Ключевые слова: метод, прогноз, параметр, сеть автомобильных дорог.

Parameters of a network of highways, methods which are used at long-term forecasting of parameters of a network of highways are analysed and is established that the main shortcoming at long-term forecasting of parameters of a network of highways is lack of system approach and detailed consideration of organizational characteristics of system when using a method of evolutionary and probabilistic modeling.

Keywords: method, forecast, parameter, network of highways.

УДК 65.011.56

Е. П. ПАВЛЕНКО, канд. техн. наук, доц., ХНУРЭ, Харьков;
И. А. КРИВОРОТЕНКО, студент, ХНУРЭ, Харьков;
В. А. АЙВАЗОВ, ст. преп., ХНУРЭ, Харьков

РАЗРАБОТКА МОДУЛЯ ЗАЩИТЫ ИНФОРМАЦИОННОЙ СИСТЕМЫ ПОЛИГРАФИЧЕСКОГО ПРЕДПРИЯТИЯ

Рассмотрена проблема защиты информационного обеспечения информационной системы полиграфического предприятия. Исследованы способы разработки модуля защиты с использованием языка высокого уровня Java, а также проведено сравнение фреймворков Spring Security и Apache Shiro.

Ключевые слова: защита информации, Spring framework, Enterprise Java Beans.

Введение. Компания «Радуга» занимается производством продукции допечатной подготовки. Информационные системы в области полиграфии пользуются большой популярностью. Это обусловлено значительным уменьшением времени, затраченного на подготовку печати и послепечатные процессы.

Информационные системы позволяют автоматизировать основные процессы производства, за счет чего увеличить объемы производства, и следовательно приумножить прибыль предприятия. Проекты в области автоматизации рассматриваются полиграфическим предприятием как стратегическая инвестиция средств, которая должна окупиться за счет улучшения управленческих процессов, повышения эффективности производства, сокращения издержек.

Проектирование информационных систем является длительным процессом, требующим согласования разрабатываемых элементов. Процесс проектирования базируется на функциональной структуре системы, определяющей множество функций, поддерживаемых системой.

Информационная система является механизмом для повышения эффективности управления, принятия правильных стратегических и тактических решений на основе своевременной и достоверной информации, выдаваемой компьютером. В основе разработки информационной системы для полиграфического предприятия лежит принцип создания единого хранилища данных, содержащего информацию, накопленную организацией в процессе ведения бизнес-процессов, включая финансовую информацию, данные, связанные с производством, управлением персоналом, данные складского учета.

Цель работы. Целью работы является исследование методов разработки программного обеспечения для модуля защиты ИС полиграфического предприятия

Постановка задачи. Для разработки приложения была выбрана платформа JavaEE языка программирования Java. JavaEE позволяет строить эффективные серверные приложения, в которых необходима гибкость, масштабируемость и надежность. Одним из наиболее важных модулей системы является модуль защиты,

который выполняет функции распределения доступа пользователей к информации. В модуль защиты входят задачи аутентификации пользователей, авторизации пользователей, шифрования данных.

Аутентификация – это процесс идентификации личности пользователя, который позволяет уточнить, кем является пользователь. Авторизация – это процесс управления доступом пользователя к некоторым ресурсам, который позволяет определить, кто имеет доступ к чему.

Эти задачи являются ключевыми в реализации механизма защиты информации от различного рода нарушений, поэтому при разработке информационной системы им следует уделять особое внимание.

Основные тенденции разработки модуля защиты с использованием языка Java. В среде Java EE на данный момент существуют готовые решения, позволяющие сократить время на разработку модуля защиты. Наиболее развитыми и эффективными являются Java-фреймворки Spring Security и Apache Shiro.

Spring Security дает возможность разработать модули аутентификации и авторизации, а также другие возможности обеспечения безопасности для кроссплатформенных приложений, созданных с помощью Spring Framework. Spring Security легко интегрируется с приложениями, которые используют Spring Framework [1-2].

Фреймворк реализует стандартную идею безопасности Java-аутентификации Principal (java.security.Principal), с помощью которого однозначно представляется аутентифицированный пользователь.

Spring Security поддерживает различные протоколы авторизации и поддерживает механизмы их настройки. В этом процессе задействуются такие компоненты, как фильтры. Это высокоуровневые компоненты, обеспечивающие функционирование модулей защиты. В зависимости от требований к защите приложения можно применять разные виды фильтров. Spring Security имеет структуру, с помощью которой можно первоначально настроить функциональность. Затем конфигурация расширяется, чтобы получить необходимый контроль над приложением.

Конфигурация Spring Security, задействованная при разработке модуля защиты, представлена ниже:

```
<?xml version="1.0" encoding="UTF-8"?>
<beans:beans xmlns="http://www.springframework.org/schema/security"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:beans="http://www.springframework.org/schema/beans"
  xsi:schemaLocation="
http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd
http://www.springframework.org/schema/security
http://www.springframework.org/schema/security/
spring-security-3.0.xsd">
  <http auto-config="true">
    <intercept-url pattern="/*" access="ROLE_USER" />
  </http>
  <authentication-manager alias="authenticationManager">
    <authentication-provider>
      <user-service>
        <user authorities="ROLE_USER" name="guest" password="guest" />
```

```
      </user-service>
    </authentication-provider>
  </authentication-manager>
</beans:beans>
```

Аутентификация в Spring Security настраивается при помощи декларирования тега authentication-provider. Провайдер аутентификации выполняет проверку идентификационных данных пользователя. Чтобы написать свою функцию аутентификации, необходимо имплементировать интерфейс AuthenticationProviderInterface, или выполнить наследование своего класса от абстрактного класса. AuthenticationProviderInterface включает методы authenticate() и supports(). На вход данных методов поступает объект token, содержащий совокупность данных, идентифицирующих пользователя. Авторизация также настраивается при помощи задания ограничений на определенный перечень адресов с помощью тега intercept-url.

Основа, на которой базируется разработка схемы аутентификации - это класс SecurityContextHolder. Класс содержит информацию о контексте приложения, о пользователях, которые с ним работают. Для хранения информации о пользователях используется объект Authentication, с помощью которого можно получить, например, логин пользователя.

С помощью интерфейса UserDetails можно получить расширенную информацию о пользователе. Для получения объекта UserDetails, необходимо воспользоваться интерфейсом UserDetailsService, который поддерживает метод, выполняющий поиск пользователя по логину и возвращение UserDetails. Объект UserDetails содержит данные о правах пользователя, которые необходимо проверять при обращении к защищенному ресурсу.

Spring Security можно выбрать, если необходимо разработать простой модуль защиты для обеспечения авторизации и аутентификации пользователей. Этот фреймворк достаточно гибок, и позволяет настроить нужную конфигурацию с помощью XML-тегов.

Альтернатива Spring Security – это фреймворк Apache Shiro. Это мощный и гибкий фреймворк, позволяющий упростить выполнение авторизации и аутентификации.

Основным объектом в Apache Shiro является SecurityManager, который инициализируется при помощи ряда настроек [3]. Для настроек можно использовать ini-файл. Существует возможность настраивать конфигурацию при помощи set-методов, XML-тегов. Apache Shiro также легко интегрируется с фреймворком Spring.

Конфигурация Apache Shiro на основе ini-файла, которая применялась при разработке модуля защиты, представлена ниже:

```
[main]
sampleauthc = shiro.sample.SampleFormAuthenticationFilter
sampleauthc.loginUrl = /login
# Users and their (optional) assigned roles
# username = password, role1, role2, ..., roleN
[users]
root = secret, admin
guest = guest, user
[urls]
/login = anon
```

/admin/**= roles[admin]

/user/** = roles[user], roles[admin]

Для получения данных аутентификации Apache Shiro использует концепцию Realm. Realm – это компонент, связывающий Apache Shiro с источником данных, который хранит в себе информацию о пользователях. Apache Shiro позволяет настраивать несколько таких компонентов для взаимодействия с различными объектами.

Авторизация в Apache Shiro реализуется путем обьявления фильтров с различными параметрами в конфигурационном файле. Помимо ролей, используются разрешения, которые позволяют распределить права доступа пользователей.

Фреймворк Apache Shiro успешно применяется к двум технологиям Web-приложений. Можно использовать интерфейс Apache Shiro для работы с сервлетами. С другой стороны, можно применить HTTP-фильтры. Применение фильтров базируется на встроенном сервере Web-приложений. Строки для этих фильтров добавляются в конфигурационный файл web.xml вручную.

Фреймворк поддерживает подключаемые модули, которые позволяют выполнить автоматическую генерацию файла web.xml, что более удобно для разработчика. Подключаемые модули содержат скрипты, которые запускаются для создания различных конфигураций системы защиты. Имеется возможность написать свой подключаемый модуль для выполнения специфических процедур защиты.

Объект Session рассматриваемого фреймворка позволяет работать с сеансом пользователя. Применение этого объекта позволяет использовать один и тот же программный код, даже если этот код не исполняется в Web-приложении. Таким образом, Apache Shiro можно использовать даже в командной строке. Код, написанный с помощью Apache Shiro, позволяет разработчику создавать основанные на командной строке приложения для соединения с LDAP-сервером.

Помимо ряда функций защиты, Apache Shiro поддерживает управление сессией, кеширование данных, реализации криптографических алгоритмов. Фреймворк предоставляет возможность хранить сессию по выбору разработчика - при помощи интерфейса SessionDao. Особенность Shiro - это интеграция EHCache. Фреймоворк можно настроить на использование кеширования данных при авторизации и аутентификации.

Выводы. Spring Security является приемлемым вариантом в случае простого приложения, в котором не требуется реализации сложной логики распределения доступа к данным, так как предоставляет возможность быстрой настройки с параметрами по умолчанию. С другой стороны, если разрабатываемое приложение требует высокого уровня защиты данных, более подходящим фреймворком можно назвать Apache Shiro. Реализации функций доступа к данным в обоих фреймворках не уступают друг другу, но Apache Shiro предоставляет множество средств, которые могут значительно упросить реализацию сложной логики доступа к данным.

Приложение, выполняющее функции учета произведенной продукции полиграфического предприятия «Радуга», является распределенным, поэтому необходимо уделить большое внимание модулю защиты данных. Целесообразным в данном случае будет использование фреймворка Apache Shiro, который предоставляет не только функции защиты данных, но и дополнительные функции кеширования, управления сессией.

Список литературы: 1. *Walls, C.* Spring in Action. Third Edition [Текст] / *C. Walls* – Manning Publications Co., 2011.- 426 с. **2.** *Mularien, P.* Spring Security 3 [Текст] / *P. Mularien* –PACKT Publications, 2010 – 396 с. **3.** Apache Shiro Documentation [Электронный ресурс] – режим доступа: http://shiro.apache.org/documentation.html, 11.03.2013

Надійшла до редколегії 20.04.2013

УДК 65.011.56

Разработка модуля защиты информационной системы полиграфического предприятия / Е. П.Павленко, И. А.Криворотенко, В. А.Айвазов // Вісник НТУ «ХПІ». Серія: Нові рішення в сучасних технологіях. – Х: НТУ «ХПІ», – 2013. - № 26 (999). – С.30-34 . – Бібліогр.: 3 назв.

Розглянуто проблему захисту інформаційного забезпечення інформаційної системи поліграфічного підприємства. Досліджені основні тенденції розробки модуля захисту з використанням мови високого рівню Java, також проведено порівняння фреймворків Spring Security та Apache Shiro.

Ключові слова: захист інформації, Spring framework, Enterprise Java Beans.

It was considered the problem of information provision security of printing company. It was researched the main trends of development security module using high-level language Java, also it was compared Spring Security and Apache Shiro frameworks. Bibliogr.: 3.

Keywords: information security, Spring framework, Enterprise Java Beans.

УДК 519.237:616-006

Е. В. ВЫСОЦКАЯ, канд. техн. наук, проф. каф., ХНУРЭ, Харьков;
В. И. ЖУКОВ, д-р мед. наук, проф., зав. каф., Харьковский национальный медицинский университет;
А. П. ПОРВАН, канд. техн. наук, с. н. с., ХНУРЭ, Харьков;
А. С. МОИСЕЕНКО, м. н. с., ГУ «Институт общей и неотложной хирургии НАМН Украины», Харьков;
ФАМ ТХИ ХУЭН ЧАНГ, студент, ХНУРЭ, Харьков

МЕТОД ОПРЕДЕЛЕНИЯ СТЕПЕНИ ТЯЖЕСТИ АДЕНОКАРЦИНОМЫ ЖЕЛУДКА

В статье предложен метод определения степени тяжести аденокарциномы желудка у человека, полученный в результате дискриминантного анализа клинических, инструментальных и лабораторных показателей.

Ключевые слова: дискриминантный анализ, аденокарцинома, территориальная карта.

Введение. Несмотря на кардинальные хирургические вмешательства и адекватную терапию, смертность больных от рака желудка остается на высоком уровне и составляет до 10% смертельных случаев от опухолей всех локализаций. Следует отметить, что после лечения больных, которое состоит в резекции желудка, субтотальной или тотальной гастроэктомии, лимфоденэктомии, пятилетнее выживание составляет всего до 30%. Поэтому нет сомнения, что дальнейшее изучение молекулярно-метаболических особенностей и патогенетических механизмов формирования рака желудка будет способствовать поиску новых подходов для повышения диагностики, эффективности лечения и прогнозирования выживания больных [1].

В связи с выше сказанным использование мониторинговых метаболических показателей, которые отражают состояние основных видов обмена веществ и энергии у больных с аденокарциномой желудка (АКЖ), является актуальным. Это позволяет объективно оценить изменения со стороны белкового, углеводного, липидного, и минерального обмена в зависимости от степени тяжести заболевания, стадии развития патологического процесса, что может быть прогностически значимым при выборе