ООО «ПАЛЛАДА»

**РУКОВОДСТВО АДМИНИСТРАТОРА**

«ИАС-ПОИСК»

МОСКВА

2020

Оглавление

[1. Описание системы «ИАС-Поиск» 3](#_Toc35264247)

[1.1 Расположение системы «ИАС-Поиск» на сервере 3](#_Toc35264248)

[1.2 Настройка docker-compose 3](#_Toc35264249)

[1.3 Настройка вэб сервера Apache 2. 7](#_Toc35264250)

[2. Настройка брокера сообщений Apache ActiveMQ. 9](#_Toc35264251)

[2.1 Конфигурационные файлы 9](#_Toc35264252)

[ Основной конфигурационный файл activemq.xml 9](#_Toc35264253)

[ Конфигурационный файл маршрутизации camel.xml 10](#_Toc35264254)

[ Сервлет jetty.xml 11](#_Toc35264255)

1. **Описание системы «ИАС-Поиск»**
	1. **Расположение системы «ИАС-Поиск» на сервере**

Программа «**ИАС-Поиск**» располагается в docker-контейнере (poiskavia), адрес которого по умолчанию 172.18.0.2.

Создание, запуск и настройка контейнера производится средствами docker-compose.

* 1. **Настройка docker-compose**

Конфигурационный файл docker-compose:

version: "3"

services:

 poiskavia:

 image: poiskavia:1.0.0

 stop\_signal: SIGRTMIN+3

 shm\_size: 1g

 cap\_add:

 - SYS\_ADMIN

 tmpfs:

 - /tmp

 - /run

 volumes:

 - /sys/fs/cgroup:/sys/fs/cgroup:ro

 - poiskaviadb:/var/lib/pgsql

 - ${data}:/data

 ports:

 - "80:80"

volumes:

 poiskaviadb:

Подключение к контейнеру производится командой:

docker exec -ti poiskavia bash

Расположение компонентов программы «ИАС-Поиск» в контейнере:

* Пользовательская часть системы:

/srv/www/htdocs

* Серверная часть системы:

/srv/xsp/

Конфигурационный файл пользовательской части:

/srv/www/htdocs/config.js

function main(boot) {

 var jsBase = boot.parseURI(boot.combineURI(

 document.location.toString(),

 boot.expandURI("~/js"))).path;

 function local(name) {

 return [jsBase, name].join("/");

 }

 return {

 title: "Поиск-Авиа",

 version: "предпоказ",

 style: "tundra",

 dojo: {

 base: "/ajax/dojo-release-1.13.0",

 main: "dojo.js",

 config: {

 paseOnLoad: false,

 async: true,

 map: {

 '\*': {

 core: 'implab',

 ol3: 'djol'

 }

 },

 packages: [{

 name: "implab",

 location: "/ajax/implab-src/implab",

 }, {

 name: "utest",

 location: "/ajax/implab-src/utest",

 }, {

 name: "poisk",

 location: local("poiskavia"),

 }, {

 name: "app",

 location: local("app"),

 }, {

 name: "modules",

 location: local("modules")

 }, {

 name: "mock",

 location: local("mock")

 }, {

 name: "djol",

 location: "/ajax/implab-src/djol"

 }, {

 name: "config",

 location: local("config")

 }, {

 name: "ol",

 location: "/ajax/implab-src/djol",

 main: "ol-stub"

 }, {

 name: "moment",

 location: "/ajax/moment",

 main: "moment-with-locales.min"

 }, {

 name: "timezone",

 location: "/ajax/moment",

 main: "moment-timezone-with-data.min"

 }, {

 name: "gridx",

 location: "/ajax/gridx"

 }, {

 name: "turf",

 location: "/ajax/turf",

 main: "turf"

 }, {

 name: "pdfmake",

 location: "/ajax/pdfmake"

 }]

 }

 },

 js: [

 "/ajax/proj4js/proj4.js",

 "/ajax/ol/4.5.0/ol-debug.js"

 ],

 trace: {

 level: 4,

 match: /^(modules|poisk|app|djol|implab\/dom\/inject)/

 },

 app: "poisk",

 // mixin to config/app.js

 services: {

 wms: boot.expandURI("~/geoserver/poiskavia/wms"),

 wfs: boot.expandURI("~/geoserver/poiskavia/wfs"),

 amq: boot.expandURI("~/app/messaging"),

 appBase: boot.expandURI("~/app/"),

 units: 'nautical', // nautical, metric

 baseLayers: [{

 $type: "ol3/TileLayer",

 params: {

 source: {

 $factory: function (ol) {

 return new ol.source.OSM();

 },

 params: {

 $type: "ol3"

 }

 },

 layerType: "base",

 label: "OSM"

 }

 }, {

 $type: "ol3/TileLayer",

 params: {

 source: {

 $factory: function (ol) {

 return new ol.source.OSM({

 url: 'map/{z}/{x}/{y}.png'

 });

 },

 params: {

 $type: "ol3"

 }

 },

 layerType: "base",

 visible: false,

 label: "ОSM Россия"

 }

 }, {

 $type : "ol3/TileLayer",

 params : {

 source : {

 $type : "ol3/TileWMSSource",

 params : {

 params : {

 LAYERS : "ural\_map"

 },

 url : "/geoserver/poiskavia/wms"

 }

 },

 visible: false,

 label : "ural\_map"

 }

 }],

 mapCenter: [39, 55],

 mapProj: 'EPSG:3857',

 mapZoom: 4,

 },

 css: [

 "/ajax/ol/4.5.0/ol.css",

 "/ajax/dojo-release-1.13.0/dojox/layout/resources/ScrollPane.css",

 "~/css/global.css",

 "~/icons/geo/css/geo-font.css",

 "~/icons/geo/css/animation.css",

 "~/icons/awesome/css/font-awesome.css",

 "~/icons/ionicons/css/ionicons.css",

 "/ajax/gridx/resources/claro/Gridx.css"

 ],

 };

}

* 1. **Настройка вэб сервера Apache 2.**

Конфигурационный файл вэб сервера расположен по адресу:

/etc/apache2/httpd.conf

LoadModule ldap\_module /usr/lib64/apache2-prefork/mod\_ldap.so

LoadModule authnz\_ldap\_module /usr/lib64/apache2-prefork/mod\_authnz\_ldap.so

LoadModule actions\_module /usr/lib64/apache2-prefork/mod\_actions.so

LoadModule alias\_module /usr/lib64/apache2-prefork/mod\_alias.so

LoadModule auth\_basic\_module /usr/lib64/apache2-prefork/mod\_auth\_basic.so

LoadModule authz\_host\_module /usr/lib64/apache2-prefork/mod\_authz\_host.so

LoadModule authz\_core\_module /usr/lib64/apache2-prefork/mod\_authz\_core.so

LoadModule authz\_user\_module /usr/lib64/apache2-prefork/mod\_authz\_user.so

LoadModule cgi\_module /usr/lib64/apache2-prefork/mod\_cgi.so

LoadModule dir\_module /usr/lib64/apache2-prefork/mod\_dir.so

LoadModule env\_module /usr/lib64/apache2-prefork/mod\_env.so

LoadModule expires\_module /usr/lib64/apache2-prefork/mod\_expires.so

LoadModule log\_config\_module /usr/lib64/apache2-prefork/mod\_log\_config.so

LoadModule mime\_module /usr/lib64/apache2-prefork/mod\_mime.so

LoadModule negotiation\_module /usr/lib64/apache2-prefork/mod\_negotiation.so

LoadModule setenvif\_module /usr/lib64/apache2-prefork/mod\_setenvif.so

LoadModule rewrite\_module /usr/lib64/apache2-prefork/mod\_rewrite.so

LoadModule dav\_module /usr/lib64/apache2-prefork/mod\_dav.so

LoadModule dav\_fs\_module /usr/lib64/apache2-prefork/mod\_dav\_fs.so

LoadModule proxy\_module /usr/lib64/apache2-prefork/mod\_proxy.so

LoadModule proxy\_http\_module /usr/lib64/apache2-prefork/mod\_proxy\_http.so

LoadModule headers\_module /usr/lib64/apache2-prefork/mod\_headers.so

ServerName 127.0.0.1

DocumentRoot /srv/www/htdocs

Listen 0.0.0.0:80

PidFile httpd.pid

ErrorLog httpd.log

CustomLog httpd.log "%h %l %u %t \"%r\" %>s %b"

#LogLevel debug

Alias /ajax/implab-src/ /srv/www/htdocs/Implab/

Alias /ajax/ /srv/www/htdocs/ajax/

ProxyTimeout 1200

<Location /app>

 ProxyPreserveHost On

 ProxyPass http://localhost:8081/app

 ProxyPassReverse http://localhost:8081/app

</Location>

<Location /geoserver>

 ProxyPreserveHost On

 ProxyPass http://localhost:8080/geoserver

 ProxyPassReverse http://localhost:8080/geoserver

</Location>

<Location /api/amq>

 ProxyPreserveHost On

 RequestHeader set Authorization "Basic YWRtaW46YWRtaW4="

 ProxyPass http://localhost:8161/api/amq

 ProxyPassReverse http://localhost:8161/api/amq

</Location>

<Location /map>

 ProxyPreserveHost On

 ProxyPass http://osm-map/map/osm\_tiles/

 ProxyPassReverse http://osm-map/map/osm\_tiles/

</Location>

<Directory .>

 Options +FollowSymLinks

 Require all granted

</Directory>

1. **Настройка брокера сообщений Apache ActiveMQ.**

Брокер ActiveMQ установлен на хосте gkcps-host по адресу:

/opt/activemq

* 1. **Конфигурационные файлы**

Конфигурационные файлы ActiveMQ расположены по адресу:

 /opt/activemq/conf

* Основной конфигурационный файл **activemq.xml**

<beans

 xmlns="http://www.springframework.org/schema/beans"

 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

 xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd

 http://activemq.apache.org/schema/core http://activemq.apache.org/schema/core/activemq-core.xsd">

 <bean class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">

 <property name="locations">

 <value>file:${activemq.conf}/credentials.properties</value>

 </property>

 </bean>

<bean id="logQuery" class="io.fabric8.insight.log.log4j.Log4jLogQuery"

 lazy-init="false" scope="singleton"

 init-method="start" destroy-method="stop">

 </bean>

 <broker xmlns="http://activemq.apache.org/schema/core" brokerName="mq1" dataDirectory="${activemq.data}">

 <sslContext>

 <sslContext

 trustStore="ssl/truststore.jks" trustStorePassword="84sL^7^6"

 keyStore="ssl/keystore.jks" keyStorePassword="password"

 />

 </sslContext>

 <destinationPolicy>

 <policyMap>

 <policyEntries>

 <policyEntry topic=">" >

 <pendingMessageLimitStrategy>

 <constantPendingMessageLimitStrategy limit="1000"/>

 </pendingMessageLimitStrategy>

 </policyEntry>

 </policyEntries>

 </policyMap>

 </destinationPolicy>

<managementContext>

 <managementContext createConnector="false"/>

 </managementContext>

<persistenceAdapter>

 <kahaDB directory="${activemq.data}/kahadb"/>

 </persistenceAdapter>

<systemUsage>

 <systemUsage>

 <memoryUsage>

 <memoryUsage percentOfJvmHeap="70" />

 </memoryUsage>

 <storeUsage>

 <storeUsage limit="100 gb"/>

 </storeUsage>

 <tempUsage>

 <tempUsage limit="50 gb"/>

 </tempUsage>

 </systemUsage>

 </systemUsage>

 <transportConnectors>

<transportConnector name="openwire+ssl" uri="ssl://0.0.0.0:61617?maximumConnections=1000&amp;wireFormat.maxFrameSize=104857600&amp;transport.enabledProtocols=TLSv1,TLSv1.1,TLSv1.2"/>

 </transportConnectors>

 <shutdownHooks>

 <bean xmlns="http://www.springframework.org/schema/beans" class="org.apache.activemq.hooks.SpringContextHook" />

 </shutdownHooks>

 </broker>

 <import resource="jetty.xml"/>

 <import resource="camel.xml"/>

</beans>

* Конфигурационный файл маршрутизации **camel.xml**

<beans

 xmlns="http://www.springframework.org/schema/beans" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

 xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd

 http://activemq.apache.org/schema/core http://activemq.apache.org/schema/core/activemq-core.xsd

 http://camel.apache.org/schema/spring http://camel.apache.org/schema/spring/camel-spring.xsd"

>

 <camelContext id="camel" xmlns="http://camel.apache.org/schema/spring">

 <route>

 <from uri="activemq:queue:ASG\_Queue\_Ship"/>

 <to uri="activemq:queue:DutyShips.own.GKCPS"/>

 <to uri="activemq:queue:DutyShips.own.monitor1"/>

 <to uri="activemq:queue:ASG.Ship.pallada.test"/>

 <to uri="activemq:queue:ASG.Ship.pallada.layout"/>

 <to uri="activemq:queue:ASG.Ship.SMRCC.1"/>

 <to uri="activemq:queue:ASG.Ship.SMRCC.2"/>

 <to uri="activemq:queue:ASG.Ship.SMRCC.3"/>

 </route>

 <route>

 <from uri="activemq:queue:ASG\_Queue\_Craft"/>

 <to uri="activemq:queue:ASG.Craft.pallada.test"/>

 <to uri="activemq:queue:ASG.Craft.pallada.layout"/>

 <to uri="activemq:queue:ASG.Craft.SMRCC.1"/>

 <to uri="activemq:queue:ASG.Craft.SMRCC.2"/>

 <to uri="activemq:queue:ASG.Craft.SMRCC.3"/>

 </route>

 <route>

 <from uri="activemq:queue:ASG\_Queue\_OtherShip"/>

 <to uri="activemq:queue:ASG.OtherShip.pallada.test"/>

 <to uri="activemq:queue:ASG.OtherShip.pallada.layout"/>

 <to uri="activemq:queue:ASG.OtherShip.SMRCC.1"/>

 <to uri="activemq:queue:ASG.OtherShip.SMRCC.2"/>

 <to uri="activemq:queue:ASG.OtherShip.SMRCC.3"/>

 </route>

 <route>

 <from uri="activemq:topic:Topic"/>

 <to uri="activemq:queue:QueueLogger"/>

 </route>

 </camelContext>

</beans>

* Сервлет **jetty.xml**

<beans xmlns="http://www.springframework.org/schema/beans" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

 xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd">

 <bean id="securityLoginService" class="org.eclipse.jetty.security.HashLoginService">

 <property name="name" value="ActiveMQRealm" />

 <property name="config" value="${activemq.conf}/jetty-realm.properties" />

 </bean>

 <bean id="securityConstraint" class="org.eclipse.jetty.util.security.Constraint">

 <property name="name" value="BASIC" />

 <property name="roles" value="user,admin" />

 <!-- set authenticate=false to disable login -->

 <property name="authenticate" value="false" />

 </bean>

 <bean id="adminSecurityConstraint" class="org.eclipse.jetty.util.security.Constraint">

 <property name="name" value="BASIC" />

 <property name="roles" value="admin" />

 <!-- set authenticate=false to disable login -->

 <property name="authenticate" value="false" />

 </bean>

 <bean id="securityConstraintMapping" class="org.eclipse.jetty.security.ConstraintMapping">

 <property name="constraint" ref="securityConstraint" />

 <property name="pathSpec" value="/api/\*,/admin/\*,\*.jsp" />

 </bean>

 <bean id="adminSecurityConstraintMapping" class="org.eclipse.jetty.security.ConstraintMapping">

 <property name="constraint" ref="adminSecurityConstraint" />

 <property name="pathSpec" value="\*.action" />

 </bean>

 <bean id="secHandlerCollection" class="org.eclipse.jetty.server.handler.HandlerCollection">

 <property name="handlers">

 <list>

 <bean class="org.eclipse.jetty.webapp.WebAppContext">

 <property name="contextPath" value="/admin" />

 <property name="resourceBase" value="${activemq.home}/webapps/admin" />

 <property name="logUrlOnStart" value="true" />

 </bean>

 <!-- Enable embedded file server for Blob messages -->

 <!-- <bean class="org.eclipse.jetty.webapp.WebAppContext"> <property name="contextPath"

 value="/fileserver" /> <property name="resourceBase" value="${activemq.home}/webapps/fileserver"

 /> <property name="logUrlOnStart" value="true" /> <property name="parentLoaderPriority"

 value="true" /> </bean> -->

 <bean class="org.eclipse.jetty.webapp.WebAppContext">

 <property name="contextPath" value="/api" />

 <property name="resourceBase" value="${activemq.home}/webapps/api" />

 <property name="logUrlOnStart" value="true" />

 </bean>

 <bean class="org.eclipse.jetty.server.handler.ResourceHandler">

 <property name="directoriesListed" value="false" />

 <property name="welcomeFiles">

 <list>

 <value>index.html</value>

 </list>

 </property>

 <property name="resourceBase" value="${activemq.home}/webapps/" />

 </bean>

 <bean id="defaultHandler" class="org.eclipse.jetty.server.handler.DefaultHandler">

 <property name="serveIcon" value="false" />

 </bean>

 </list>

 </property>

 </bean>

 <bean id="securityHandler" class="org.eclipse.jetty.security.ConstraintSecurityHandler">

 <property name="loginService" ref="securityLoginService" />

 <property name="authenticator">

 <bean class="org.eclipse.jetty.security.authentication.BasicAuthenticator" />

 </property>

 <property name="constraintMappings">

 <list>

 <ref bean="adminSecurityConstraintMapping" />

 <ref bean="securityConstraintMapping" />

 </list>

 </property>

 <property name="handler" ref="secHandlerCollection" />

 </bean>

 <bean id="contexts" class="org.eclipse.jetty.server.handler.ContextHandlerCollection">

 </bean>

 <bean id="jettyPort" class="org.apache.activemq.web.WebConsolePort" init-method="start">

 <!-- the default port number for the web console -->

 <property name="host" value="0.0.0.0"/>

 <property name="port" value="8161"/>

 </bean>

 <bean id="Server" depends-on="jettyPort" class="org.eclipse.jetty.server.Server"

 destroy-method="stop">

 <property name="handler">

 <bean id="handlers" class="org.eclipse.jetty.server.handler.HandlerCollection">

 <property name="handlers">

 <list>

 <ref bean="contexts" />

 <ref bean="securityHandler" />

 </list>

 </property>

 </bean>

 </property>

 </bean>

 <bean id="invokeConnectors" class="org.springframework.beans.factory.config.MethodInvokingFactoryBean">

 <property name="targetObject" ref="Server" />

 <property name="targetMethod" value="setConnectors" />

 <property name="arguments">

 <list>

 <bean id="Connector" class="org.eclipse.jetty.server.ServerConnector">

 <constructor-arg ref="Server" />

 <!-- see the jettyPort bean -->

 <property name="host" value="#{systemProperties['jetty.host']}" />

 <property name="port" value="#{systemProperties['jetty.port']}" />

 </bean>

 <!--

 Enable this connector if you wish to use https with web console

 -->

 <!-- bean id="SecureConnector" class="org.eclipse.jetty.server.ServerConnector">

 <constructor-arg ref="Server" />

 <constructor-arg>

 <bean id="handlers" class="org.eclipse.jetty.util.ssl.SslContextFactory">

 <property name="keyStorePath" value="${activemq.conf}/broker.ks" />

 <property name="keyStorePassword" value="password" />

 </bean>

 </constructor-arg>

 <property name="port" value="8162" />

 </bean -->

 </list>

 </property>

 </bean>

 <bean id="configureJetty" class="org.springframework.beans.factory.config.MethodInvokingFactoryBean">

 <property name="staticMethod" value="org.apache.activemq.web.config.JspConfigurer.configureJetty" />

 <property name="arguments">

 <list>

 <ref bean="Server" />

 <ref bean="secHandlerCollection" />

 </list>

 </property>

 </bean>

 <bean id="invokeStart" class="org.springframework.beans.factory.config.MethodInvokingFactoryBean"

 depends-on="configureJetty, invokeConnectors">

 <property name="targetObject" ref="Server" />

 <property name="targetMethod" value="start" />

 </bean>