

Система «SC-НДО» обслуживает инкассацию наличных денежных средств, платежных инструкций, драгоценных металлов, драгоценных камней и иных ценностей в 173 филиалах и отделениях в 7 крупнейших банках Республики Беларусь. «SC-УЧЕТ ЦЕННОСТЕЙ» является единственной специализированной системой в Республике Беларусь для централизованного учета операций банка со всеми видами ценностей. АБС «SC-BANK» внедрена в ОАО «АСБ Беларусбанк» в 1996 году и сегодня обеспечивает работу 144 филиалов и ЦБУ по всей Беларуси, в каждом из которых обслуживается в среднем около 35 000 счетов и совершается в день в среднем 20000 - 25000 операций. Систему «SC-CARD» используют 23 из 31 белорусских банков, эмитирующих карточки, в том числе ОАО «АСБ Беларусбанк», ОАО «Белагропромбанк», ОАО «БПС-Банк», а также Национальный банк Республики Беларусь.

Литература

1. www.softclub.by – сайт ЗАО «СОФТКЛУБ»;
2. www.nbrb.by – сайт Национального банка Республики Беларусь;
3. www.tup.km.ua - доклад «Банковские Информационные Системы», Г. Котелевец, А. Орехов.

УДК 004.6

МАТЕМАТИЧЕСКИЕ МОДЕЛИ ЗАЩИТЫ ИНФОРМАЦИОННЫХ РЕСУРСОВ МАЛЫХ ПРЕДПРИЯТИЙ

Охримчук В.В.

УО «Брестский государственный университет им. А.С.Пушкина», г. Брест

В течение 2008-2009 года нами реализованы ряд программных систем управления базами данных малых предприятий.

Выполненные реализации СУБД показали, что удобно применять средства автоматизации проектирования, которые обладают следующими преимуществами: не требуют значительных временных затрат на разработку, внедрение и эксплуатацию; свободно распространяются и имеют достаточно полное описание для ведения собственных работ; имеют средства защиты данных от несанкционированного использования.

Практика эксплуатации систем управления базами данных позволила установить, что простота их разработки с использованием имеющихся средств автоматизации обогрывается недостаточным вниманием к средствам защиты коммерческой информации от несанкционированного доступа.

Нами выполнен анализ возможных подходов к созданию модели данных, реализована такая модель для малого предприятия, разработана СУБД. Программные коды, реализующие представленную систему, включают методы их защиты от несанкционированного доступа.

В выполненных работах реализованы различные подходы, позволяющие снизить уязвимость простейших информационных систем для малых предприятий. Представим краткое описание одного из проектных решений, предназначенного для автоматизации управления малым предприятием – брачное агентство. Изучение соответствующей предметной области и проектирование выполнены в соответствии с заказом индивидуального предпринимателя Крылова Н.А., работающего на основе лицензии 02010/0338190 от 21.12.2007г., выданной Министерством внутренних дел Республики Беларусь.

Созданная нами система опирается на многопользовательские базы данных, для которой удобно применить реляционные модели. Реализация СУБД показала, что на различных этапах разработки удобно применять следующие средства автоматизации: AllFusion ERWin Data Modeler (ErWin), базовый пакет «Denwer», включающий набор дистрибутивов для разработки сайтов на локальном компьютере: Apache, PHP, MySQL, Perl и другие [1]. Такой выбор объясняется условиями функционирования малых предприятий. Их основным требованием является создание программного продукта для управления данными, размещёнными в глобальной сети, наиболее эффективным образом. Параметрами эффективности, выдвигаемыми заказчиками, являются: время создания, затраты на проектирование, создание и внедрение, эксплуатацию.

Процесс внедрения и эксплуатации указанных СУБД показал необходимость обязательного использования дополнительных средств защиты информации. Одной из основных причин защиты информации в условиях малых предприятий является недостаток квалифицированных специалистов для сопровождения программных продуктов и баз данных. Так, например, в случае использования Интернет версии, как правило, проблем не возникает, серверные услуги предоставляются специализированными предприятиями. Однако условия заказчика в подавляющем большинстве проектов предполагают эксплуатацию системы на локальном компьютере. При этом безопасности приложений путем правильной настройки сервера не всегда уделяется достаточное внимание. В описанной ситуации требуется обеспечить защиту каждого обращения к базе данных. Для примера приведем фрагмент обращения к функциям кодирования, используемым при редактировании данных об абонентах, информация которых содержится в базе данных (см. Фрагмент).

Фрагмент. Редактирование данных об абонентах.

```

if( isset($_POST['save']) ){
    $occupation = ""; // занятость
    for ($z=1; $z<=6; $z++)
        if( isset($occup[$z]) )
            $ocp[] = $occup[$z];
    if( isset($ocp) )
        $occupation = implode(' ', $ocp);

    $invalid = (isset($_POST['invalid'])) ? $_POST['invalid'] : 0 ;
    $ogranich = (isset($_POST['ogranich'])) ? $_POST['ogranich'] : 0;

    // кодируем некоторые параметры
    $_POST['adres'] = xorShipher( $_POST['adres'] , $password );
    $_POST['pasport'] = xorShipher( $_POST['pasport'] , $password );
    $_POST['dr'] = xorShipher( $_POST['dr'] , $password );
    $_POST['tel'] = xorShipher( $_POST['tel'] , $password );
    $_POST['surname'] = xorShipher( $_POST['surname'] , $password );
    $_POST['name'] = xorShipher( $_POST['name'] , $password );

    // SQL-запрос по редактированию абонента в базе данных
    $result=mysql_query("UPDATE `abonent` SET `pol` = '{$_POST['pol']}' `dr` =
'{$_POST['sr']}', `rost` = '{$_POST['rost']}', `ves` = '{$_POST['ves']}', `region` =
'{$_POST['region']}', `city` = '{$_POST['city']}', `education` = '{$_POST['education']}',
`occupation` = '{$_POST['occupation']}', `religion` = '{$_POST['religion']}', `dom` =
'{$_POST['dom']}', `smoking` = '{$_POST['smoking']}', `alcohol` = '{$_POST['alcohol']}', `sims`

```

```
= '{$_POST['sims']}', `perezd` = '{$_POST['perezd']}', `koldetey` = '{$_POST['koldetey']}',
`kid` = '{$_POST['kid']}', `priemdetey` = '{$_POST['priemdetey']}', `socznach` =
 '{$_POST['socznach']}', `rassylka` = '{$_POST['rassylka']}', `databegin` =
 '{$_POST['databegin']}', `dataend` = '{$_POST['dataend']}', `inet` = '{$_POST['inet']}', `tel` =
 '{$_POST['tel']}', `showtel` = '{$_POST['showtel']}', `name` = '{$_POST['name']}', `surname` =
 '{$_POST['surname']}', `fathename` = '{$_POST['fathename']}', `adres` = '{$_POST['adres']}',
 `pasport` = '{$_POST['pasport']}', `hobbi` = '{$_POST['hobbi']}', `invalid` = '{$_POST['invalid']}',
 `invalid_desc` = '{$_POST['invalid_desc']}', `ogranich` = '{$_POST['ogranich']}', `target` =
 '{$_POST['target']}' WHERE `id` = '{$_POST['oldid']}";
```

```
print("<center><font class=\"good\"><a href=\"index.php?do=founum&num={$id}\">Абонент
№ {$id} успешно изменён.</a></font></center><br>");
$user=getabonent($who); }
```

Приведенный выше код потребовал усовершенствования путем применения стандартных и авторских подходов.

В результате нами реализованы некоторые подходы для защиты php кода и баз данных. Используются как базовые, так и дополнительные алгоритмы шифрования. Например, хэширующий алгоритм md5, для создания цифровых подписей, позволяющих однозначно идентифицировать отправителя. В PHP для указанного алгоритма существует специальная функция. Другие подходы по защите информации баз данных малых предприятий будут продемонстрированы в ходе доклада.

Литература

1. Охримчук, В.В. Организация работы малого предприятия с использованием автоматизированной системы управления «Брачное агентство» / В.В. Охримчук // Материалы II международной студенческой конференции «Студенческая наука – будущее государства» (Пинск, 25 марта 2008 года). – Пинск: Изд-во Полеский ГУ. – 2008. – Ч.2. – С. 103.
2. Охримчук, В.В. Информационные технологии WEB-программирования базы данных управления малым предприятием // Материалы IV международной научно-практической студенческой конференции «Содружество наук. Барановичи 2008» (Барановичи, 21 мая 2008 года). – Барановичи: Изд-во РИО БарГУ. – 2008. – Ч.1. – С. 42.
3. Соколов, А. Защита от компьютерного терроризма: справочное пособие / А. Соколов, О. Степанюк. – Арлит.: БХВ-Петербург, 2002. – С. 408 – 409.

УДК 519.171.2

НЕКОТОРЫЕ СПОСОБЫ РИСОВАНИЯ ГРАФОВ НА ПЛОСКОСТИ

Райченок С.В.

УО «Брестский государственный технический университет», г. Брест

Определение проблемы. Нарисовать граф красиво - это проблема, которая постоянно возникает в приложениях, таких как отображение файла или каталога деревьев, диаграммах в схемотехнике. Алгоритм должен таким образом отражать структуру графа, чтобы зритель мог лучше понять его.

Определим несколько более жестких критериев, чем понятие «красота», оценки качества рисования графов: минимизация числа пересечений, ограничение свободного места, минимизация изломов минимизация общей длины ребер.