

Возвращает текущее время в формате уууу-ММ-ddНН:мм:ss.SSSSSS

Если система запущена на текущем компьютере и использован порт 11180, то строка запуска будет следующей:

<http://localhost:11180/smev/rest/gis-gmp-3/ping>

Проверить работоспособность сервиса: GET /gis-gkh-3/ping

Аналогично

Ошибка! Недопустимый объект гиперссылки.

Возвращает текущее время в формате уууу-ММ-ddНН:мм:ss.SSSSSS

Если система запущена на текущем компьютере и использован порт 11180, то строка запуска будет следующей:

<http://localhost:11180/smev/rest/gis-gkh-3/ping>

Приложение 1. Перечень зависимостей (справочно)

Ниже приведен список зависимостей (библиотеки и др.), которые могут требоваться для работы приложения. Специальные действия (кроме указанных выше) для установки не требуются. Список приведен для справки.

Зависимости	Лицензия
https://mvnrepository.com/artifact/javax/javaee-api	CDDL
https://mvnrepository.com/artifact/org.apache.maven.plugins/maven-enforcer-plugin	Apache
https://mvnrepository.com/artifact/org.projectlombok/lombok	MIT
https://mvnrepository.com/artifact/org.apache.commons/commons-lang3	Apache
https://mvnrepository.com/artifact/org.eclipse.persistence/eclipselink	EDL, EPL
https://mvnrepository.com/artifact/commons-io/commons-io	Apache
https://mvnrepository.com/artifact/org.apache.maven.plugins/maven-checkstyle-plugin	Apache
https://mvnrepository.com/artifact/commons-beanutils/commons-beanutils	Apache
https://mvnrepository.com/artifact/org.apache.logging.log4j/log4j-api	Apache
https://mvnrepository.com/artifact/joda-time/joda-time	Apache
https://mvnrepository.com/artifact/org.apache.logging.log4j/log4j-core	Apache
https://mvnrepository.com/artifact/org.yaml/snakeyaml	Apache
https://mvnrepository.com/artifact/com.sun.xml.bind/jaxb-impl	EDL
https://mvnrepository.com/artifact/org.slf4j/slf4j-log4j12	MIT
https://mvnrepository.com/artifact/org.slf4j/slf4j-api	MIT
https://mvnrepository.com/artifact/org.jamel.dbf/dbf-reader	Apache
https://mvnrepository.com/artifact/javax/javaee-api	CDDL
https://mvnrepository.com/artifact/io.swagger/swagger-core	Apache
https://mvnrepository.com/artifact/org.codehaus.mojo/versions-maven-plugin	Apache
https://mvnrepository.com/artifact/org.apache.maven.plugins/maven-toolchains-plugin	Apache
https://mvnrepository.com/artifact/org.jruby/jruby-complete	EPL
https://mvnrepository.com/artifact/org.asciidoctor/asciidoctorj-pdf	Apache
https://mvnrepository.com/artifact/com.github.kongchen/swagger-maven-plugin	Apache
https://maven.apache.org/plugins/maven-ejb-plugin/	Apache
https://mvnrepository.com/artifact/org.apache.maven.plugins/maven-war-plugin	Apache
https://mvnrepository.com/artifact/org.asciidoctor/asciidoctor-maven-plugin	Apache
https://mvnrepository.com/artifact/org.asciidoctor/asciidoctorj	Apache
https://mvnrepository.com/artifact/org.apache.maven.plugins/maven-javadoc-plugin	Apache
https://mvnrepository.com/artifact/org.apache.maven.plugins/maven-resources-plugin	Apache
https://mvnrepository.com/artifact/com.edugility/jpa-maven-plugin	MIT

