На правах рукописи

Отавин Алексей Дмитриевич

ИНТЕГРАЦИОННЫЙ ПОДХОД К ПОСТРОЕНИЮ ЗАЩИЩЕННЫХ РАСПРЕДЕЛЕННЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

Специальность: 05.13.19 – " Методы и системы защиты информации, информационная безопасность"

Автореферат диссертации на соискание ученой степени кандидата технических наук

Санкт-Петербург – 2001

Работа выполнена в Санкт-Петербургском Государственном Техническом Университете.

Научный руководитель:
доктор технических наук, профессор Зегжда П.Д.

Официальные оппоненты:

доктор технических наук, профессор Корниенко А. А.

кандидат технических наук Карпов А.Г.

Ведущая организация:
Санкт-Петербургский Государственный Университет Телекоммуникаций им. проф. М.А. Бонч-Бруевича

Защита диссертации состоится "___"_____ 2002 г.
в ___ часов на заседании диссертационного совета Д 212.229.22 Санкт-Петербургского государственного технического университета по адресу: 195251, Санкт-Петербург, Тихорецкий пр., 21, ЦНИИ РТК.

С диссертацией можно ознакомиться в библиотеке Санкт-Петербургского государственного технического университета.

Автореферат разослан "___" _____ 2002 г.

Ученый секретарь
диссертационного совета
д.т.н., проф. Шашихин В.Н.

Общая характеристика работы

Актуальность

В последнее время распределенные вычислительные системы (РВС) все шире применяются не только в сфере обслуживания и управления производством, но и в государственной сфере. Обеспечение безопасности информации, хранящейся и обрабатываемой в государственных организациях, является национальной задачей, от выполнения которой зависит благополучие всей страны. Ни у кого не вызывает сомнения, что успешное ведение государственных дел, военных действий, торговли, банковской деятельности и личных дел во многом зависит и от уровня защищенности информационных систем. В этой связи разработка методов проектирования защищенных распределенных вычислительных систем (ЗРВС) представляется не только актуальной, но и необходимой.

В настоящий момент на мировом рынке программного обеспечения присутствует большое разнообразие прикладных программных систем обработки информации, которые могут удовлетворить нужды по обработке данных практически любых организаций. Однако отсутствие в этих прикладных системах функций защиты или же недоверие к реализации этих функций часто не позволяют использовать существующие прикладные средства в областях применения, требующих гарантий безопасности обрабатываемых данных. В данной работе предлагается разрешить данное противоречие путем разработки технологии интеграции средств защиты в существующие небезопасные, но удобные прикладные системы.

Теоретические основы методов разработки защищенных информационных систем рассматриваются в работах ведущих российских ученых В. А. Герасименко, С. П. Расторгуева, Л. М. Ухлинова, а также зарубежных К. Лендвера, Д. МакЛина, Р. Сандху и многих других. Проблемам обеспечения безопасности ВС с учетом современных требований к уровню защищенности посвящены исследования, проведенные компанией Secure Computing Corporation в рамках работ по развитию микроядра Mach и использования его в защищенной операционной системе (ОС) DTOS. Необходимо также отметить многочисленные работы по внедрению средств обеспечения безопасности в прикладные системы, проводимые практически всеми крупными компаниями разработчиками программного обеспечения (ПО). Примерами могут служить технология Common Data Security Architecture (CDSA), развиваемая фирмой Intel, подсистема безопасности CriptoAPI в операционных системах фирмы Microsoft, служба безопасности (Security Object Service) в стандарте CORBA консорциума OMG, технология управления рабочими станциями WfM (Wired for Management) фирмы Intel и др.

В настоящее время не существует общепринятого подхода к разработке ЗРВС и обоснованию архитектуры системы обеспечения информационной безопасности в ЗРВС. Это связано с тем, что на проектные решения оказывают влияние не только требования безопасности, но и взаимозависимость средств обработки информации и механизмов защиты, предъявляемые к системе функциональные требования и требования совместимости. В диссертационной

работе предложен и применен на практике интеграционный подход к построению ЗРВС, состоящий в интеграции в прикладную систему комплекса средств защиты. Интегрированные средства защиты позволяют обеспечить контроль доступа к информационным ресурсам системы, а также гарантировать целостность образа операционной среды и обеспечить возможность управления конфигурацией ПО автоматизированных рабочих мест (АРМ). Разработанный подход основывается на концепции разделения среды обработки и среды хранения информации, на абстракции информационного ресурса, на стандартизации и отделении средств защиты от прикладных средств, на разделении механизмов контроля доступа и реализации правил политики безопасности, а также на максимальной унификации всех взаимодействий в системе.

Выполнение указанных функций средств защиты не требует использования специальных средств обработки информации или же модификации стандартных прикладных средств. Более того, набор средств защиты и методы их внедрения выбраны таким образом, что они позволяют компенсировать уязвимости прикладных средств обработки. В результате этого, предоставляется возможность создания безопасных систем, в которых для обработки информации используется исходно небезопасное прикладное ПО.

Объектом исследований в данной работе является технология интеграции средств защиты в распределенную вычислительную систему.

Целью диссертационной работы является разработка способа построения защищенных распределенных вычислительных систем путем создания технологии интеграции в защищаемую систему средств защиты информации в виде средств контроля доступа, средств обеспечения целостности и средств управления конфигурацией ПО автоматизированных рабочих мест.

Для достижения поставленной цели в работе решались следующие задачи:

1. Анализ и обобщение существующих подходов к разработке ЗРВС.
2. Разработка технологии интеграции в РВС средств защиты информации.
3. Разработка метода интеграции в РВС средств контроля доступа к информационным ресурсам.
4. Разработка метода обеспечения целостности образа операционной среды АРМ.
5. Разработка метода управления конфигурацией ПО АРМ.
6. Практическое использование разработанной технологии интеграции средств защиты для построения ЗРВС.

Методы исследования

Для решения поставленных в работе задач использовались методы объектно-ориентированного анализа и моделирования, теории алгоритмов,

теории множеств, математической логики, дискретной математики, теории программирования и системного анализа.

Научная новизна диссертационной работы состоит в следующем:

1. Систематизированы подходы к разработке ЗРВС.
2. Сформулированы основные принципы технологии интеграции в РВС средств защиты информации.
3. Разработан метод интеграции в РВС средств контроля доступа к информационным ресурсам.
4. Разработан метод обеспечения целостности образа операционной среды АРМ.
5. Разработан метод управления конфигурацией ПО АРМ.
6. Разработана методика удаленной загрузки операционной среды на АРМ.

Практическая ценность работы определяется возможностью использования предложенной в ней технологии интеграции средств защиты и разработанных методов и методик для построения ЗРВС. К таким результатам относятся:

1. Разработка метода внедрения средств контроля доступа в ЗРВС (акт об использовании от НИИ системотехники ХК “Ленинец”).
2. Разработка метода управления конфигурацией АРМ (акт об использовании от в/ч 55342).
3. Разработка метода обеспечения целостности среды обработки информации (акт об использовании от в/ч 55342).
4. Разработка обобщенной методики удаленной загрузки операционной среды АРМ для различных операционных систем.
5. Разработка и программная реализация системы контроля доступа (СКД) к сетевым информационным ресурсам в системе обработки конфиденциальной информации на базе защищенной ОС Феникс.
6. Разработка и программная реализация службы удаленной принудительной загрузки операционной среды АРМ в системе обработки конфиденциальной информации на базе защищенной ОС Феникс.

Апробация работы. Основные теоретические и практические результаты работы обсуждались на республиканской научно-технической конференции “Методы и технические средства защиты информации” (1998–2001 гг.), на конференции “Информационная безопасность автоматизированных систем” в 1998 г, на международной научно-технической конференции студентов, аспирантов и молодых специалистов стран СНГ “Техника и технология связи” в 2000 г, на VIII-ой всероссийской научно-практической конференции “Проблемы информационной безопасности в системе высшей школы” в 2000 и 2001 гг, на ведомственной конференции “Проблемы обеспечения информационной безопасности на федеральном железнодорожном транспорте” в 2001 г, на международном симпозиуме MMM-ACNS “Сетевая безопасность:

методы, модели и архитектура” в 2001 г, на II-ой межрегиональной конференции "Информационная безопасность регионов России" в 2001 г.

Публикации. По теме диссертации опубликовано 16 работ, в том числе 12 научных статей и докладов, из них 2 на международных конференциях.

Основные положения, выносимые на защиту

1. Систематизация подходов к разработке ЗРВС.
2. Основные принципы технологии интеграции в РВС средств защиты информации.
3. Метод интеграции в РВС средств контроля доступа к информационным ресурсам.
4. Метод обеспечения целостности образа операционной среды АРМ.
5. Метод управления конфигурацией ПО АРМ.

Объем и структура диссертационной работы

Диссертационная работа состоит из введения, четырех глав, заключения и списка литературы.

Содержание работы

В первой главе рассматриваются проблемы разработки защищенных распределенных вычислительных систем (ЗРВС).

Методологической основой изучения подходов к разрешению проблем безопасности при распределенной обработке являются рекомендации международной организации по стандартизации (МОС), отраженные в эталонной модели взаимодействия открытых систем (ЭМВОС).

Современные системы обеспечения безопасности реализуют множество функций. Анализ требований показал, что система безопасности должна включать в себя следующие основные подсистемы (Рис. 1):

- идентификация и аутентификация;
- управление доступом;
- обеспечение конфиденциальности;
- протоколирование и наблюдение за выполняемыми операциями с ресурсами (аудит);
- обеспечение целостности программного обеспечения и данных.

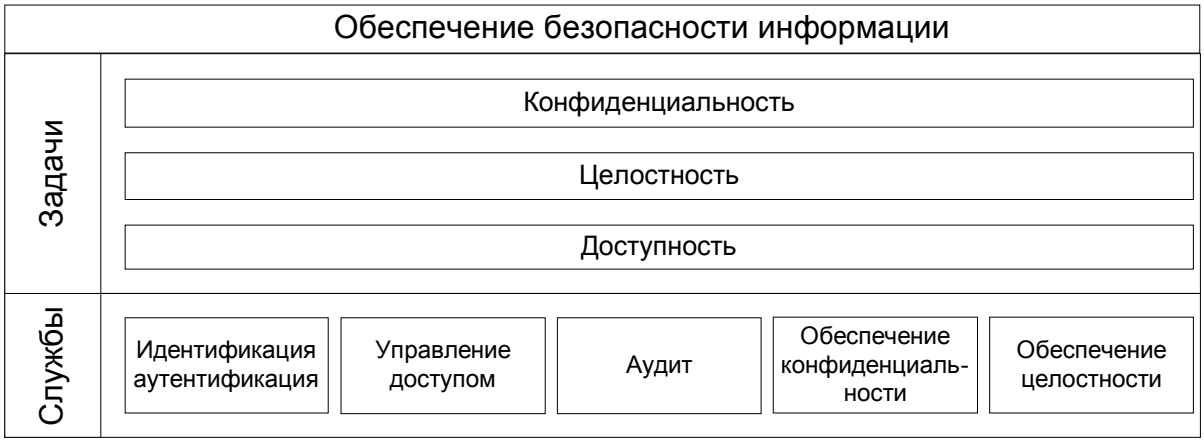

Рис. 1. Задачи и структура системы обеспечения безопасности информации

Проведенный анализ функций, механизмов, методов и средств обеспечения безопасности информации позволил сделать вывод о том, что только концепция комплексного подхода к защите информации может обеспечить современные требования к обеспечению безопасности информации. С точки зрения комплексного решения проблем обеспечения безопасности данных наибольший интерес представляют системы контроля доступа (СКД) и разграничения полномочий, как наиболее универсальные средства защиты информации.

Специфика построения защищенных систем обработки информации и разработки средств защиты для них состоит в том, что в силу тотального господства на отечественном и мировом рынке приложений, изначально лишенных функций защиты, попытки внедрения продуктов, не совместимых с этими популярными средствами обработки информации, не могут привести к успеху. Поэтому перед современными разработчиками защищенных информационных систем достаточно часто встает задача доработки уже существующей прикладной системы с целью повышения ее безопасности, что приводит к проблеме интеграции средств защиты в прикладную систему со средствами обработки информации, которые лишены функций защиты.

Большинство из используемых в настоящее время как автономных, так и сетевых операционных систем (ОС) были разработаны без учета требований к защите информации. Поэтому они оказались либо вообще незащищенными, либо средства защиты и контроля доступа в них играют роль дополнений к исходной системе. Операционные системы, в которых вопросы обеспечения безопасности информации находились под жестким контролем с самого начала их разработки, начали появляться только в последние 10 лет.

В настоящее время сложилась ситуация, для которой характерно сосуществования самых различных операционных систем в рамках одной РВС. Поэтому, остается актуальной задача обеспечения взаимодействия операционных систем с разной степенью защищенности. Например, на практике часто возникает необходимость обеспечить работу пользователя за рабочей станцией, на которой установлена небезопасная, но привычная для пользователя ОС.

На основании анализа существующих ЗРВС и методов их разработки выделены три основных подхода к построению защищенных систем (Рис.2):

1) Разработка и внедрение новых систем, в рамках которых решается весь комплекс проблем защиты информации (*креативный подход*).
2) Модификация существующих прикладных систем с целью дополнения их функциями защиты информации (*аддитивных подход*).
3) Разработка подсистем (продуктов) защиты информации, решающих отдельные задачи обеспечения безопасности данных, и их интеграция с существующими прикладными системами (*интеграционный подход*).

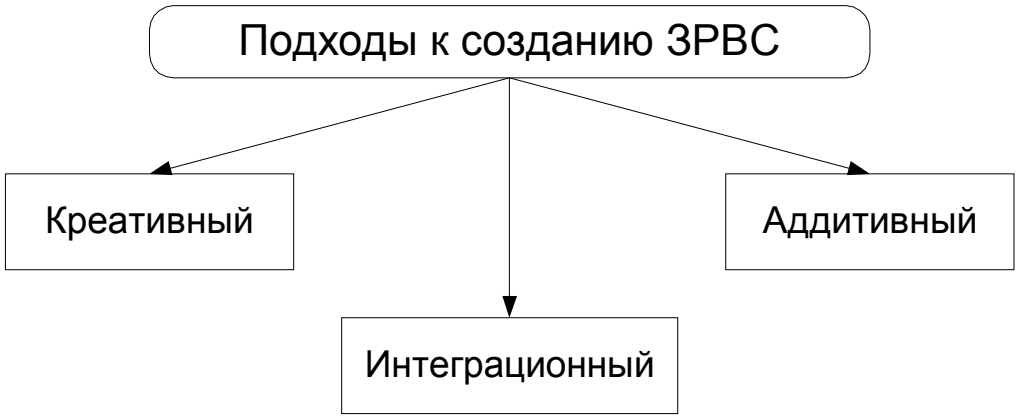

Рис.2. Систематизация подходов к созданию ЗРВС

Креативный подход предлагает наиболее радикальный способ решения проблемы обеспечения безопасности информации. Однако построение сложной информационной системы “с нуля” требует больших временных и финансовых затрат. В связи с этим широкое распространение получил аддитивный подход к построению защищенных систем. Применение данного подхода позволяет сократить время на разработку защищенной системы за счет использования готовых прикладных средств обработки информации. Однако, анализ существующих разработок, построенных на основе аддитивного подхода, показывает высокую уязвимость подобных систем. Это, в частности, связано с многовариантностью путей обмена информацией в современных прикладных системах, что не позволяет обеспечить надежный контроль над всеми информационными потоками.

Интеграционный подход является развитием этих двух подходов. Его качественно отличие от рассмотренных ранее состоит в том, что система создается путем интеграции стандартных программ, выполняющих прикладные функции, со средствами (продуктами) защиты. При использовании интеграционного подхода, в отличие от креативного, система создается не “с нуля”, а строится на основе готовых блоков интеграции, что заметно снижает трудоемкость разработки. Коренное отличие от аддитивного подхода состоит в самой технологии объединения прикладных средств и средств защиты, которая подразумевает органичную плановую интеграцию средств защиты в прикладную систему. Таким образом, сохраняя преимущества как аддитивного (совместимость со стандартным ПО) так и креативного (системный подход к

построению архитектуры безопасности) подходов, интеграционный подход позволяет успешно бороться с некоторыми присущими им недостатками.

В настоящий момент отсутствует достаточная методологическая база интеграции системы защиты с прикладными системами. Первоочередными задачами в этой области являются разработка соответствующей архитектуры безопасности ЗРВС, разработка методов и технологий интеграции и накопление банка универсальных подсистем (продуктов) защиты для внедрения в различные прикладные системы.

Основной задачей настоящей работы являлась разработка технологии интеграции в РВС средств защиты информации. Разработанная технология интеграции позволяет:

- повысить безопасность эксплуатирующихся в настоящее время систем;
- обеспечить возможность разработки защищенных систем на основе существующих незащищенных систем;
- сохранять совместимость защищенных систем с популярными средствами обработки информации;
- повысить технологичность создания защищенных систем;
- стандартизировать интерфейсы подсистем защиты.

Во второй главе изложены основы предлагаемой технологии интеграции в РВС средств защиты информации.

Предлагаемая технология интеграции основывается на следующих основных принципах:

1) разделение среды обработки и среды хранения информации;
2) унификация всех взаимодействий в системе;
3) абстрагирование информационных ресурсов;
4) выделение средств защиты в отдельные подсистемы, отделенные, как от прикладных средств обработки информации, так и от операционной системы;
5) стандартизация интерфейса взаимодействия подсистем защиты с другими подсистемами;
6) централизация схемы авторизации доступа;
7) обеспечение прозрачности механизмов контроля доступа;
8) отделение механизмов управления доступом от логики политики безопасности.

Разделение среды обработки и среды хранения информации обусловлено необходимостью перехвата обращений к информационным ресурсам и управления доступа к ним. Принципиальная возможность внедрения механизмов управления потоком обращений зависит от способа взаимодействия компонентов в системе. Для гарантированного перехвата информационного потока необходимо, чтобы все взаимодействие между средой обработки и средой хранения информации осуществлялось только на основе механизма передачи сообщений и не как иначе.

Унификация взаимодействий в системе позволит упростить реализацию механизма управления доступом, и соответственно снизить вероятность ошибок в реализации данного компонента. Унификация работы с ресурсами различных типов основана на абстракции информационного ресурса.

Реализация средств защиты в виде отдельных подсистем позволяет разрабатывать их независимо от модулей, реализующих прикладные функции. При этом требуется стандартизация интерфейса взаимодействия подсистем защиты между собой и с другими подсистемами

Анализ мировых стандартов безопасности показывает, что все средства защиты могут быть предоставлены на прикладном уровне ЭМВОС (а некоторые средства, например, функция контроля за участниками взаимодействия, могут быть предоставлены только на прикладном уровне).

Представление средств защиты на прикладном уровне дает возможность реализовать средства обеспечения безопасности вне рамок операционных систем. Поэтому разработчики приложений или опытные пользователи могут реализовывать механизмы защиты внутри приложений, не полагаясь на производителей конечных систем. Недостаток реализации средств и механизмов защиты внутри приложений состоит в том, что они могут стать специфичными для конкретного приложения, что не позволит повторно использовать уже разработанные механизмы защиты и ведет к дублированию разработок. Более того, так как разработка и реализация механизмов защиты – это достаточно сложный процесс, реализация функций защиты в рамках приложений увеличивает риск наличия ошибок в каждом конкретном продукте.

Для устранения данного недостатка предлагается предоставлять приложениям доступ к функциям защиты, используя общий стандартный интерфейс сервиса прикладного уровня, что позволит избежать дублирования реализаций, а также решить проблему добавления новых средств защиты в систему.

Прозрачность механизмов контроля доступа позволяет разрабатывать подобные средства защиты независимо от используемых прикладных средств.

Разделения механизмов управления доступом и правил политики безопасности позволяет обеспечить инвариантность механизмов контроля доступа по отношению к политике безопасности, что в свою очередь дает возможность поддерживать широкий класс моделей разграничения доступа.

Предлагаемая технология интеграции средств защиты является достаточно универсальной. Ее применение возможно в различных информационных системах, обмен информацией в которых происходит на основе передачи сообщений. В этом случае возможно разделение прикладной системы на среду обработки и среду хранения информации с внедрением механизмов перехвата обращений к ресурсам. Поэтому, данная технология с равным успехом может применяться как при разработке РВС, так и при построении защищенных ОС и сложных прикладных программных комплексов.

Использование средств защиты в виде самостоятельных служб, реализованных вне рамок операционных систем, требует разработки

специальных методов, позволяющих интегрировать такие службы в прикладную систему.

В третьей главе рассматриваются основные методы интеграции в РВС средств защиты информации.

Рассмотрим модель прикладной системы, в которую предполагается внедрение средств защиты. Введем понятия субъекта, объекта и пользователя системы. Под *пользователем* будем понимать лицо (физическое лицо), аутентифицируемое некоторой информацией и управляющее субъектами системы через органы управления компьютером. *Субъект доступа* – активный ресурс, осуществляющий какие-либо действия над другими ресурсами. *Объект доступа* – пассивный ресурс, используемый субъектом доступа для выполнения операций. Обозначим через $\boldsymbol{O}=\{o_i\}$ множество всех объектов системы. Назовем данное множество средой хранения информации. Обозначим через $\boldsymbol{S}=\{s_i\}$ множество всех субъектов системы. Назовем данное множество средой обработки информации.

Взаимодействие субъектов и объектов системы осуществляется путем посылки субъектами сообщений объектам. Пусть $\boldsymbol{P}$ – множество сообщений между субъектами и объектами. Данное множество разобьем на два непересекающихся подмножества $\boldsymbol{P}=\boldsymbol{N}\cup\boldsymbol{L}$, $\boldsymbol{N}\cap\boldsymbol{L}=\varnothing$, где $\boldsymbol{N}$ – множество сообщений, характеризующее несанкционированный доступ, $\boldsymbol{L}$ – множество легальных сообщений.

Критерий разбиения на множества $\boldsymbol{N}$ и $\boldsymbol{L}$ определяет заданная политика безопасности. Правила разграничения доступа субъектов к объектам есть формально описанные сообщения, принадлежащие множеству $\boldsymbol{L}$. Политика безопасности должна включать:

- множество методов доступа $\boldsymbol{A}=\{a_i\}$;
- для каждой пары “субъект, объект” (s_i, o_j) подмножество $\boldsymbol{A'}, \boldsymbol{A'}\subseteq\boldsymbol{A}$ методов доступа, которые разрешены для данной пары.

Введем в систему специальный субъект – монитор безопасности объектов. Монитор безопасности объектов – это монитор обращений, который разрешает только сообщения, принадлежащие множеству легальных сообщений $\boldsymbol{L}$. Основной операцией, выполняемой монитором безопасности объектов, является проверка каждого отдельного сообщения:

$r = Access(s_i, o_j, a_k)$,

где $a_k\in\boldsymbol{A}$ – конкретный тип доступа из множества допустимых;

$r\in\{\text{TRUE, FALSE}\}$ – решение о предоставлении доступа, причем r = TRUE, если доступ a_i субъекта s_i к объекту o_j разрешен и r = FALSE в противном случае.

Для принятия решений о предоставлении доступа требуется база данных политики безопасности, в которой хранятся атрибуты безопасности всех субъектов и объектов, а также сами правила разграничения доступа. Вся информация о работе средств защиты протоколируется в журнале аудита.

Полученная в результате модель прикладной системы с внедренными средствами контроля доступа приведена на Рис.3.

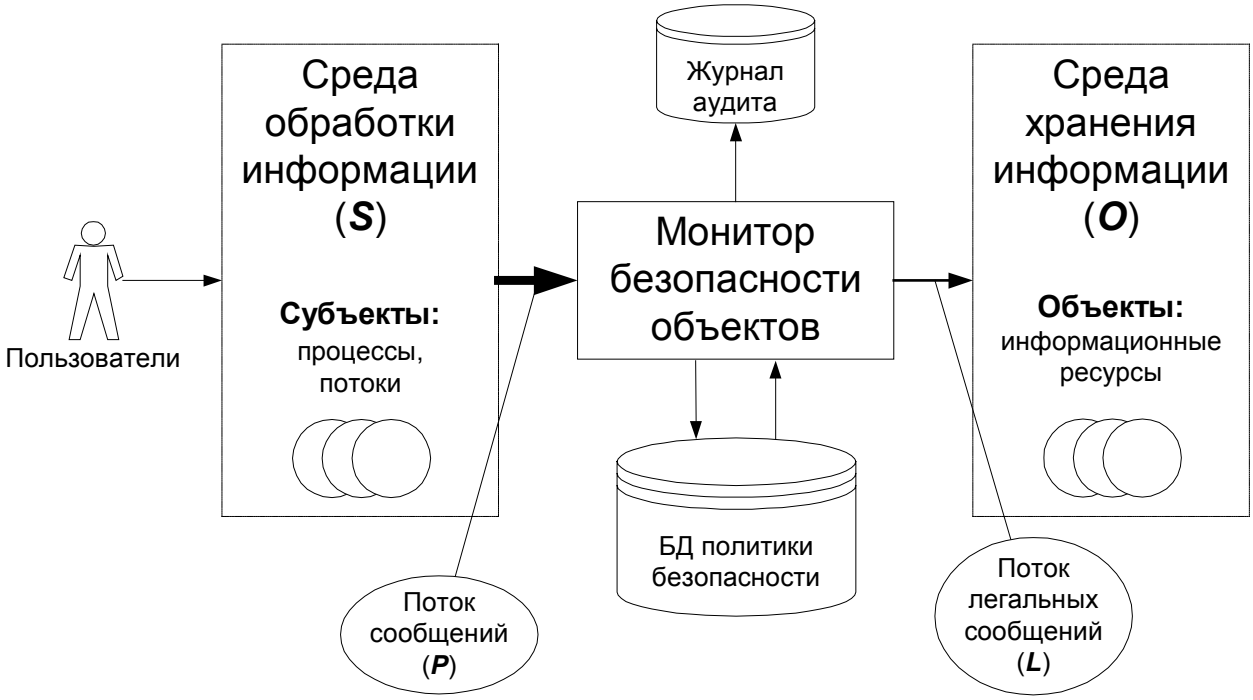

Рис.3. Модель прикладной системы с внедренными средствами контроля доступа

Для предложенной модели системы на основании анализа формальных моделей безопасности (ФМБ) сделан вывод о возможности внедрения ФМБ, построенных по принципу предоставления прав доступа. В частности, данный класс ФМБ включает в себя модели дискреционного и мандатного доступа, использование которых регламентируется требованиями информационной безопасности. Такие ФМБ определяются с использованием терминов субъект, объект и метод доступа.

Для разных ФМБ функция *Access*() реализуется по-своему. Например, для дискреционной модели разграничения доступа $Access(s_i, o_j, a_k) = M_{i,j,k}$, где $M_{i,j,k} \in \{\text{TRUE, FALSE}\}$ – трехмерная матрица доступа.

На основе рассмотренной модели системы сформулированы следующие требования к РВС, соблюдение которых позволяет интегрировать в систему средства контроля доступа:

1. Разделение среды обработки и среды хранения информации.
2. Взаимодействие между средой обработки и средой хранения информации только на основе передачи сообщений.
3. Идентифицируемость всех объектов и субъектов системы.
4. Задание множества возможных типов сообщений от субъектов к объектам (операций).
5. Представимость каждого сообщения субъекта к объекту в терминах методов доступа, т.е. существование однозначного отображения операций на методы доступа.

Для решения задач обеспечения целостности среды обработки информации и для управления конфигурацией ПО рабочих станций выбран метод принудительной загрузки операционной среды на АРМ.

Применение принудительной загрузки операционной среды на АРМ позволяет обеспечить целостность образа среды обработки информации, а именно – целостность программ обработки данных и целостность начальной конфигурации АРМ. Поскольку образ операционной среды хранится на сервере, то при старте АРМ происходит считывание данного образа с сервера, проверка его на целостность и загрузка в память станции всего необходимого системного и прикладного программного обеспечения. Выполнение этой процедуры гарантирует, что при каждом включении станция будет находиться в безопасном начальном состоянии (ПО станции находится в заданной администратором конфигурации и целостность его проверена).

Принудительная загрузка операционной среды АРМ может также применяться и для управления конфигурацией ПО АРМ. На практике часто возникает ситуация, когда за одним и тем же рабочим местом в разное время должны работать разные операторы, т.е. существует необходимость организации в системе многопользовательских терминалов. При этом каждому оператору могут быть необходимы различные программы обработки с различными правами на доступ к информации. Конфигурация и состав ПО на АРМ для каждого из операторов в этом случае должны быть различными. Применение принудительной загрузки операционной среды АРМ позволяет загружать на АРМ необходимую для конкретного оператора среду обработки информации, для чего на сервере организуется база образов программного обеспечения АРМ для различных операторов. В ходе старта АРМ оператор идентифицируется и на рабочую станцию передается образ с соответствующей конфигурацией и составом ПО.

Для описания методов обеспечения целостности образа операционной среды АРМ и управления конфигурацией ПО АРМ дополним рассмотренную модель защищаемой системы. Пусть $\boldsymbol{U}=\{u_i\}$ множество пользователей системы, $\boldsymbol{T}=\{t_i\}$ – множество терминалов РВС, $\boldsymbol{OS}=\{os_i\}$, $\boldsymbol{OS}\subseteq\boldsymbol{O}$ – множество образов операционных сред, $\boldsymbol{Z}=\{z_i\}$ – множество операционных сред.

Введем в систему специальный субъект – сервер загрузки ПО АРМ. Сервер загрузки ПО АРМ – это подсистема, которая осуществляет загрузку операционной среды на АРМ. Основной функцией, выполняемой сервером загрузки ПО АРМ, является порождение операционной среды АРМ на основе информации о пользователе, терминале, с которого инициирована загрузка, и заранее подготовленного образа операционной среды АРМ:

$Create(t_m, u_l, os_i) \rightarrow z_k$

Архитектура подсистемы загрузки операционной среды на АРМ приведена на Рис. 4.

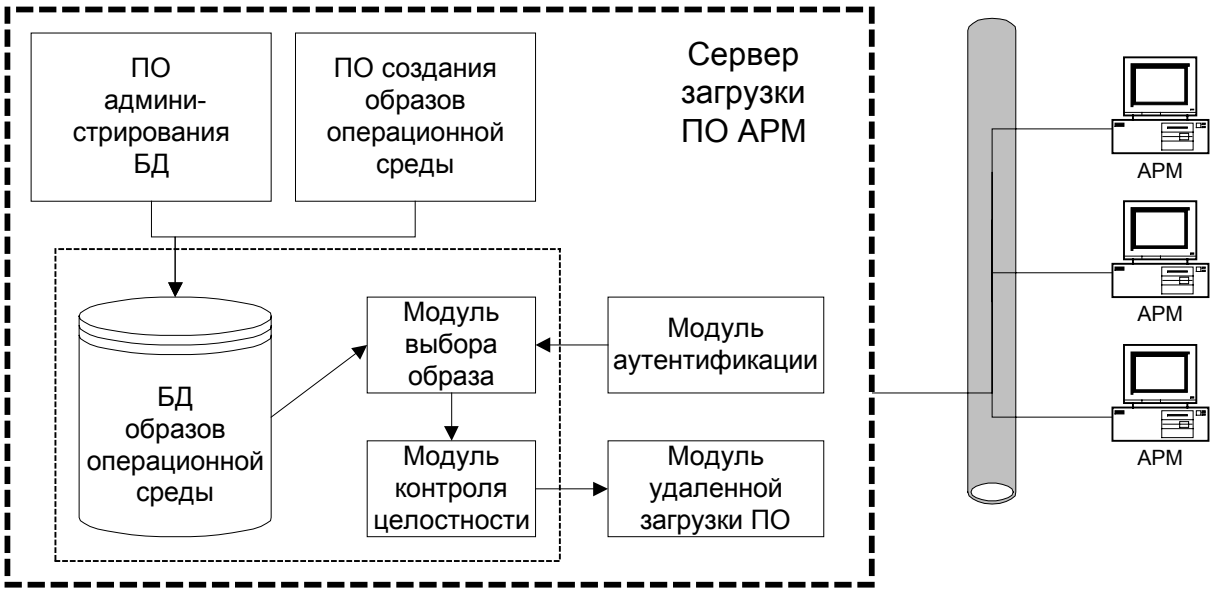

Рис. 4. Архитектура подсистемы принудительной загрузки операционной среды АРМ

Алгоритм загрузки операционной среды на АРМ включает в себя следующие основные шаги:

1) идентификация терминала t_m;
2) идентификация пользователя u_l;
3) проверка прав пользователя по использованию АРМ t_m;
4) выбор образа операционной среды для загрузки os_i;
5) контроль целостности образа операционной среды os_i;
6) очистка остаточной памяти терминала t_m;
7) принудительная загрузка проверенной на целостность операционной среды os_i на терминал t_m.

Выполнение данного алгоритма позволяет обеспечить целостность образа операционной среды АРМ путем хранения данного образа на сервере и принудительной загрузки данного образа в память АРМ при каждом старте станции.

Для целей управления конфигурацией ПО АРМ введем функцию выбора образа операционной среды (шаг 4 алгоритма) в зависимости от конкретного пользователя и терминала $os(t_m, u_l)$. Тогда функция порождения операционной среды АРМ примет следующий вид:

$Create(t_m, u_l, os(t_m, u_l)) \rightarrow z_k$

Такая функция дает возможность управления конфигурацией ПО АРМ путем принудительной загрузки на АРМ образа с персональной конфигурацией пользователя.

Принудительная загрузка операционной среды АРМ наиболее просто реализуется на основе технологии удаленной загрузки. В настоящее время получили распространение различные не совместимые между собой технологии удаленной загрузки. Обычно эти технологии привязаны к определенной версии и типу серверной ОС. В ходе удаленной загрузки применяются сетевые

протоколы передачи начального загрузочного образа на рабочую станцию, которые используются на начальном этапе загрузки, и протоколы доступа к сетевым ресурсам, которые используются на завершающих стадиях загрузки и в ходе работы пользователя.

Протоколы передачи начального загрузочного образа реализуются специальным сетевым оборудованием станции (например, прошиваются в BootPROM сетевой карты) и используются для получения различных параметров загрузки и считывания образа начальной загрузки. В качестве протоколов начальной загрузки используются протоколы BOOTP/DHCP/TFTP (в ОС UNIX), NCP (в ОС Novell Netware) и DCL RPL (в ОС Microsoft Windows NT). В последнее время наметилась тенденция к стандартизации, результатом которой стала публикация перспективного стандарта PXE, однако поддержка этого стандарта производителями большинства ОС еще не реализована.

Протоколы доступа к сетевым ресурсам – это обычно более сложные прикладные протоколы, используемые в ходе обычной работы пользователя в сети. В качестве протоколов доступа к сетевым файловым ресурсам используются протоколы: NFS в сетях UNIX, NCP в ОС Novell Netware и SMB (CIFS) в Microsoft Windows.

Таким образом, выбор конкретной технологии удаленной загрузки зависит от возможностей операционных систем сервера и клиента, а также используемых сетевых протоколов. В работе обобщены известные в настоящее время технологии удаленной загрузки рабочих станций и в результате их анализа разработана методика удаленной загрузки для широкого класса систем. Данная методика включает в себя следующие этапы (Рис. 5):

1) *Инициация удаленной загрузки.* Данный этап выполняется аппаратурой удаленной загрузки. В ходе этапа выполняются следующие основные действия: поиск сервера в сети, идентификация терминала, получение параметров начальной загрузки с сервера (сетевое имя станции, сетевой адрес и др.), получение образа начальной загрузки и передача ему управления.
2) *Запуск начального загрузчика.* Данный этап выполняется кодом начального загрузчика, полученным на предыдущем этапе с сервера. В ходе этапа выполняются следующие основные действия: дополнительная идентификация и аутентификация терминала, идентификация и аутентификация пользователя, инициация загрузки и подготовка к запуску на станции штатной оболочки для данного пользователя.
3) *Запуск оболочки рабочей станции.* Данный этап выполняется кодом оболочки рабочей станции. В ходе этапа выполняется загрузка ядра операционной системы, а также дополнительная идентификация и аутентификация пользователя, загрузка рабочего стека протоколов, после чего инициируется процесс настойки персональной конфигурации ПО пользователя.
4) *Настройка персональной конфигурации ПО пользователя.* Данный этап выполняется кодом оболочки рабочей станции. В ходе этапа выполняются следующие основные действия: подключение сетевых

ресурсов, окончательная загрузка и настройка среды работы пользователя.

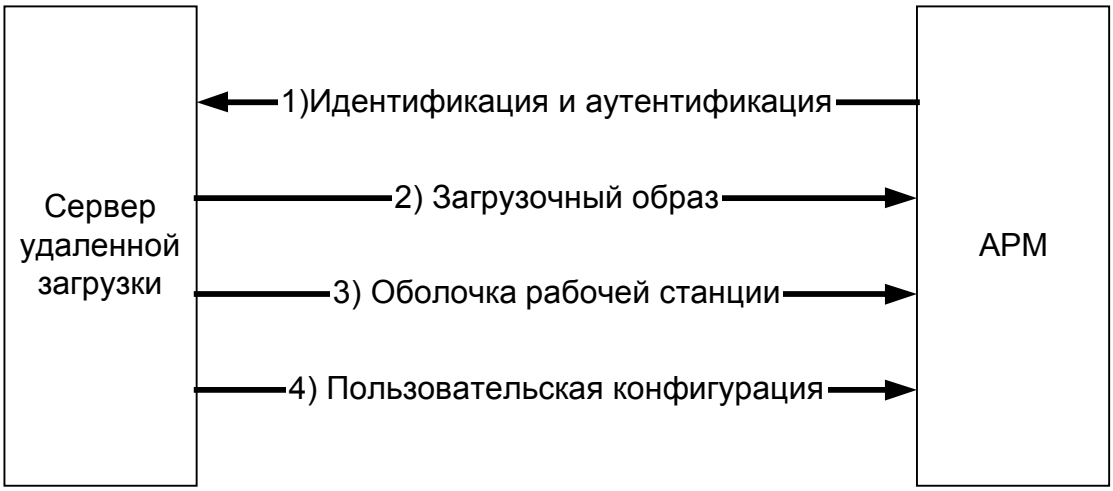

Рис. 5 Методика удаленной загрузки операционной среды АРМ

Применение данной методики позволяет универсальным образом осуществлять загрузку операционной среды на АРМ в гетерогенной среде. Этапы методики выбраны таким образом, что дают возможность осуществлять загрузку на станцию набора ПО, зависящего как от аппаратной конфигурации станции, так и от конкретного пользователя, инициировавшего запуск АРМ. Это позволяет применять данную методику как способ загрузки операционной среды при реализации функции управления конфигурацией ПО АРМ.

В четвертой главе описывается пример применения предлагаемой технологии построения ЗРВС на основе интеграционного подхода. Показано, что предлагаемый интеграционный подход особенно эффективен в сочетании с использованием защищенной ОС (ЗОС), например, МСВС, Феникс, DTOS.

В качестве примера применения подхода описывается проект “Феникс ЛВС”, в котором были использованы полученные в диссертации результаты. Проект был реализован автором в составе коллектива разработчиков Специализированного Центра Защиты Информации СПбГТУ. В задачу проекта входило построение ЛВС, предназначенной для организации защищенного документооборота и состоящей из сервера, работающего под управлением специализированной ЗОС Феникс, и АРМ, работающих под управлением популярной пользовательской ОС MS Windows 95.

В данном проекте необходимо было обеспечить возможность работы пользователя с защищенным документооборотом за рабочими станциями с незащищенной операционной системой. Для решения задач защиты информации в данной РВС была применена технология и методы, разработанные в диссертационной работе.

В проекте “Феникс ЛВС” данный подход к построению защищенных систем был реализован следующим образом:

1. Все информационные ресурсы хранятся централизованно на сервере, то есть сервер является средой хранения информации.
2. Обработка информации осуществляется пользователями с Windows-терминалов, то есть совокупность Windows-терминалов является средой обработки информации.
3. Весь информационный обмен между средой хранения и средой обработки информации осуществляется путем обмена сообщениями по сети.
4. Весь информационный обмен между средой хранения и средой обработки информации перехватывается ЗОС Феникс, которая выполняет роль монитора безопасности объектов.
5. Доступ пользователя к информационным ресурсам контролируется специальным компонентом ЗОС Феникс (сервер безопасности), который реализует политику безопасности.
6. На севере организовано хранение базы образов операционных сред.
7. При каждом старте рабочей станции осуществляется очистка остаточной памяти данной станции и формирование персональной операционной среды на данной станции путем удаленной загрузки.

Сервер в “Феникс ЛВС” выполняет несколько ролей: сервер информационных ресурсов, сервер безопасности (монитор безопасности объектов, хранение и реализация политики безопасности), сервер удаленной загрузки и сервер конфигурации АРМ.

Все Windows-терминалы в “Феникс ЛВС” являются бездисковыми станциями, это позволяет гарантировать, что все информационные ресурсы хранятся только на сервере. Использование бездисковых рабочих станций в “Феникс- ЛВС” позволило удовлетворить высоким требованиям к безопасности информации и в то же время использовать привычный для пользователей графический интерфейс, а также сохранить совместимость с популярными средствами обработки документов (MS Office).

Описанный проект неоднократно демонстрировался на различных конференциях и семинарах, проводимых Специализированным Центром Защиты Информации СПбГТУ и прошел государственные испытания в в/ч 43753.

В работе получены следующие основные результаты:

1. Обобщены и систематизированы существующие подходы к построению ЗРВС.
2. Обоснован интеграционный подход к построению ЗРВС и сформулированы основные принципы технологии интеграции в РВС средств защиты информации.
3. Разработан метод интеграции в РВС средств контроля доступа к информационным ресурсам.
4. Разработан метод обеспечения целостности образа операционной среды АРМ.
5. Разработан метод управления конфигурацией ПО АРМ.
6. Реализована ЗРВС на основе интеграционного подхода.

Основные результаты диссертации изложены в 16 печатных работах. Ниже приведены основные из них:

1. Отавин А.Д. Объектный подход к обмену информацией в защищенных информационных системах // Проблемы информационной безопасности. Компьютерные системы. – СПб., 1999. – № 1. – С.56-58
2. Отавин А.Д. Безопасность в системах клиент-сервер: технологии DCOM и CORBA // Вестник связи. – СПб., 1999. – № 4. – С.109-115
3. Отавин А.Д. Организация безопасного взаимодействия в IP-сетях // Проблемы информационной безопасности. Компьютерные системы. – СПб., 1999. – № 2. – С.86-88
4. Отавин А.Д. Перспективы разработки и применения доверенных систем обработки информации // 2-я международная научно-техническая конференция студентов, аспирантов и молодых специалистов стран СНГ "Техника и технология связи". Тез. док. – СПб., 2000. – С.405-406
5. Отавин А.Д. Удаленная загрузка рабочих станций в ЛВС Феникс // Проблемы информационной безопасности. Компьютерные системы. – СПб., 2000. – № 3. – С.51-54
6. Отавин А.Д. Использование тонких клиентов для повышения безопасности систем обработки информации // Российская научно-техническая конференция "Методы и технические средства защиты информации". Тез. докл. Под ред. П.Д. Зегжды. – СПб.: СПбГТУ, 2000. – С.169-173
7. Отавин А.Д. О возможностях использования небезопасных клиентов для обработки информации в защищенных информационных системах // VIII всероссийская научно-практическая конференция "Проблемы информационной безопасности в системе высшей школы". Тез. докл. – М., 2001.
8. Зегжда П.Д., Отавин А.Д., Построение безопасных систем обработки информации на базе защищенной операционной системы // Первая ведомственная конференция "Проблемы обеспечения информационной безопасности на федеральном железнодорожном транспорте". Тез. докл. – СПб., 2001. – С.127-130
9. Dmitry P. Zegzhda, Pavel G. Stepanov, Alexey D. Otavin, Fenix Secure Operating System: Principles, Models, and Architecture // International Workshop MMM-ACNS 2001 Information Assurance in Computer Networks: Methods, Models, and Architectures for Network Security, St. Petersburg, Russia, May 21-23, 2001.
10. Отавин А.Д. Анализ методов разработки защищенных информационно-вычислительных систем // Российская научно-техническая конференция "Методы и технические средства защиты информации". Тез. докл. Под ред. П.Д. Зегжды. – СПб.: СПбГТУ, 2001. – С.179-181
11. Отавин А.Д. Интеграционный подход к построению защищенных информационных систем // II межрегиональная конференция "Информационная безопасность регионов России" ИБРР-2001. Тез. докл. – СПб., 2001.