Разработка и реализация технологии безопасного взаимодействия клиентских приложений с файлами и базами данных в SCADA «Систел»

Н. Ю. Кульман - к. ф.-м. н.,
Е.И. Пугачева,
Е.А. Михалевский
(ООО «Систел», г. Протвино)

Кульман Никита Юрьевич — зам. генерального директора ООО «Систел» по системам диспетчерского управления, канд. физ.-мат. наук. Kulman.nik@gmail.com.

Пугачева Екатерина Игоревна — начальник группы разработки программного обеспечения в ООО «Систел» Pugacheva.katerina@gmail.com

Михалевский Евгений Александрович — студент 4 курса Международного университета природы, общества и человека «Дубна», филиал Протвино. <Mikhea@rambler.ru>

Статья посвящена рассмотрению вопросов безопасного взаимодействия клиентских приложений с файлами, хранящимися на сервере, и системами управления базами данных. Рассматриваются проблемы информационной безопасности в системах промышленной автоматизации. Описывается реализация безопасного доступа к серверной информации в SCADA «Систел» [1].

Ключевые слова: *безопасность, промежуточный сервер, файловый сервис.*

На современном этапе становления информационного общества процесс информатизации является одним из основных факторов его развития. Благодаря этому процессу человек включается в глобальное информационное пространство, становясь при этом его частью. Это пространство связывает мир в единое целое и делает все государства информационно взаимозависимыми, происходит объединение всего человечества, что является, несомненно, большим плюсом, но в то же время оно таит в себе много опасностей мирового масштаба, связанных с обеспечением информационной безопасности.

Поскольку сегодняшний этап развития общества связан с освоением и использованием новых глобальных возможностей информационной сферы, таких как интернет, виртуальное пространство, новейших беспроводных средств коммуникации, а также создания искусственного интеллекта, постольку весьма актуальным становится проблема информационной безопасности, а в особенности становится актуальным осознание этой проблемы, обеспечение которой представляется очень сложным, многофункциональным процессом, зависящим от различных внешних и внутренних факторов. Особого внимания заслуживают проблемы влияния различных нововведений в области информационных технологий и знание основ использования этих средств в любой деятельности.

Обеспечивая беспрецедентные возможности накопления и использования информации, новые технологии и средства одновременно создают фундаментальную зависимость от их нормального функционирования всех сфер жизнедеятельности общества и государства.

Наглядными примерами, иллюстрирующими необходимость защиты информации и обеспечения информационной безопасности, являются участившиеся сообщения о компьютерных «взломах» банков, росте компьютерного пиратства, распространении компьютерных вирусов. Число компьютерных преступлений растет, и также увеличиваются их масштабы.

Одной из основных причин потерь, связанных с компьютерами, является недостаточная образованность в области безопасности. Информация — это сложное, многогранное всеобъемлющее явление; его отдельные стороны, грани выступают предметом исследования многих наук, которые, существуя самостоятельно, развиваются в неразрывном единстве, дополняя и обогащая друг друга. Под информационной безопасностью понимается защищенность информации от случайных или преднамеренных воздействий естественного или искусственного характера, чреватых нанесением ущерба владельцам или пользователям информации, поэтому в первую очередь необходимо обезопасить ценности системы, защитить и гарантировать точность и целостность информации и минимизировать разрушения, которые могут иметь место, если информация будет модифицирована или разрушена.

На практике информационная безопасность обычно рассматривается как совокупность следующих трёх базовых свойств защищаемой информации [2]:

- конфиденциальность, означающая, что доступ к информации могут получить только легальные пользователи;
- целостность, гарантирующая, что во-первых, защищаемая информация может быть изменена только законными и имеющими соответствующие полномочия пользователями, а во-вторых, информация внутренне непротиворечива и (если данное свойство применимо) отражает реальное положение вещей;
- доступность, гарантирующая беспрепятственный доступ к защищаемой информации для законных пользователей.

Современный компьютерный мир представляет собой разнообразную и весьма сложную совокупность вычислительных устройств, систем обработки информации, телекоммуникационных технологий, программного обеспечения и высокоэффективных средств его проектирования. Вся эта многогранная и взаимосвязанная система решает огромный круг проблем в различных областях человеческой деятельности. Чем сложнее задача автоматизации и чем ответственнее область, в которой используются компьютерные информационные технологии, тем все более и более критичными становятся свойства надежности и безопасности информационных ресурсов, задействованных в процессе сбора, накопления, обработки, передачи и хранения компьютерных данных. Конфиденциальность становится все более уязвимой по мере появления возможности доступа к постоянно растущим объемам информации.

Угрозы безопасности компьютерной системы следует внимательно анализировать, т. е. изучать потенциально возможные происшествия, которые могут оказать нежелательное воздействие на саму систему, а также на информацию, хранящуюся в ней. Различают следующие угрозы безопасности [3]:

- Угроза раскрытия заключается в том, что информация становится известной тому, кому не следовало бы ее знать. В терминах компьютерной безопасности угроза раскрытия имеет место всякий раз, когда получен доступ к некоторой конфиденциальной информации, хранящейся в вычислительной системе или передаваемой от одной системы к другой.
- Угроза целостности включает в себя любое умышленное изменение (модификацию или удаление) данных, хранящихся в вычислительной системе или передаваемых из одной системы в другую.
- Угроза отказа в обслуживании возникает всякий раз, когда в результате некоторых действий блокируется доступ к некоторому ресурсу вычислительной системы. Реально блокирование может быть постоянным, так, чтобы запрашиваемый ресурс никогда не был получен, или оно может вызвать только задержку запрашиваемого ресурса, достаточно долгую для того, чтобы он стал бесполезным.

Чтобы предупреждать и своевременно раскрывать компьютерные преступления, необходимо исследовать способы и средства их совершения.

1. УГРОЗЫ СУБД

Среди всех компонентов информационной системы любого предприятия или организации большую ценность представляет хранилище данных, так как значение информации как таковой, а также оперативного доступа к ней, во все времена было априори победоносным фактором, а накопленная информация, как правило, является конфиденциальной. Вполне очевидно, что обрабатывать, хранить и структурировать большие объемы информации без набора удобных, функциональных и надежных инструментов достаточно сложно. Такими инструментами являются системы управления базами данных (СУБД), использование которых оказалось чрезвычайно практичным методом работы с большими объемами данных, а реляционные СУБД стали доминирующим инструментом хранения таких массивов информации. Все современные приложения используют не файловые структуры операционных систем, а многопользовательские клиент/серверные СУБД. Из этого следует, что обеспечение информационной безопасности именно серверных компонентов базы данных приобретает решающее значение для безопасности организации в целом. Любая СУБД должна решать задачи организации доступа к данным и восстановления их в непредвиденных ситуациях. Именно поэтому ключевыми задачами являются обеспечение бесперебойного доступа ко всем ресурсам и уменьшение потерь данных, возникающих вследствие сбоев.

Главный источник угроз, специфичных для СУБД, лежит в самой природе баз данных. Для взаимодействия приложений с СУБД основным средством является язык структурированных запросов SQL - мощный непроцедурный инструмент для определения и манипулирования данными. В его состав входят хранимые процедуры, добавляющие к этой функциональности управляющие конструкции. В итоге злоумышленник может выполнить процедуру без наличия на то полномочий, используя предоставляющий возможность выстраивать сложные цепочки действий механизм правил. И, как результат, злоумышленник получает в свои руки мощный и удобный инструментарий, а все развитие СУБД направлено на то, чтобы сделать этот инструментарий еще мощнее и удобнее.

Все существующие СУБД подвержены SQL-инъекциям, наиболее опасным из видов атак. С их использованием злоумышленник может не только получить закрытую информацию из базы данных, но и, при определенных условиях, внести туда изменения. Атака, связанная с различного рода инъекциями, подразумевает внедрение сторонних команд или данных в работающую систему с целью изменения хода работы системы, а в результате — получение доступа к закрытым функциям и информации, либо дестабилизации работы системы в целом.

В зависимости от типа используемой СУБД и условий внедрения, атакующий может выполнить произвольный запрос к базе данных (например, прочитать содержимое любых таблиц, удалить, изменить или добавить данные), получить возможность чтения и/или записи локальных файлов и выполнения произвольных команд на атакуемом сервере. В случае недостаточной фильтрации входных данных в его руках оказывается полный доступ к базе данных и операционной системе сервера.

При использовании средств языка SQL могут возникнуть следующие угрозы:

- Получение информации путем логических выводов. Нередко путем логического вывода можно извлечь из базы данных информацию, на получение которой стандартными средствами у пользователя не хватает привилегий. Если пользователь имеет право доступа только к одной таблице, он, тем не менее, может получить косвенную информацию о другой таблице. Еще один пример - выяснение набора первичных ключей таблицы при наличии только привилегии INSERT. Если набор возможных значений ключей примерно известен, можно пытаться вставлять новые строки с интересующими ключами и анализировать коды завершения SQL-операторов.
- Агрегирование данных. Агрегирование - это метод получения новой информации путем комбинирования данных, добытых легальным образом из различных таблиц. Агрегированная информация может оказаться более секретной, чем каждый из компонентов, ее составивший. Повышение уровня секретности данных при агрегировании вполне естественно - это следствие закона перехода количества в качество. Бороться с агрегированием можно за счет тщательного проектирования модели данных и максимального ограничения доступа пользователей к информации.
- Покушения на высокую готовность (доступность). Если пользователю доступны все возможности SQL, он может довольно легко затруднить работу других пользователей (например, инициировав длительную транзакцию, захватывающую большое число таблиц).

Атака с помощью SQL-инъекции может быть возможна из-за некорректной обработки входных данных, используемых в SQL-запросах. Разработчик прикладных программ, работающих с базами данных, должен знать о таких уязвимостях и принимать меры противодействия внедрению SQL.

Современные многопотоковые серверы СУБД отражают лишь самые прямолинейные атаки. Для обеспечения безопасности системы настоятельно рекомендуется не предоставлять пользователям непосредственного SQL- доступа к базе данных, используя в качестве фильтров серверы приложений. Выбор подобной архитектуры разумен и по многим другим соображениям. Конфигурация, к которой имеет доступ хотя бы один программист, не может считаться безопасной. Поэтому обеспечение информационной безопасности баз данных - дело весьма сложное во многом в силу самой природы реляционных СУБД.

2. ВЗЛОМЫ

Назревшую проблему безопасности еще раз подтверждает статистика взломов самых мощных и распространенных систем. К ним относятся:

Взломы UNIX систем.

- 31 августа 2011 весь мир облетела новость о взломе kernel.org с текстами ядра Linux.
- Через неделю стало известно о взломе linux.com и linuxfoundation.org.
- В сентябре был взломан mysql.com.
- За последние месяцы в Android находят по несколько сотен уязвимостей ежемесячно. Большинство известных вредоносных приложений обычно маскируются под популярные программы или бесплатные версии платных приложений для этой системы. Так эксперты антивирусной компании Lookout обнаружили новую троянскую программу для Android, которая автоматически загружается на устройство пользователя при посещении зараженного сайта
- В начале апреля 2012г. специалистами компании «Доктор Веб» была обнаружена первая в истории масштабная бот-сеть, состоящая из компьютеров, работающих под управлением операционной системы Mac OS X. Всего было зарегистрировано 824 739 ботов, из них проявляло активность 334 592, поразивших более 800 000 работающих под управлением Mac OS X компьютеров.

Взломы банков.

- 2 ноября 2010 года компания «Доктор Веб» сообщила о широком распространении троянца Trojan.PWS.Multi.201, предназначенного для кражи параметров доступа к аккаунтам дистанционного банковского обслуживания (ДБО), принадлежащим юридическим лицам.
- Еженедельно (по данным на апрель 2012г.) с банковских карточек в России воруется примерно10 млн. рублей.

3. СЕРВЕР ПРИЛОЖЕНИЙ

Как видно из приведенных данных, проблема безопасности касается многих отраслей человеческой деятельности. Не обошла стороной она и системы промышленной автоматизации, в частности, SCADA системы. Поэтому в процессе эксплуатации программного комплекса SCADA «СИСТЕЛ» также назрела необходимость ограничения доступа конечных пользователей к СУБД. С этой целью авторами было предложено использовать сервер приложений, действующий как центральный узел, с помощью которого можно управлять доступом к базе данных, что является безусловным преимуществом защиты. Сервер приложений представляет собой сервер, предназначенный для выполнения прикладных процессов. Он взаимодействует с клиентами, получая задания, и взаимодействует с базой данных, выбирая данные, необходимые для обработки.

Известно, что классическая двухзвенная клиент-серверная модель подходит для небольших организаций с ограниченным числом пользователей и невысокой нагрузкой на сервер. По мере внедрения клиент-серверных технологий, обслуживающих многих пользователей и обрабатывающих существенные массивы данных, стали очевидны недостатки двухзвенных решений, таких как ограниченные возможности масштабирования и необходимость изменения клиентских приложений при изменении работы серверной логики.

Решить эти проблемы позволяет переход на трех- и многозвенные модели, в которых прикладная или бизнес-логика вынесена в отдельный уровень - сервер приложений. Это позволяет эффективно распределить нагрузку и обеспечить прозрачное наращивание как функциональности сервера приложений, так и числа обслуживаемых пользователей. В одном из случаев, собственное промежуточное ПО — это встроенные механизмы доступа для серверов баз данных. К этому виду как раз и относится поддержка стандартов языка SQL. Несмотря на то, что SQL является международным стандартом, разработчики СУБД самостоятельно определяют, какие из возможностей SQL они будут поддерживать в своих системах. В результате, конкретная СУБД может, с одной стороны, не поддерживать или поддерживать частично некоторые команды SQL, а с другой — представлять разработчикам приложений нестандартные языковые расширения.

В своей работе SCADA «СИСТЕЛ» использует конфигурационные и архивные БД. Конфигурационные БД, которые работают под управлением различных систем управления базами данных, MS SQL Server, Oracle, MS Access, являются важной частью SCADA-системы. Они определяют конфигурацию сервера SCADA «СИСТЕЛ» и содержат описание каналов связи, принимаемых данных, параметров их обработки и объединения их по различным признакам. Архивные БД включают в себя таблицы, в которых запоминаются все величины с определенным временным интервалом (порядка нескольких минут - суточный архив и диспетчерская ведомость с шагом полчаса), а также таблица со значениями по изменениям. Сервер также осуществляет ведение архивов данных, событий и действий диспетчерского персонала. Указание категорий оперативной информации, подлежащей записи в архив, проводится администратором в конфигурационной БД системы.

Для решения задачи доступа клиентского приложения к БД через сервер приложений был реализован соответствующий алгоритм взаимодействия (Рис. 1).

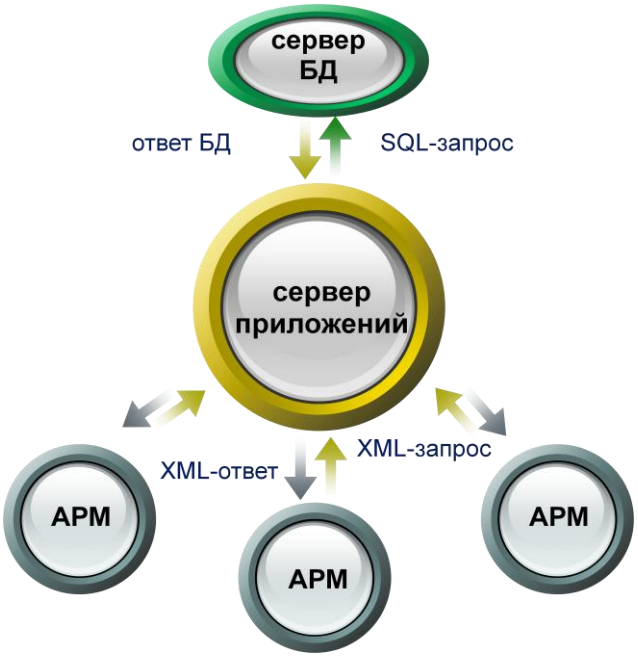

Рис.1 Взаимодействие клиентского приложения с сервером БД

Связь между клиентом и сервером поддерживается посредством передачи сообщений, содержащих XML-строку с запросом к БД (при передаче от клиента серверу) и ответа от базы данных (при передаче в обратном направлении) [4]. Передача данных осуществляется с использованием стека протоколов TCP/IP [5]. Полученная сервером XML-строка проверяется на корректность и преобразуется в SQL-запрос, который направляется БД. Полученный ответ снова преобразуется сервером в XML-строку и направляется клиенту. В связи с тем, что результат выполнения запроса может содержать большое количество данных, передача такой строки клиенту не всегда удобна. Для решения этой проблемы был применен алгоритм сжатия данных. Таким образом, сервер передает XML-строку в сжатом виде, а клиент, получив ее, производит распаковывание. В результате

выполнения данного алгоритма клиент получает необходимую информацию от базы данных.

4. ФАЙЛОВЫЙ СЕРВИС

Еще одной группой ресурсов, нуждающихся в безопасном доступе, являются файлы, содержащие мнемосхемы и другую информацию.

Изначально, для корректной работы комплекса было необходимо открывать всем пользователям доступ к серверу (на чтение), хранящему необходимые файлы (схемы и т.д.). Такое решение не отвечает требованиям безопасности, вследствие чего было решено разработать файловый сервис (Рис.2), обеспечивающий управление файлами, распределение прав доступа и упрощение процедуры резервного копирования.

Файловый сервис предназначен для разделения файлов с другими компьютерами, соединенными с ним через сеть, располагается одновременно в двух сетях и для доступа к файлам использует технологию WCF.

Windows Communication Foundation (WCF) — программный фреймворк, используемый для обмена данными между приложениями входящими в состав .NET Framework [6].

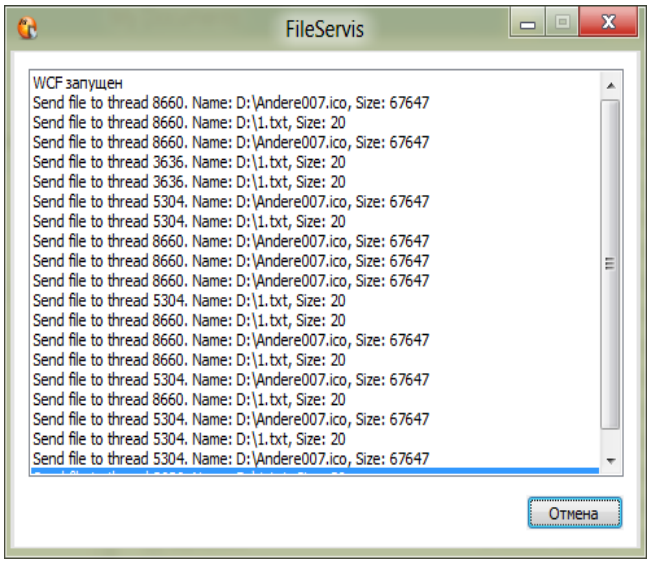

Рис.2 Файловый сервис

WCF делает возможным построение безопасных и надёжных транзакционных систем через упрощённую унифицированную программную модель межплатформенного взаимодействия. Комбинируя функциональность существующих технологий .NET по разработке распределённых приложений, WCF предоставляет единую инфраструктуру разработки, при умелом применении повышающую производительность и снижающую затраты на создание безопасных, надёжных и транзакционных служб нового поколения.

Традиционно безопасность сетевой информационной системы состоит из:

1. Аутентификации,
2. Авторизации пользователя на сервере и серверного приложения для клиента (проверка подлинности службы),
3. Безопасности передачи сообщений,
4. Выбора идентичности (личности) от имени которой на сервере будут выполняться операции,
5. Дополнительных мер безопасности в организации.

Технология WCF поддерживает ряд механизмов аутентификации, безопасности передачи, авторизации и идентичности. Необходимо только выбрать подходящую привязку и настроить её.

Первый механизм представляет собой процедуру, которая выясняет, является ли вызывающая сторона именно тем, за кого она себя выдаёт. WCF поддерживает ряд механизмов аутентификации:

- Без аутентификации – служба не проверяет от кого поступили вызовы. Обращение разрешено практически любому желающему.
- Аутентификация Windows – служба работает с удостоверениями клиента (credentials) и проверяет их средствами Windows.
- Имя пользователя и пароль – служба проверяет полученные имя и пароль по некоторому источнику, например, по учётным записям Windows или по пользовательской БД .
- Сертификат X509 – клиент идентифицирует себя при помощи сертификата. Служба обращается на сервере для подтверждения подлинности сертификата, а следовательно и клиента. Служба также может автоматически доверять стороне выдавшей сертификат.
- Пользовательский механизм – WCF позволяет заменить механизм аутентификации любым протоколом и типом удостоверений, например, биометрическими данными.
- Выдача электронного ключа – например, Windows CardSpace.

Авторизация определяет какие действия разрешены вызывающей стороне, обычно – какие операции службы разрешено вызывать клиенту. Авторизация бессмысленна без аутентификации. Служба должна найти роль вызывающей стороны в своём хранилище и убедиться в том, что вызывающая сторона принадлежит к запрашиваемой роли. Для этого WCF поддерживает несколько типов хранения авторизационных удостоверений. Наиболее популярные – это:

1. Группы (учётные записи) Windows.
2. Провайдеры ASP.NET для хранения пользовательских учётных записей.

WCF позволяет идентифицировать и проверять подлинность службы для защиты от фишинга. Удостоверение конечной точки службы генерируется WSDL-кодом службы и распространяется по всем прокси-клиентам. При связи со службой клиент

сравнивает её удостоверение с имеющимся действительным значением. Если значения совпадают, значит клиент связался с ожидаемой конечной точкой службы.

Выбор правила режима безопасности передачи является самым главным решением при защите службы. Необходимо обеспечить целостность сообщения, его конфиденциальность и взаимную аутентификацию клиента и сервера. Последнее должно предотвращать атаки замещения (злоумышленное повторное использование допустимого сообщения) и атаки на отказ в обслуживании.

WCF поддерживает следующие режимы передачи данных:

1. None – служба не получает удостоверение клиента, сообщения передаются открыто;
2. Transport – передача данных шифруется при HTTPS, TCP, IPC, MSMQ. Здесь обеспечивается целостность и конфиденциальность данных. Взаимная аутентификация реализуется шифрованием удостоверений клиента вместе с содержимым. Это самый простой и производительный способ гарантии безопасности при подключении клиента к службе без посредников;
3. Message – шифрование сообщения, обеспечивает конфиденциальность, целостность и проверку подлинности на уровне сообщений SOAP для небезопасных протоколов, таких как HTTP.
4. TransportWithMessageCredential – объединяет в себе два предыдущих режима.

При установке того или иного режима передачи данных указывается и тип учётных данных клиентов.

5. РЕЗЕРВИРОВАНИЕ СЕРВЕРОВ

Помимо вопросов безопасного доступа, файловый сервис решает вопрос резервирования серверов, который также является основополагающим фактором, влияющим на надежную работу всего комплекса. Для этого в конфигурационном файле сервиса прописываются адреса основного и резервного серверов, и при выходе из строя одного сервера, файловый сервис запросит необходимый файл у другого (Рис.3).

Спектр подходов к проблеме обеспечения информационной безопасности как субъекта, участвующего в информационных отношениях, так и самой информации, хранящейся на серверах и персональных компьютерах, достаточно широк. Не смотря на всё это, концепции информационной безопасности можно свести к общему знаменателю. Последний заключается в том, что грамотный разработчик сам в состоянии организовать свою информационную безопасность не только в сети и на персональном компьютере, но и в других ситуациях, связанных с воздействием на него и на технологии, участвующие в информационном процессе. Таким образом, чем выше уровень информационной культуры субъекта, тем меньше проблем у него возникает во время процесса обеспечения информационной безопасности.

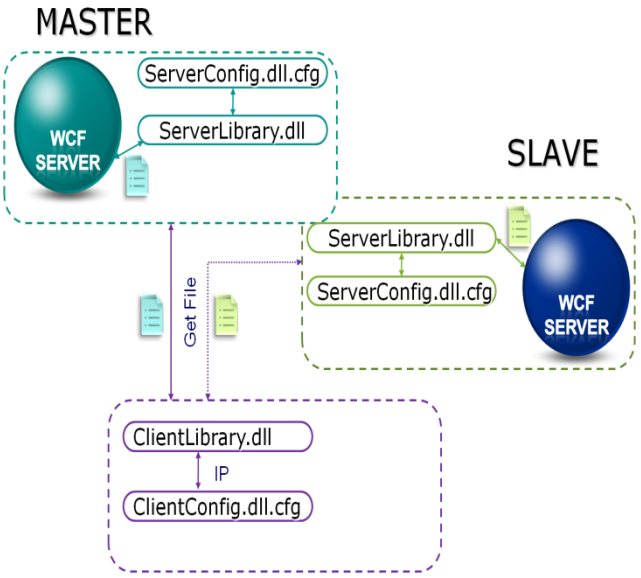

Рис.3 Резервирование серверов

СПИСОК ЛИТЕРАТУРЫ

[1] Рыкованов, С.Н. Оперативный информационный управляющий комплекс «Систел» / С.Н. Рыкованов, Н.Ю. Кульман, В.И. Ухов. Межотраслевой производственно-технический журнал «Автоматизация от А до Я», Минск, №1 (32), 2007 г., с.9-11.

[2] Цирлов, В.Л. Основы информационной безопасности автоматизированных систем / В. Л. Цирлов. Издательский дом «Феникс», 2008. – 173 с.

[3] Бармен, Скотт. Разработка правил информационной безопасности.: Пер. с англ. – М.: Издательский дом «Вильямс», 2002. – 208 с.:ил.

[4] Шеферд, Дж. Программирование на Microsoft Visual C++.NET / Дж. Шеферд. Пер. с англ. – М.: Русская редакция, 2003. – 928 с.: ил.

[5] Хортон, A.Visual C++ 2010: полный курс / А. Хортон. Пер. с англ. – М: ООО «И.Д.Вильямс», 2011. – 1216 с.: ил.

[6] Джувел Леве. Создание служб Windows Communication Foundation. – СПб.: Питер, 2008 . – 592 с.: ил.