Twitter-ID. Текст должен быть написан непосредственно пользователем и иметь эмоциональный окрас. Например, пользователь в данном тексте может рассказать о себе, своей жизни, приятных моментах из детства либо своём взгляде на какую-либо проблему. Далее данный текст через API отправляется на обработку в IBM Watson, в ответ мы получаем портрет личности, представленный численно в трёх моделях. Полученные значения сверяются с эталонными для специальностей и в качестве ответа выбирается специальность, значения которой оказываются наиболее «близко» от значений пользователя. Самой большой проблемой является выбор этих самых эталонных значений для специальностей. В данный момент используются тексты с описанием специальностей, представленные на сайте университета и пропущенные также через Personality Insights. Выбор этих текстов нельзя назвать оптимальным. Наиболее лучшим вариантом был бы преданализ текстов выпускников данных специальностей, и не просто выпускников – а тех, что действительно осознанно сам выбрал эту специальность и считает, что он попал на учёбу в нужное место. Далее, проанализировав данные тексты, можно будет получить более точную оценку эталонных значений.

Нами было написано приложение, реализующее данный функционал с использованием когнитивных возможностей IBM Watson, развёрнутое на базе облачной платформы IBM Bluemix. Приложение было написано и успешно представлено на хакатоне IBM Bluemix Hackathon Moscow в 2015-м году, где вошло в тройку лучших проектов.

Список литературы:

1. IBM Watson. *What is Watson* [Электронный ресурс]. – Электронные данные. – Режим доступа: http://www.ibm.com/smarterplanet/us/en/ibmwatson/what-is-watson.html
2. IBM Watson Developer Cloud. *The science behind the Personality Insights service* [Электронный ресурс]. – Электронные данные. – Режим доступа: http://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/doc/personality-insights/science.shtml

АНАЛИЗ ИСПОЛЬЗОВАНИЯ МЕТОДОВ КРИПТОГРАФИЧЕСКИХ ПРЕОБРАЗОВАНИЙ ДЛЯ ЗАЩИТЫ ПОЛЬЗОВАТЕЛЬСКИХ ДАННЫХ

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Рогов М. Г.

Егорова Н. Г. – доцент, канд. техн. наук

Защита персональных данных – это целый комплекс мероприятий технического, организационного и организационно-технического характера, которые необходимы для защиты информации, относящейся к определенному или определяемому на основании такой информации физическому лицу (субъекту персональных данных) [1].

Проблема защиты данных с использованием криптографических преобразований имеет множество вариантов решения. Для осуществления безопасности имеющихся в наличии данных нужно соблюдать основные правила:

- обеспечивать конфиденциальность информации, передаваемой или хранимой в памяти;
- всегда подтверждать подлинность и целостность информации;
- при каждой установке соединения и входе пользователя в систему производить аутентификацию абонента.

Возможность осуществления какой-либо операции преобразования и изменения информации, которая выполняется одним либо несколькими пользователями системы, владеющими так называемым секретом, не зная которого нет возможности осуществить эту операцию, является основой всех криптографических преобразований информации [2].

Криптографические технологии аутентификации, шифрования и цифровой подписи используются для реализации указанных функций.

К криптографическим методам защиты в общем случае относятся:

- шифрование (дешифрование) информации;
- формирование и проверка цифровой подписи электронных документов.

Чтобы обеспечить конфиденциальность данных используются различные методы симметричного и ассиметричного шифрования, а также взаимная аутентификация на основе одноразовых и многоразовых паролей, смарт-карт, цифровых сертификатов и многого другого [3].

Современные криптографические системы и методы защиты информации должны обладать рядом общепринятых свойств:

- возможность расшифровать сообщение существует только при наличии;
- владение алгоритмом никоим образом не должно влиять на защищенность информации;
- без изменений остаются структурные элементы алгоритма шифрования;

- изменение самого вида зашифрованного сообщения происходит даже при незначительном изменении ключа шифрования;
- нет понятных зависимостей между ключами, если в процессе шифрования последовательно используются несколько ключей;
- все дополнительные биты, если таковые вводятся в сообщение, хорошо скрыты в зашифрованном сообщении;
- любой ключ из множества возможных способен обеспечить надежную защиту информации.

В ходе работы было изучено множество методов шифрования, используемых как в информационных системах, так и в повседневной жизни. Также были изучены различные особенности защиты информационных систем. Был проведен анализ стойкости различных методов криптографических преобразований. Выявленные в результате данного анализа положительные и отрицательные особенности различных методов криптографических преобразований позволили выделить наиболее верные пути защиты информации. Также были выявлены особенности некоторых методов защиты информации с точки зрения удобства для рядового пользователя.

Список использованных источников:

1. Скембрей Дж. Секреты хакеров / Дж. Скембрей, Ст. Мак-Клар – Москва: «Вильямс», 2004. – 512 с.
2. Смарт Н. Cryptography: An Introduction / Н. Смарт – Москва: «Техносфера», 2006. – 528 с.
3. Сингх С. Книга шифров / С. Сингх – Москва: «Астрель», 2006. – 447 с.

СРАВНИТЕЛЬНАЯ ХАРАКТЕРИСТИКА ВЫСОКОНАГРУЖЕННЫХ ВЕБ-ПРИЛОЖЕНИЙ НА ПЛАТФОРМЕ JAVA

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Вакульчик Е. Н.

Куликов С. С. – к.т.н., доцент

Современным веб-приложениям приходится обрабатывать сотни и даже тысячи запросов в секунду. Чтобы обеспечить необходимую производительность, необходимо определить потенциальные узкие места и предпринять меры по их устранению. Стадии оптимизации производительности должна предшествовать стадия тщательного исследования и анализа.

В настоящее время существуют две конкурирующие платформы для разработки веб-приложений одинаковой сложности, использующих в качестве основы язык Java: Java EE и Spring. Целью данного исследования являлась разработка двух веб-приложений на платформах Java EE и Spring соответственно, осуществляющих запись и выборку данных из БД, и сравнение их производительности. В качестве СУБД использовался MySQL, а в качестве сервера приложений – GlassFish. Для проведения нагрузочного тестирования и отметки ключевых показателей использовалось средство автоматизации нагрузочного тестирования JMeter.

В связке Spring и Hibernate для эффективного доступа к базе данных использовался пул соединений, представляющий собой кэш для открытых соединений между приложением и СУБД. Использование такого решения позволяет сильно сократить накладные расходы на установку и инициализацию нового соединения между приложением и СУБД.

Нагрузочное тестирование веб-приложений проводилось с помощью варьирования количеством пользователей и регистрируя ключевые показатели. Наиболее важные регистрируемые показатели: пропускная способность (з\с - число запросов в секунду), время реакции, процент загрузки ЦП серверного компьютера, процент загрузки ЦП БД.

Анализируя результаты зависимости з\с и времени ответа от количества пользователей (запись в БД) для Java EE приложения можно сделать вывод о том, что до 750 количества пользователей производительность линейно возрастает, а время отклика остается относительно небольшим. От 750 до 1250 время отклика резко повышается, а производительность достигает своего максимума в точке 1250. Дальнейший рост числа пользователей только уменьшает производительность и значительно увеличивает время отклика.

Анализируя результаты зависимости з\с и времени ответа от количества пользователей (выборка из БД) для Java EE приложения можно сделать вывод, что до 1000-1250 пользователей производительность растет, а время отклика остается небольшим, после этого значения время отклика резко возрастает, а производительность при этом остается практически неизменной.

Проводя анализ зависимости з\с и времени ответа от количества пользователей (запись в БД) для Spring приложения, наблюдается, что до 1500-1750 пользователей время отклика остается на низком уровне, а производительность растет. Далее время отклика резко возрастает, а производительность остается на