

АХИОМ JDK

Доверенный репозиторий Руководство пользователя

АО «АКСИОМ» | Апрель 2026

Copyright © 2019-2026 Все права защищены АО "АКСИОМ" (АКСИОМ)

Программное обеспечение АКСИОМ содержит программное обеспечение с открытым исходным кодом. Дополнительная информация о коде сторонних разработчиков доступна на сайте https://axiomjdk.ru/third_party_licenses. Для дополнительной информации о том, как получить копию исходного кода, можно обратиться по адресу info@axiomjdk.ru.

ДАННАЯ ИНФОРМАЦИЯ МОЖЕТ ИЗМЕНЯТЬСЯ БЕЗ ПРЕДВАРИТЕЛЬНОГО УВЕДОМЛЕНИЯ. АКСИОМ ПРЕДОСТАВЛЯЕТ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ "КАК ЕСТЬ" БЕЗ КАКИХ-ЛИБО ГАРАНТИЙ, АКСИОМ ПРЯМО ОТКАЗЫВАЕТСЯ ОТ ВСЕХ ПОДРАЗУМЕВАЕМЫХ ГАРАНТИЙ, ВКЛЮЧАЯ, НО НЕ ОГРАНИЧИВАЯСЬ ПОДРАЗУМЕВАЕМЫМИ ГАРАНТИЯМИ ТОВАРНОЙ ПРИГОДНОСТИ И ПРИГОДНОСТИ ДЛЯ ОПРЕДЕЛЕННОЙ ЦЕЛИ.

АКСИОМ НИ ПРИ КАКИХ ОБСТОЯТЕЛЬСТВАХ НЕ НЕСЕТ ОТВЕТСТВЕННОСТИ ЗА ЛЮБЫЕ КОСВЕННЫЕ, СЛУЧАЙНЫЕ, СПЕЦИАЛЬНЫЕ, ШТРАФНЫЕ ИЛИ КОСВЕННЫЕ УБЫТКИ, ИЛИ УБЫТКИ ОТ ПОТЕРИ ПРИБЫЛИ, ДОХОДА, ДАННЫХ ИЛИ ИСПОЛЬЗОВАНИЯ ДАННЫХ, ПОНЕСЕННЫЕ ВАМИ ИЛИ ЛЮБОЙ ТРЕТЬЕЙ СТОРОНОЙ, БУДЬ ТО В РЕЗУЛЬТАТЕ ДЕЙСТВИЯ ДОГОВОРА ИЛИ ДЕЛИКТА, ДАЖЕ ЕСЛИ АКСИОМ БЫЛО ПРЕДУПРЕЖДЕНО О ВОЗМОЖНОСТИ ТАКИХ УБЫТКОВ.

Использование любого программного продукта АКСИОМ регулируется соответствующим лицензионным соглашением, которое никоим образом не изменяется условиями данного уведомления. Программные продукты и фирменные наименования: Axiom JDK, Axiom JDK Pro, Axiom Runtime Container Pro, Axiom Linux, Libercat, Libercat Certified и АКСИОМ принадлежат АКСИОМ и их использование допускается только с разрешения правообладателя.

Товарный знак Linux® используется в соответствии с сублицензией от Linux Foundation, эксклюзивного лицензиата Линуса Торвальдса, владельца знака на всемирной основе. Java и OpenJDK являются товарными знаками или зарегистрированными товарными знаками компании Oracle и/или ее аффилированных лиц. Другие торговые марки являются собственностью их соответствующих владельцев и используются только в целях идентификации.

Содержание

1. Введение	4
Риски при использовании стороннего репозитория	4
2. Просмотр и скачивание содержимого репозитория в браузере.....	6
3. Доступ к репозиторию.....	7
4. Настройка инструментов сборки для работы с репозиторием	8
Maven	8
Gradle	10
Ant	10

1. Введение

Данный документ предназначен для аудитории, которая взаимодействует с доверенным репозиторием для выполнения своих задач:

- Пользователи, которые используют репозиторий для загрузки или обновления программного обеспечения;
- Разработчики и тестировщики, которые пользуются репозиторием для загрузки библиотек, фреймворков и других зависимостей, необходимых для их проектов;
- ИТ-специалисты, устанавливающие и настраивающие ПО на рабочих местах сотрудников, используя доверенный репозиторий.

Риски при использовании стороннего репозитория

В настоящее время сборка Java приложений на 95% зависит от сторонних библиотек. Использование сторонних библиотек в разработке программного обеспечения может ускорить разработку и добавить функциональность, но также несет определенные риски. Вот некоторые из них:

- Использование кода из открытых источников создает угрозы безопасности для внутреннего периметра приложений.
- Отсутствуют средства управления рисками при использовании сторонних библиотек.
- Сканеры безопасности проводят только сверку версии библиотеки с публичными данными о наличии в ней уязвимостей.

При использовании сторонних библиотек могут возникать риски недокументированного поведения:

- Неверифицированный репозиторий исходного кода

Последствия:

- Нарушение безопасности данных пользователей.
 - Потеря доверия со стороны клиентов и партнеров.
 - Финансовые и репутационные убытки.
- Неверифицированные средства сборки: компилятор, maven, gradle

Последствия:

- Возможность использования уязвимостей злоумышленниками.
- Нестабильность и ненадежность программного обеспечения.
- Увеличение затрат на исправление ошибок и доработку ПО.
- Недостаточное тестирование исходного кода на безопасность

Последствия:

- Компрометация данных пользователей.
- Правовые последствия и штрафы за нарушение нормативных требований.
- Ущерб репутации и финансовые потери.
- недостаточная информированность о существующих уязвимостях библиотек

Последствия:

- Использование известных уязвимостей злоумышленниками.
- Потеря данных и нарушение работы системы.
- Ущерб репутации и финансовые убытки.

Внедрение доверенного репозитория играет ключевую роль в управлении процессами разработки программного обеспечения, обеспечения безопасности и автоматизации.



2. Просмотр и скачивание содержимого репозитория в браузере

Содержимое репозитория можно просматривать по ссылке <https://repo.axiomjdk.ru/> с помощью браузера (например, Яндекс Браузер, Microsoft Edge, Google Chrome, Firefox и др.).

При сборке проекта нужные артефакты должны быть скачаны из репозитория axiom и использованы для построения вашего проекта.

3. Доступ к репозиторию

Авторизация для доступа к репозиторию осуществляется с помощью имени пользователя - это имя проекта и пароля - это токен.

Имя проекта можно увидеть и скопировать в личном кабинете (ЛК) на [портале поддержки](#) под заголовком "Поддержка" в строке **Имя проекта**. Токен необходимо создать в том же личном кабинете на [портале поддержки](#).

Информацию по созданию токена смотри в разделе "Аутентификация" главы "Автоматизация загрузок с партнерским API" в документе "[Инструкция по работе с порталом поддержки](#)". Там же находится общая информация по работе с порталом поддержки, включая авторизацию, доступ к продуктам и т.д.

4. Настройка инструментов сборки для работы с репозиторием

Maven

Ниже приведены настройки Maven для работы с репозиторием axiom.

Доверенный репозиторий может быть определен как внутри сборочного файла проекта `pom.xml`, так и внутри общего файла настроек Maven `settings.xml`. Определения в файле проекта имеют приоритет над определениями в файле настроек. При этом определения в файле настроек являются глобальными и доступны во всех проектах.

Для добавления доверенного репозитория в файл проекта `pom.xml` нужно в секции `<repositories>` добавить элемент `<repository>` следующим образом:

```
<project>
...
  <repositories>
    <repository>
      <id>trusted-repo</id>
      <name>Axiom Trusted Repository</name>
      <url>https://repo.axiomjdk.ru/</url>
    </repository>
    ...
  </repositories>
  ...
</project>
```

Аналогичным образом репозиторий может быть определен в файле настроек `settings.xml`, находящимся в корне локального репозитория (обычно `~/.m2`):

```
<settings>
...
  <profiles>
    ...
    <profile>
      <id>myprofile</id>
      <repositories>
        <repository>
```

```
        <id>trusted-repo</id>
        <name>Axiom Trusted Repository</name>
        <url>https://repo.axiomjdk.ru/</url>
    </repository>
</repositories>
</profile>
...
</profiles>

<activeProfiles>
<activeProfile>myprofile</activeProfile>
</activeProfiles>
...
</settings>
```

Поскольку для доступа к доверенному репозиторию требуется пройти процедуру аутентификации, необходимо в файл настроек `settings.xml` добавить предоставленные имя пользователя и пароль:

```
<settings>
  <servers>
    <server>
      <id>trusted-repo</id>
      <username>имя пользователя</username>
      <password>ваш токен</password>
    </server>
  </servers>
</settings>
```

**Важно:**

Обратите внимание на элемент `<id>` внутри элемента `<server>`, он должен совпадать с элементом `<id>` внутри элемента `<repository>`. В противном случае maven не сможет использовать предоставленные аутентификационные данные для доверенного репозитория.

В целях безопасности, мы рекомендуем использовать шифрование пароля, для более детальных инструкций ознакомьтесь с [официальной документацией](#).

Gradle

Доверенный репозиторий может быть добавлен в секции `repositories` внутри сборочного файла проекта `build.gradle` (или иным способом, плагин, скрипт инициализации и т.п.).

```
repositories {  
    ...  
    maven {  
        name "Axiom Trusted Repository"  
        url "https://repo.axiomjdk.ru/"  
        credentials {  
            username "имя пользователя"  
            password "ваш токен"  
        }  
    }  
}
```

В целях безопасности, мы рекомендуем использовать специальные переменные для хранения пароля (имени пользователя) и передавать их в качестве параметров командной строки.

Ant

В Ant не существует встроенного механизма работы с зависимостями проекта. И для того, чтобы сделать сборку проекта, могут быть использованы различные подходы. Например, скачать все необходимые артефакты из репозитория <https://repo.axiomjdk.ru/> с помощью браузера или утилиты `wget`.

Еще одна из многочисленных альтернатив, использование расширения Apache Ivy™.

