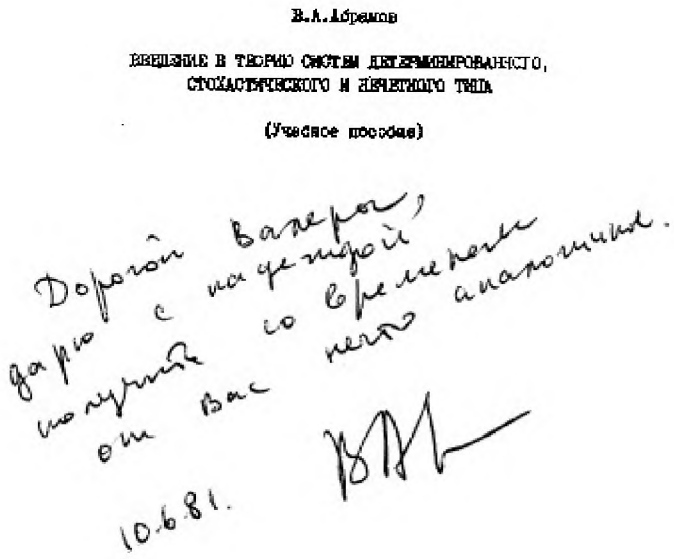
УПРАВЛЕНИЕ ЗАЩИТОЙ ИНФОРМАЦИИ НА БАЗЕ СЗИ НСД «АККОРД»

Светлой памяти
моего учителя профессора
Владимира Абрамовича Абрамова,
щедро даривше го нам
свои мудрость, талант
и книги:


В.А.Абрамов

ВВЕДЕНИЕ В ТЕОРИЮ СИСТЕМ ДЕТЕРМИНИРОВАННОГО,
СТОХАСТИЧЕСКОГО И НЕЧЕТКОГО ТИПА

(Учебное пособие)

Дорогой Валера,
дарю с надеждой,
что будешь
он Вас со временем
10.6.81.

В.А.Конявский

УПРАВЛЕНИЕ ЗАЩИТОЙ ИНФОРМАЦИИ НА БАЗЕ СЗИ НСД «АККОРД»

Москва
«Радио и связь»
1999

УДК 519.7+681.322

Рецензенты
д.т.н. профессор П.И. Братухин,
к.т.н. А.П. Курило,
д.т.н. профессор А.Ю. Щербаков

В.А. Конявский
Управление защитой информации на базе СЗИ НСД «Аккорд».
— М.: Радио и связь, 1999. — 325 с., ил. — ISBN 5-256-01494-3

В книге рассматривается применение аппарата теории алгоритмов к задачам, традиционно относимым к области «защиты информации». Описывается применение полученных результатов в разработке СЗИ НСД. Приведены практические рекомендации по управлению защитой с применением СЗИ НСД «Аккорд». Рассматриваются концептуальные и теоретические основы нового направления — защиты электронных документов. Описаны ряд реализаций предложенных методов на объектах информатизации разных уровней.

Для научных работников, специалистов в области защиты информации, студентов и аспирантов соответствующих специальностей.

ISBN 5-256-01494-3

Предисловие

В свое время усилия разработчиков средств защиты информации были сосредоточены на создании программного обеспечения, реализующего те или иные функции защиты. Опыт этих разработок наглядно продемонстрировал ограниченность такого подхода и привел к осознанию необходимости аппаратной защиты. Именно по этому пути все последнее время продвигается коллектив ОКБ САПР, разрабатывающий семейство СЗИ НСД «Аккорд».

К настоящему времени методология аппаратной защиты стала стандартом де-факто, и СЗИ «Аккорд» применяется на большинстве крупных объектов информатизации, собственники которых уделяют вопросам защиты достойное внимание. Накопленный опыт эксплуатации СЗИ «Аккорд» позволил сформулировать ряд вопросов, на которые и призвана дать ответ эта книга.

Первая глава посвящена теоретическим вопросам защиты информации. Полученные в первой главе результаты развиваются во второй главе на основе анализа практики защиты. Здесь же рассматриваются принципы технологической защиты электронных документов. В третьей главе приведены практические рекомендации по управлению защитой информации на базе СЗИ «Аккорд». В четвертой главе описаны некоторые системы, в которых впервые применены принципы технологической защиты электронных документов.

Предлагаемая книга аккумулирует опыт ОКБ САПР, основанный, в свою очередь, на опыте наших коллег и партнеров. Конечно, эта книга не могла бы состояться без многолетнего заинтересованного взаимодействия со специалистами Банка России и Главных Управлений ЦБ РФ, Сбербанка России, Государственного Таможенного Комитета, Пенсионного Фонда России, Федеральной Пограничной Службы, Министерства Обороны РФ, Гостелекома России, Госкомзема России и многих других. Качество защитных функций, обеспечиваемых СЗИ «Аккорд», не могло быть достигнуто без внимания, анализа и рекомендаций со стороны ФАПСИ и Гостехкомиссии России.

Автор выражает свою искреннюю признательность всем нашим партнерам и коллегам, и, в первую очередь, сотрудникам фирмы «Инфокрипт», внесшим значительный вклад в разработку СЗИ «Аккорд». Ощутимое влияние на СЗИ «Аккорд» оказало и наше взаимодействие с фирмами «МОПНИЭИ» и «Техинформконсалтинг», обеспечивающих криптографическую защиту информации, а также с целым рядом предприятий и организаций, специализирующихся в этой непростой области деятельности.

Большой вклад в подготовку книги к изданию внесли сотрудники дизайн-бюро «РФК-Имидж Лаб».

При всем при этом написание книги было бы невозможно, если бы не было 45000 СЗИ «Аккорд», каждый из которых сделан руками сотрудников ОКБ САПР. Для них — особая благодарность и надежда на дальнейшее плодотворное сотрудничество.

Автор будет благодарен, если замечания, пожелания и рекомендации будут направляться по **1@accord.ru.**

ВВЕДЕНИЕ

Информационное взаимодействие является основой деятельности людей. История развития цивилизации неразрывно связана с развитием информационных технологий [4, 19]. Так, в до-письменный период особую роль играли люди преклонного возраста, являясь носителями опыта и знаний, которые могли передаваться из поколения в поколение лишь в устной форме. Рывок в развитии общества связан с появлением письменности, а в дальнейшем — с книгопечатанием. Задумываться о необходимости защиты информации общество стало уже тогда, причем различая, что некоторые сведения нужно сохранять в тайне, а другие популяризировать. Вот как пишет об этом в VII веке неизвестный автор [6]: «Тайну цареву прилично хранить, а о делах божиих объявлять похвально». Эта же максима впоследствии не раз используется в литературе, с том числе и в «Житие Сергия Радонежского»[5]: «Тайну царскую следует хранить, а дела божьи проповедовать похвально; ибо не хранить царской тайны - пагубно и опасно, а молчать о делах божьих славных — беду душе приносить». Тогда же появляется и осознание, что не все носители информации являются одинаково надежными. В [14] сказано: «Лучше в дырявой ладье плыть, нежели злой жене тайны поведать».

Очередные революционные изменения в информационном взаимодействии осуществляются сейчас, с появлением таких носителей, как Интернет.

Значимость сферы информационных взаимодействий непрерывно возрастает. Соответственно растут и попытки приобрести информацию незаконным путем. Так, в [15] отмечается: «Все большее распространение получают преступления, совершаемые путем несанкционированного доступа в компьютерные и телекоммуникационные банковские системы. Об опасности этого сравнительно нового явления свидетельствует тот факт,

что за 1993-1997 годы было совершено несколько сотен попыток несанкционированного проникновения в компьютерные сети Банка России, Сбербанка РФ и наиболее крупных коммерческих банков. Характерным является пример, когда сотрудник Автобанка изменил информацию в базе данных платежной системы и похитил из банка $324тыс.».

Известна попытка хищения 68 млрд. рублей путем внедрения подложных электронных документов в платежную систему ЦБ РФ [12]. Согласно [11]: «Зачисление средств произошло из-за умышленного добавления к массиву входных данных программного комплекса «Операционный день РКЦ» дополнительных записей «электронных» банковских документов. Эти документы были обработаны на ЕС ЭВМ Межрегионального Центра информатизации при ЦБ РФ (МЦИ ЦБ РФ) и выполнены проводки по начислению средств. Фальшивые записи были введены под номером участника, обслуживаемого ГРКЦ ЦБ РФ по г. Москве, по выписке, которая формируется после передачи «электронных» документов из ГРКЦ в МЦИ ЦБ РФ. В целях сокрытия проведенной операции «электронные» банковские документы преступниками были уничтожены. Это незаконная операция стала возможной из-за недостатков в системе комплексной защиты информации».

Известен [11] и пример хищения денежных средств с применением системы платежей «Клиент-Банк». Преступник, воспользовавшись компьютером и дискетой с электронными подписями распорядителей кредитов, составил фиктивное платежное поручение и незаконно списал 3,2 млн. долларов США с валютного счета фирмы, переводя их на заранее открытый по фиктивным документам валютный счет. В дальнейшем похищенные средства были переведены по подложным документам в один из коммерческих банков США. Похищенные денежные средства были возвращены в Россию только благодаря работе сотрудников УЭП ГУВД города Москвы во взаимодействии с правоохранительными органами США

Если в 1996 году было выявлено 15 компьютерных преступлений, то в 1997 — уже 101 [13], причем размер понесенного ущерба достиг 20 миллиардов рублей.

Атакам подвергаются, конечно, не только платежные системы России. Так, 17 мая 1996 года злоумышленник, используя портативный персональный компьютер с модемом, подключился к электронным почтовым ящикам государственного предприятия «Белорусский межбанковский расчетный центр» во время сеанса связи и передачи плановых платежей [1]. Он внес изменения в номер и код счета. Переданная ложная информация была обработана межбанковским расчетным центром.

В августе 1994 г. бывший работник банка, убедившись, что снятая им копия криптографической дискеты с секретным ключом филиала того же банка после его увольнения не изменена, ввел в компьютер фиктивную информацию о зачислении на счет МП «Анжелика» 6 млн. 795 тыс. тенге и перечислении этих средств из Алатауского филиала КРАМДС-Банка в АКБ Казкоммерцбанк на счет ТОО «Хасар»[9].

Когда начались бомбежки Югославии, был взломан правительственный сайт США. Вместо американского флага над Белым домом стал развиваться пиратский «Веселый Роджер» [2]. Таким образом, последствия несанкционированного доступа могут носить весьма серьезный характер не только в экономической, но и в политический сфере. В [10] отмечается, что: «В связи с политическим аспектом проблемы уместно вспомнить о созданной в России автоматизированной системе управления «Выборы». Если не уделить в ней серьезного внимания защите обрабатываемой информации от несанкционированного доступа, результаты голосования могут быть сфальсифицированными и определяться не теми, кто голосует, а теми, кто эти голоса считает».

В связи с актуальностью проблемы защиты информации совершенно необходимо определить объект защиты. В [16] отмечается: «Так как с помощью материальных средств можно защищать только материальный объект, то объектами защиты являются материальные носители информации».

Такой подход представляется конструктивным, особенно с учетом современных представлений об информации и объектах информатизации.

Известен [18] вероятностный подход к определению количества информации, при котором с помощью энтропии выражается количество информации в случайном объекте:

$$H = -\sum_i P_i \log P_i ,$$

где P_i - вероятности получения сигналов.

Фактически P_i отражают сведения о получателе сообщения, и для различных получателей P_i могут быть различны.

Алгоритмический подход [7] основан на теории алгоритмов. Описанием слова S в алфавите A относительно способа описания f называется слово α в алфавите $\{0,1\}$ такое, что $f(\alpha)=S$, а сложность данного слова — длины кратчайшего описания. Ясно, что f отображает здесь характеристики получателя сообщения.

Н.Винер [3] определил информацию, как «обозначение содержания, полученного из внешнего мира в процессе нашего приспособления к нему и приспособления к нему наших чувств». Обращает на себя внимание та же связь информации с получателем сообщения. Н. Винер показал, что управляющая информация определяется целью управления. На приемник информация действует не как физическая причина, так как энергетические характеристики сигнала играют второстепенную роль.

Рассматривая эти и другие подходы как предпосылки создания нового направления науки — «информатики взаимодействия», в [8] дается следующее определение: «Информация, заключенная в сообщениях, есть

сущность, определяющая изменение знаний при получении сообщения», или, применительно к формальным описаниям: «Информация есть сущность, сохраняющая при вычислимом изоморфизме».

Известны, однако, и другие определения информации. Так, в [20] в рамках науки «Информациология», дается следующее определение: «Информация — это фундаментальный генерализационно-единый безначально-бесконечный законопроцесс резонансно-сотового, частотно-квантового и волнового отношения, взаимодействия, взаимопревращения и взаимосохранения (в пространстве и времени) энергии, движения, массы и антимассы на основе материализации и дематериализации в микро- и макроструктурах Вселенной».

Несколько по-другому определяется информация в [17], а именно, как: «сведения о лицах, предметах, фактах, событиях, явлениях и процессах независимо от формы их представления».

В словаре Ф.А. Брокгауза-И.А. Ефрона говорится: «Информация — прошение малороссийских гетманов московскому царю или польскому королю» (цитата по [19], т.к. в доступном издании словаря данное определение не найдено).

Такое разнообразие подходов есть отражение сложности проблемы. Мы же будем исходить из того, что гарантированная защита возможна лишь в том случае, когда она реализуется, начиная с момента порождения объекта защиты. Естественно в этом случае рассматривать защиту информации как защиту ОИ и их составляющих, являющихся носителями сведений (вычислительные машины, комплексы, сети, электронные документы и т.д.).

Функционирование ОИ связано с решением ряда важнейших задач, таких, как обеспечение целостности, конфиденциальности, доступности и других, чем традиционно и занимаются специалисты в области защиты информации. Развитие этой сферы целесообразно, но расширение непродуктивно, так как информация может порождаться только человеком, и в этом смысле техническая защита информации у ее источника невозможна. Напротив, порождение и обработка электронных документов (ЭлД) связаны с функциональностью ОИ. Именно здесь возникают задачи защиты ЭлД, выработки соответствующих методов и механизмов.

Проиллюстрируем тезис о развитии без расширения на примере антивирусной защиты. Средства поиска компьютерных вирусов активно разрабатываются и продаются. Тем не менее, вполне разумной является и постановка следующего вопроса: «Можно ли (в общем случае) детектировать вирус?». Из общих соображений можно заметить, что детектирование вирусов очень близко к проблеме самоприменимости, которая, как известно, в общем случае неразрешима. Тогда успехи «науки вирусологии» — это лишь успехи в поисках частных случаев. Но есть ли примеры, свидетельствующие об обратном? Известен ли вариант разрушающего программного воздействия, которое нельзя обнаружить за реальное время путем анализа содержимого внешней памяти, как это делается сейчас? Другими словами — является ли «вирусология» наукой?

Вот один из уже широко известных примеров.

Пусть K — код РПВ;

S_1 — случайное число;

$\oplus$ — оператор сложения по модулю 2.

Вычислим

$S_2 = S_1 \oplus K$

Очевидно, что S_2 — случайное число.

Пусть теперь в состав компьютерной системы входят S_1 и S_2 (случайные числа) и операция $\oplus$, которая не опасна. Анализ S_1 и S_2 на наличие РВП дает отрицательный результат в силу их случайности. Однако, если при некоторых условиях будет выполнено сложение этих чисел, то результатом этого будет РПВ. Действительно,

$$S_1 \oplus S_2 = S_1 \oplus S_1 \oplus K = 0 \oplus K = K$$

Очевидно, что вирус, построенный на основе данного принципа, не может быть обнаружен за счет анализа данных. Отсюда и вывод — «вирусология» — скорее искусство, чем наука, и является хоть и важным, но вспомогательным механизмом, так как ни один набор антивирусных программ не может гарантировать детектирование всех РПВ.

Другим примером непродуктивного расширения может служить попытка автоматического анализа программных средств на отсутствие недекларируемых возможностей. Речь здесь идет о том, что найти некоторые конкретные недекларированные механизмы может быть и возможно, но вот дать гарантию их отсутствия, конечно, нельзя. И более того — что является гарантией отсутствия недекларированных возможностей у собственно средств анализа на отсутствие недекларируемых возможностей?

На наш взгляд, эти проблемы могут быть проиллюстрированы известным теологическим парадоксом: «Если Бог всемогущ, то может ли он создать камень, который не сможет поднять?»

Именно в этом смысле расширенные подходы кажутся нам непродуктивными. Более осмысленными представляются подходы, связанные с созданием условий, в которых РПВ, недекларируемые возможности и другие опасности просто не могут реализоваться — так называемой изолированной программной средой (ИПС).

Забавная иллюстрация факта неразрешимости некоторых алгоритмических задач попалась автору при подготовке этой книги. На стр. 34 курсивом выделен абзац, при обработке которого в режиме «проверки грамматики» Winword непременно зависает. Чтобы в этом убедиться, достаточно ввести в Winword текст этого абзаца. Вслед за этим (пока компьютер будет перегружаться) появится время поразмышлять: 1) можно ли корректно защищать некорректно написанную программу?, и 2) почему данная проблема относится к проблеме неразрешимости?

Отметим, что результат исполнения программ далеко не всегда может быть описан соотношением $y = f(x)$, где f функция, x — данные. Действительно, всегда ли проверенная программа, использующая проверенные данные, дает правильный результат? Очевидно, что нет. Проверенной должна быть и среда, в которой исполняется программа. И далее — проверенной должна быть и аппаратура, в том числе и с микропрограммным управлением. Учитывая, что инициализация любой вычислительной системы осуществляется не одномоментно, а в некоторой последовательности, можно рассмотреть возможность организации доверенной загрузки, и создания ИПС на этой основе.

Выше уже отмечалось, что ОИ порождает не информацию, а электронные документы. Ежегодно в России в обращении находится 800-850 млрд. документов [19], что делает задачу создания защищенного электронного документа крайне актуальной. Во все времена подделка документов квалифицировалась как преступление.

Для того, чтобы следы компьютерных преступлений можно было анализировать, нужно, по-крайней мере, создать условия, при которых они (следы) остаются.

Еще лучше, если остаются не просто следы, а следы отчетливые. Так, следы на взрыхленной земле обычно отчетливей, чем следы на асфальте. Именно поэтому вдоль границы государства создается КСП — контрольно-следовая полоса. Но и этого недостаточно — нужны еще и пограничники, которые регулярно осматривают КСП, а иногда и взрыхляют ее.

Представляется, что такая метафора достаточно хорошо описывает действия, которые необходимо предпринимать для того, чтобы расследование компьютерных преступлений было успешным. А именно: ОИ должны быть защищены, а анализ электронных документов должен позволять сделать вывод о том — подлинник это, копия или фальшивка. Хорошо было бы фиксировать и анализировать попытки несанкционированного доступа к ОИ и электронным документам, носителям которых является ОИ.

Ниже рассматривается, как защитить эти две сущности — ОИ и электронные документы. Теоретическая часть работы основывается на развитии субъектно-объектной модели А.Ю. Щербакова.

Литература

1. Ахраменка Н.Ф., Егоров Ю.А., Козлов В.Е., Леонов А.П. «Преступление и наказание» в платежной системе электронных документов. Управление защитой информации». Т.2. №2.

2. Баршев В. «Добро пожаловать, взломщик!» Российская газета, 17 сентября 1999.

3. Винер Н. «Кибернетика и общество». Москва, Советское радио, 1958.

4. Глушков В.М. «Основы безбумажной информатики». Москва, Наука, 1982.
5. Епифаний Премудрый. «Житие Сергия Радонежского». Памятники литературы Древней Руси. XIV - середина XV века. Москва, Художественная литература, 1981.
6. «Житие Марии Египетской, бывшей блудницы, честно подвязавшейся в Иорданской пустыне». Византийский легенды. Москва, НИЦ «Ладомир», 1994.
7. Колмогоров А.А. «Три подхода к определению понятия «количество информации». Проблемы передачи информации, Т.1. Вып.1, 1965.
8. Кузнецов Н.А., Мусхелишвили Н.Л., Шрейдер Ю.А. «Информационное взаимодействие как объект научного исследования». Вопросы философии, №1, 1999.
9. Леонов А.П., Леонов К.А., Фролов Г.В. «Безопасность автоматизированных банковских и офисных систем». Минск, Национальная книжная палата Белоруссии, 1996.
10. Мельников В.В. «Защита информации в компьютерных системах». Москва, Финансы и статистика, 1997.
11. Молокостов В., Овчинский А., Наумов И. «Современное состояние и тенденции развития компьютерной преступности в банковской сфере». Аналитический банковский журнал, №7(38), 1998.
12. «Отчет МВД РФ перед гражданами России». Российская газета, 11 марта 1994.
13. Сидоров В. «Проблемы защиты банков от криминальных угроз». Аналитический банковский журнал, №7(38), 1998.
14. «Слово Данила Заточеника, еже написа своему князю Ярославу Володимеровичу». Памятники литературы Древней Руси. XII век. Москва, Художественная литература.,1980.
15. Степашин С.В. «Проблемы безопасности деятельности банков». Аналитический банковский журнал, №7(38), 1998.
16. Торокин А.А. «Основы инженерно-технической защиты информации». Москва, Ось-89, 1998.
17. Федеральный Закон «Об информации, информатизации и защите информации». Принят Государственной Думой 25 января 1995.
18. Шеннон К. «Математическая теория связи. Работы по теории информации и кибернетики. М.: Издательство иностранной литературы». 1963.
19. Шурухнов Н.Г. «Расследование неправомерного доступа к компьютерной информации». Москва, Щит-М, 1999.
20. Юзвишин И.И. «Информациология или закономерности информационных процессов и технологий в микро- и макромирах Вселенной». Москва, Радио и связь, 1996.

Глава 1.
ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ЗАЩИТЫ ИНФОРМАЦИИ

1. ВВЕДЕНИЕ. МОДЕЛИ БЕЗОПАСНОСТИ

Рассмотрим некоторые известные модели безопасности.

1. *Модель дискреционного доступа* [12]. В рамках модели контролируется доступ субъектов к объектам. Для каждой пары субъект-объект устанавливаются операции доступа (READ, WRITE и другие).

Контроль доступа осуществляется посредством механизма, который предусматривает возможность санкционированного изменения правил разграничения доступа. Право изменять правила предоставляется выделенным субъектам.

2. *Модель дискретного доступа* [6, 19, 27]. В рамках модели рассматриваются механизмы распространения доступа субъектов к объектам.

3. *Модель мандатного управления доступом Белла-Лападула* [6, 17, 23].

Формально записана в терминах теории отношений. Описывает механизм доступа к ресурсам системы, при этом для управления доступом используется матрица контроля доступа. В рамках модели рассматриваются простейшие операции READ и WRITE доступа субъектов к объектам, на которые накладываются ограничения.

Множества субъектов и объектов упорядочены в соответствии с их уровнем полномочий и уровнем безопасности, соответственно.

Состояние системы изменяется согласно правилам трансформации состояний.

В множестве субъектов могут присутствовать доверенные субъекты, которые не подчиняются ограничениям на операции READ и WRITE. В таком случае модель носит название модели доверенных субъектов [6].

4. *Модели распределенных систем (синхронные и асинхронные)* [6]. В рамках моделей субъекты выполняются на нескольких устройствах обработки. Рассматриваются операции доступа субъектов к объектам READ и WRITE, которые могут быть удаленными, что может вызвать противоречия в модели Белла-Лападула.

В рамках асинхронной модели в один момент времени несколько субъектов могут получить доступ к нескольким объектам.

Переход системы из одного состояния в другое состояние в один момент времени может осуществляться под воздействием более, чем одного субъекта.

5. *Модель безопасности военной системы передачи данных (MMS - модель)* [6, 17, 28]. Формально записана в терминах теории множеств. Субъекты могут выполнять специализированные операции над объектами сложной структуры. В модели присутствует администратор безопасности для управления доступом к данным и устройствам глобальной сети передачи данных. При этом для управления доступом используются матрицы контроля доступа. В рамках модели используются операции READ, WRITE, CREATE, DELETE доступа субъектов к объектам, операции над объектами специфической структуры, а также могут появляться операции, направленные на специфическую обработку информации.

Состояние системы изменяется с помощью функции трансформации.

6. *Модель трансформации прав доступа* [17, 33]. Формально записана в терминах теории множеств. В рамках модели субъекту в данный момент времени предоставляется только одно право доступа. Для управления доступом применяются функции трансформации прав доступа.

Механизм изменения состояния системы основывается на применении функций трансформации состояний.

7. *Схематическая модель* [17, 32]. Формально записана в терминах теории множеств и теории предикатов. Для управлением доступа используется матрица доступа со строгой типизацией ресурсов. Для изменения прав доступа применяется аппарат копирования меток доступа.

8. *Иерархическая модель* [17, 24]. Формально записана в терминах теории предикатов.

Описывает управление доступом для параллельных вычислений, при этом управление доступом основывается на вычислении предикатов.

9. *Модель безопасных спецификаций* [17, 26]. Формально описана в аксиоматике Хоара.

Определяет количество информации, необходимое для раскрытия системы защиты в целом. Управление доступом осуществляется на основе классификации пользователей.

Понятие механизма изменения состояния не применяется.

10. *Модель информационных потоков* [17, 34]. Формально записана в терминах теории множеств. В модели присутствуют объекты и атрибуты, что позволяет определить информационные потоки. Управление доступом осуществляется на основе атрибутов объекта.

Изменением состояния является изменение соотношения между объектами и атрибутами.

12. *Вероятностные модели* [6]. В модели присутствуют субъекты, объекты и их вероятностные характеристики. В рамках модели рассматриваются операции доступа субъектов к объектам READ и WRITE. Операции доступа также имеют вероятностные характеристики.

13. *Модель элементарной защиты* [9]. Предмет защиты помещен в замкнутую и однородную защищенную оболочку, называемую преградой. Информация со временем начинает устаревать, т.е. цена ее уменьшается. За условие достаточности защиты принимается превышение затрат времени на преодоление преграды нарушителем над временем жизни информации. Вводятся вероятность непреодоления преграды нарушителем $P_{СЗИ}$, вероятность обхода преграды нарушителем $P_{обх}$, и вероятность преодоления преграды нарушителем за время, меньшее времени жизни информации $P_{нр}$. Для введенной модели нарушителя показано,

что $P_{СЗИ} = min[(1 - P_{нр})(1-P_{обх})]$,

что является иллюстрацией принципа слабейшего звена. Развитие модели учитывает вероятность отказа системы и вероятность обнаружения и блокировки действий нарушителя.

14. *Модель системы безопасности с полным перекрытием* [19, 25]. Отмечается, что система безопасности должна иметь по крайней мере одно средство для обеспечения безопасности на каждом возможном пути проникновения в систему. Модель описана в терминах теории графов. Степень обеспечения безопасности системы можно измерить, используя лингвистические переменные [35, 1]. В базовой системе рассматривается набор защищаемых объектов, набор угроз, набор средств безопасности, набор уязвимых мест, набор барьеров.

15. *Модель гарантированно защищенной системы обработки информации* [5]. В рамках модели функционирование системы описывается последовательностью доступов субъектов к объектам. Множество субъектов является подмножеством множества объектов. Из множества объектов выделено множество общих ресурсов системы, доступы к которым не могут привести к утечке информации. Все остальные объекты системы являются порожденными пользователями, каждый пользователь принадлежит множеству порожденных им объектов. При условиях, что в системе существует механизм, который для каждого объекта устанавливает породившего его пользователя; что субъекты имеют доступ только к общим ресур-

сам системы и к объектам, порожденным ими, и при отсутствии обходных путей политики безопасности модель гарантирует невозможность утечки информации и выполнение политики безопасности.

16. *Субъектно-объектная модель* [11, 21]. В рамках модели все вопросы безопасности описываются доступами субъектов к объектам. Выделены множество объектов и множество субъектов. Субъекты порождаются только активными компонентами (субъектами) из объектов. С каждым субъектом связан (ассоциирован) некоторый объект (объекты), т.е. состояние объекта влияет на состояние субъекта. В модели присутствует специализированный субъект-монитор безопасности субъектов (МБС), который контролирует порождение субъектов. Показана необходимость создания и поддержки изолированной программной среды.

Из упомянутых моделей для нас наибольший интерес представляет дискреционные и мандатные механизмы разграничения доступа (как наиболее распространенные), модель гарантированно защищенной системы (в силу гарантированности) и субъектно-объектная модель (рассматривающая не только доступы, но и среду, в которой они совершаются).

Согласно [6] под сущностью понимается любая составляющая компьютерной системы.

Субъект определяется как активная сущность, которая может инициировать запросы ресурсов и использовать их для выполнения каких-либо вычислительных заданий.

Объект определяется как пассивная сущность, используемая для хранения и получения информации.

Доступ — взаимодействие между объектом и субъектом, в результате которого происходит перенос информации между ними. Взаимодействие происходит при исполнении субъектами операций. Существуют две фундаментальные операции: операция чтения (перенос информации от объекта к субъекту) и операция записи (перенос информации от субъекта к объекту).

Данные операции являются минимально необходимым базисом для описания моделей, описывающих защищенные системы.

Уровень безопасности определяется как иерархический атрибут. Каждая составляющая компонента системы ассоциирована с уровнем безопасности.

Для представления уровней безопасности введено:

L — множество уровней безопасности;

символы “<“, “≤”, “>”, “≥”, описывают иерархические отношения между элементами множества L.

2. МОДЕЛИ РАЗГРАНИЧЕННОГО ДОСТУПА

Модели разграничения доступа, построенные по принципу предоставления прав, делятся на два основных типа: модели дискреционного и мандатного доступа.

Известны [10, 15, 2] отечественные реализации дискреционного механизма, отличающиеся, в том числе, составом набора общих прав.Так, в СЗИ НСД «Dallas Lock» [10] неявно используются атрибуты

$$R_D=\{Y,N\},$$

где

Y — право полного доступа субъекта к объекту (на чтение, запись, модификацию и т.д.);

N — отсутствие такого права.

В соответствии с этим каждому субъекту-пользователю ставится в соответствие либо список разрешенных объектов, либо список запрещенных объектов.

В СЗИ НСД «Secret Net» [15] набор применяемых атрибутов шире, а именно $R_S=\{R,W,X\}$,

где

R — разрешение на чтение;

W— разрешение на модификацию;

X — разрешение на запуск задачи.

Наиболее полный набор общих прав используется в СЗИ НСД «Аккорд»[2], и включает $R_A=\{R,W,C,D,N,V,O,M,E,G,X\}$,

где

R — разрешение на открытие файлов только для чтения;

W— разрешение на открытие файлов только для записи;

C — разрешение на создание файлов на диске;

D — разрешение на удаление файлов;

N — разрешение на переименование файлов;

V — видимость файлов;

O — эмуляция разрешения на запись информации в файл;

M— разрешение на создание каталогов на диске;

E — разрешение на удаление каталогов на диске;

G — разрешение перехода в этот каталог;

X — разрешение на запуск программ.

Отметим, что набор атрибутов R_S и R_A близки к атрибутам, применяемым в ОС UNIX и NetWare соответственно.

Опуская особенности реализации, рассмотрим некоторые возможности применения различных наборов атрибутов для описания политик информационной безопасности.

Рассмотрим набор атрибутов $R_D=\{Y,N\}$.

Предположим, что в системе действуют два пользователя U_1 и U_2. Каждый пользователь имеет право полного доступа Y к некоторому множеству объектов, иными словами для пользователя U_1 определен список разрешенных объектов $\{O_{11},\ldots,O_{1n_1}\}=\{O_{1i}\}_{i=1}^{n_1}$, обозначим его $\mathbf{O}_1$.

Для пользователя U_2 соответственно определен список разрешенных объектов $\{O_{21},\ldots,O_{2n_2}\}=\{O_{2i}\}_{i=1}^{n_2}$, обозначим его $\mathbf{O}_2$.

В случае, когда $\mathbf{O}_1\cap\mathbf{O}_2\neq\varnothing$, т.е. когда существует хотя бы один объект, который содержится в списке разрешенных объектов как для пользователя U_1 так и для пользователя U_2, возможна утечка информации. Покажем это.

Пусть существует: $O\in\mathbf{O}_1$ и $O\in\mathbf{O}_2$, т.е. $\mathbf{O}_1\cap\mathbf{O}_2=O$.

Представим список разрешенных объектов пользователя U_1 следующим образом: $\mathbf{O}_1=O\cup\mathbf{O}_1'$.

Предположим, что пользователя U_2 интересуют данные, хранящиеся в объекте $O\in\mathbf{O}_1'$. При этом пользователь U_1 помещает в объект O данные из объекта O, а пользователь U_2 в свою очередь может прочитать эти данные, так как $O\in\mathbf{O}_2$. Таким образом пользователь U_2 получает доступ к информации, хранящейся в объекте $O\in\mathbf{O}_1'$, к которому U_2 не имеет права доступа, то есть происходит утечка информации. Таким образом, для описанного состава атрибутов утечка информации возможна в любом случае, когда $\mathbf{O}_1\cap\mathbf{O}_2\neq\varnothing$, т.е. существует только одна выполнимая политика безопасности, а именно: списки разрешенных пользователям объектов не имеют общих элементов. Ясно, что это очень сильное ограничение, тем более, что и его недостаточно. Действительно, если право полного доступа подразумевает возможность переименования объекта, то возможны и другие каналы утечки, аналогичные описанным ниже.

Рассмотрим теперь набор атрибутов $R_S=\{R, W, X\}$. Этот набор шире R_D. Действительно, атрибуты из R_D могут быть описаны атрибутами из R_S, а именно:

$$Y = \{R, W, X\};$$
$$N = \{-, -, -\}.$$

Обратное невозможно.

Предположим, что в системе действуют два пользователя U_1 и U_2.

Пользователь U_1 имеет права доступа $\{R\}$ к объекту O_1.

Пользователь U_2 имеет право доступа $\{R\}$ к объекту O_2.

Предположим, что пользователя U_2 интересуют данные, хранящиеся в объекте O_1. При этом пользователь U_1, пользуясь имеющимся у него правом W на модификацию объекта O_1, может переименовать его в O_2. Пользователь U_2 в свою очередь, пользуясь имеющимися у него правами $\{R\}$ на объект O_2, получает доступ к данным, которые содержались в объекте O_1, т.е. происходит утечка информации. При этом ни пользователь U_1, ни пользователь U_2 не нарушают политики безопасности. Таким образом, для данного состава атрибутов утечка информации возможна в том случае, если хотя бы один из пользователей имеет право $\{W\}$ к защищаемому объекту.

Это также очень сильное ограничение, и в этой связи возможности разграничения доступа, предоставляемые данным набором атрибутов, обычно усиливаются дополнительными механизмами.

Атрибуты из R_S в свою очередь могут быть описаны атрибутами из R_A, а именно: $\{R\}_{R_S}=\{R\}_{R_A}$;

$$\{W\}_{R_S}=\{R, W, C, D, N\}_{R_A};$$
$$\{X\}_{R_S}=\{X\}_{R_A}.$$

Обратное невозможно.

Для каждой из реализаций моделей дискреционного доступа фундаментальным является вопрос, является ли безопасной та или иная начальная конфигурация, т.е. возможна или нет утечка некоторого права. Другими словами, может ли пользователь при некоторых условиях получить доступ к объекту, если в начальном состоянии возможность такого доступа не предусмотрена (не установлено соответствующее право). В общем случае ответ на этот вопрос дается в модели дискреционного доступа с позиций распространения прав доступа.

Рассмотрим дискретную модель разграничения доступа, приведенную в [6], как модель распространения прав доступа.

2.1. Модель дискретного доступа

Система защиты представляется в виде некоторого декартова произведения множеств, составными частями которых являются составные части системы защиты: субъекты, объекты, уровни доступа, операции и т.д.

В качестве математического аппарата выбран аппарат теории множеств.

Рассмотрим типичную модель системы защиты. Она состоит из следующих частей:

1. $R=\{r_1,\dots,r_n\}$ — конечный набор общих прав;

2. S_0 — конечный набор исходных субъектов,

O_0 — конечный набор исходных объектов,

где $S_0 \subseteq O_0$;

3. C — конечный набор команд формы $\alpha(X_1,\dots X_n)$,

где α — имя,

$X_1,\dots X_n$ — формальные параметры, указывающие на объекты;

4. J — интерпретация для команд, такая, что J отображает C в последовательность элементарных операций;

Элементарными операциями являются:

— ввести право r в (s,o);

— удалить право r из (s,o);

— создать субъект s;

— создать объект o;

— разрушить субъект s;

— разрушить объект o;

где r — общее право; s — имя субъекта; o — имя объекта.

5. Условия для команд. Условие C — отображение элементов набора команд в конечный набор прав.

Право — это тройка (r,s,o), где $r \in R$, а s и o — формальные параметры.

6. P — матрица доступов со строкой для каждого субъекта из S и столбцом для каждого объекта из O.

Определение 1. Для заданной системы защиты команда $\alpha(X_1,\dots X_n)$ может привести к утечке общего права r, если ее интерпретация содержит некоторую операцию вида «ввести» r в (s,o) для некоторых $o \in O$ и $s \in S$».

Определение 2. Для заданной системы защиты и права r начальная конфигурация $(s_0\ o_0\ p_0)$ является безопасной для r в этой системе, если не существует конфигурации $(s\ o\ p)$, такой что $(s_0\ o_0\ p_0)$ ведет к $(s\ o\ p)$, и

существует команда $\alpha(X_1,\dots X_n)$, условия которой удовлетворяются в $(s\ o\ p)$ и которая для некоторых реальных параметров дает утечку права r через команду «ввести r в (s,o) для некоторых $o \in O$ и $s \in S$», причем s и o существуют во время команды «ввести», для которых r не находится в $p[s,o]$, где $p[s,o]$ — элемент матрицы доступа P.

Для моделей, построенных на основе дискретной защиты доказана следующая

Теорема. Не существует алгоритма, который может решить для произвольной дискретной защиты и общего права r, является или нет заданная исходная конфигурация безопасной.

2.2. Модель мандатного доступа

Классической моделью, лежащей в основе построения многих систем мандатного доступа является модель Белла и Лападула (БЛМ).

Рассмотрим описание БЛМ по [5], на основе [29].

Классы объектов предполагаются неизменными.

Определены конечные множества S, O, R, L.

S — множество субъектов системы;

O — множество объектов, не являющихся субъектами;

R — множество прав доступа,

где $R = \{read\ (r),\ write(w),\ execute\ (e),\ append\ (a)\}$;

L — уровни секретности.

Множество V состояний системы определяется произведением множеств:

$V = B \times M \times F \times H$,

где сомножители определяются следующим образом:

B — множество текущих доступов и есть подмножество множества подмножеств произведения $S \times O \times R$. Множество подмножеств обозначается $P(S \times O \times R)$; элементы множества B обозначаются b и они представляют в текущий момент t графы текущего доступа (в каждый момент субъект может иметь только один вид доступа к данному объекту).

M — матрица разрешенных доступов, $M = |M_{ij}|$, $M_{ij} \subseteq R$.

F — подмножество множества $L^S \times L^S \times L^O$, где каждый элемент $f = (f_S, f_O, f_C)$, $f \in F$, — вектор, который состоит из трех компонент, каждая из которых тоже вектор (или отображение).

f_S — уровень допуска субъектов (это некоторое отображение $f: S \to L$);

f_O — уровень секретности объектов (это некоторое отображение $f: O \to L$);

f_C — текущий уровень секретности субъектов (это тоже некоторое отображение $f_C: S \to L$).

Элементы подмножества F, которые допущены для определения состояния, должны удовлетворять соотношению:

$\forall S \in S \;\; f_S(S) \geqslant f_C(S)$

H — текущий уровень иерархии объектов, в [29] этот уровень не изменяется, совпадает с f_O и далее не рассматривается.

Элементы множества V состояний означаются через v.

Определены множества:

Q — запросов в систему;

D — множество решений по поводу запросов ($D = \{yes, no, error\}$).

Множество W действий системы определено как

$W \subseteq Q \times D \times V \times V = \{(q, d, v_1, v_2)\}$.

Каждое действие системы (q, d, v_2, v_1) имеет следующий смысл: если система находилась в данный момент в состоянии v_1, поступил запрос q, то принято решение d и система перешла в состояние v_2.

Пусть T — множество значений времени ($T = \mathrm{N}$ — множество натуральных чисел).

Определен набор из трех функций (x, y, z)

$x: T \to Q$,

$y: T \to D$,

$z: T \to V$,

и обозначены множества таких функций X, Y, Z соответственно.

Рассмотрим $X \times Y \times Z$ и определим понятие системы в БЛМ.

Определение 1. Системой $\Sigma(Q, D, W, z_0)$ называется подмножество $X \times Y \times Z$ такое, что,

$(x, y, z) \in \Sigma(Q, D, W, z_0) \Leftrightarrow (x_t, y_t, z_{t-1}) \in W$

для каждого значения $t \in T$, где z_0 — начальное состояние системы.

Определение 2. Каждый набор $(x, y, z) \in \Sigma(Q, D, W, z_0)$ называется реализацией системы.

Определение 3. Если (x, y, z) — реализация системы, то каждая четверка (x_t, y_t, z_t, z_{t-1}) называется действием системы.

Определение 4. Тройка $(S, O, X) \in S \times O \times R$ удовлетворяет свойству простой секретности (ss-свойство) относительно f, если $X = execute$, или $X = append$, или, если $X = read$ и $f_S(S) \geqslant f_O(O)$, или $X = write$ и $f_S(S) \geqslant f_O(O)$.

Определение 5. Состояние $v = (b, M, f, h)$ обладает *ss*-свойством, если для каждого элемента $(S, O, X) \in B$ этот элемент обладает *ss*-свойством относительно f.

Определение 6. Состояние $v = (b, M, f, h)$ обладает *-свойством, если для каждого $(S, O, X) \in B$ при $X = write$ текущий уровень субъекта $f_C(S)$ равен уровню объекта $f_O(O)$, или при $X = read$ $f_C(S) \geqslant f_O(O)$, или при $X = append$ $f_O(O) \geqslant f_C(S)$.

Определение 7. Состояние обладает *-свойством относительно множества субъектов S', $S' \in S$, если оно выполняется для всех троек (S, O, X) таких, что $S \in S'$

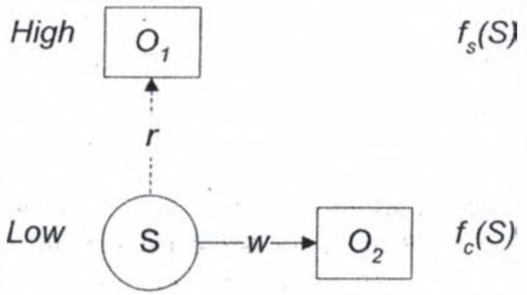
Определение 8. Субъекты из множества $S \setminus S'$ называются доверенными.

Определение 9. Состояние $v = (b, M, f, h)$ обладает *ds*-свойством, если " $\forall (S, O, X) \in b \Rightarrow X \in m_{SO}$, где $M = \|m_{SO}\|$ — матрица доступа состояния v, т.е. $\forall$ доступы из множества текущих доступов для $S \in S$ и $O \in O$ находятся в матрице разрешенных доступов.

Определение 10. Состояние $v=(b, M, f, h)$ называется безопасным, если оно обладает одновременно *ss*-свойством, *- свойством относительно S' и *ds*-свойством.

Из определения *ss*-свойства следует, что в безопасном состоянии возможно чтение вниз, кроме того, *ss*-свойство определяет ограничение на возможность модификации, которое связано с *write*: $f_S(S) \geqslant f_O(O)$.

Объясним *-свойство. Если субъект может понизить свой текущий допуск до $f_C(S)=f_O(O)$, то, согласно *-свойству, он может писать в объект. При этом он не может читать объекты на более высоких уровнях, хотя допуск $f_S(S)$ ему это может позволить. Тем самым исключается возможный канал утечки:



Таким образом, при записи информационный поток опять не может быть направлен вниз. Исключение возможно только для доверенных субъектов, которым разрешено строить информационный поток вниз. При этом доверенность субъекта означает безопасность такого потока вниз (поэтому эти потоки считаются разрешенными).

Для того, чтобы доказать, что любой поток на траектории вычислительной системы разрешен, достаточно показать, что, выходя из безопасного состояния, и следуя допустимым действиям мы опять приходим в безопасное состояние, тем самым любая реализация процесса будет безопасной. Приведем строгое обоснование этого вывода.

Определение 11. Реализация (x, y, z) системы $\Sigma(Q, D, W, z)$ обладает ss-свойством (*-свойством, *ds*-свойством), если в последовательности $(z_0, z_1, \ldots)$ каждое состояние z_n обладает *ss*-свойством (* -свойством , *ds*-свойством).

Определение 12. Система обладает *ss*-свойством (соответственно, *-свойством, *ds*-свойством), если каждая ее реализация обладает *ss*-свойством (соответственно, *-свойством, *ds*-свойством).

Определение 13. Система называется безопасной, если она обладает одновременно *ss*-свойством, *-свойством, и *ds* - свойством.

Теорема А1. $\Sigma(Q, D, W, z_0)$ обладает *ss* -свойством для любого начального z_0, которое обладает *ss*-свойством тогда и только тогда, когда W удовлетворяет следующим условиям для каждого действия $\left(q, d, \left(b^*, M^*, f^*, h^*\right), (b, M, f, h)\right)$:

(1) $\forall (S, O, X) \in b^* \mid b$ обладает *ss*-свойством относительно f^*.

(2) если $(S, O, X) \in b$ и не обладает *ss*-свойством относительно f^*, то $(S, O, X) \notin b^*$.

Теорема А2. Система $\Sigma(Q, D, W, z_0)$ обладает *- свойством относительно S' для любого начального состояния z_0, обладающего *-свойством относительно S' тогда и только тогда, когда W удовлетворяет следующим условиям для каждого действия $\left(q, d, \left(b^*, M^*, f^*, h^*\right), (b, M, f, h)\right)$:

(1) $\forall S \in S', \forall (S, O, X) \in b^* \mid b$ обладает *-свойством относительно f^*;

(2) $\forall S \in S', \forall (S, O, X) \in b$ и (S, O, X) обладает *-свойством относительно f^*, то $(S, O, X) \notin b^*$.

Теорема А3. Система $\Sigma(Q, D, W, z_0)$ обладает *ds*-свойством тогда и только тогда, когда для любого начального состояния, обладающего *ds*-свойством, W удовлетворяет следующим условиям для любого действия $\left(q, d, \left(b^*, M^*, f^*, h^*\right), (b, M, f, h)\right)$:

(1) $(S, O, X) \in b^* \mid b$, то $X \in m^*_{SO}$,

(2) $(S, O, X) \in b^* \mid b$ $X \notin m^*_{SO}$, то $(S, O, X) \notin b^*$

Теорема (Basic Security Theorem). Система $\Sigma(Q,D,W,z_0)$ — безопасная тогда и только тогда, когда z_0 — безопасное состояние и W удовлетворяет условиям теорем A1, A2, A3 для каждого действия.

Таким образом, построенная модель является безопасной, но при этом на рассматриваемую систему накладывается ряд достаточно жестких ограничений.

Как отмечено выше, *-свойство позволяет субъекту понижать его текущий допуск до уровня секретности объекта и, при этом не дает возможности читать из объекта с более высоким уровнем секретности.

Предположим, что субъект S получил доступ *read* к объекту O_1, считал интересующую его информацию, затем понизил свой текущий уровень допуска и, получив доступ *write* к объекту O_2, произвел в него запись данных, прочитанных из O_1.

Приведенный пример демонстрирует возможный канал утечки, при том, что не происходит нарушения политики безопасности, т.е. все действия субъекта являются правомерными.

Исключить ситуацию, подобную данной, возможно только лишь при условии, что система не обладает памятью. Данное ограничение, очевидно, является неприменимым к наиболее часто используемым системам.

Рассмотренное конечное множество прав доступа $R=\{read, write, execute, append\}$ обеспечивает работу системы, а именно, доступ субъектов к интересующим их объектам, только на прикладном уровне.

2.3. Модель гарантированно защищенной системы обработки информации

Модель Σ системы, которая оперирует с ценной информацией определяется [5] следующим образом.

Время дискретно и принимает значения из множества $N=\{1,2,...\}$. Информация в системе Σ, включая описание самой системы, представима в форме слов некоторого гипотетического языка $Я$ над некоторым конечным алфавитом A.

Объект в Σ — это конечное множество слов из $Я$, состояние объекта — выделенное слово из множества, определяющего этот объект. С понятием объекта связано агрегирование информации в Σ и о Σ.

Приведены следующие примеры объектов:

— объектом является принтер, который можно рассматривать как автомат с конечным множеством состояний, а эти состояния — суть слова языка $Я$.

— объект - файл; множество слов, которые могут быть записаны в файле, является конечным и определяет объект, а состояния объекта — это текущая запись в файле, которая тоже является словом в языке $Я$.

Вся информация о Σ в данный момент может быть представлена в виде состояний конечного множества объектов. Считается, что состояние системы Σ — это набор состояний ее объектов.

Объекты могут создаваться и уничтожаться, поэтому можно говорить о множестве объектов системы Σ в момент t, которое обозначается O_t, $|O_t|<\infty$.

Для каждого $t\in N$ в O_t выделено подмножество S_t субъектов: $S_t\in O_t$.

Любой субъект $S\in S_t$ есть описание некоторого преобразования информации в системе Σ.

Каждый субъект может находиться в двух состояниях: в форме описания, в котором субъект называется неактивизированным, и в форме, в которой субъект называется активизированным (процесс).

Активизировать субъект может только другой активизированный субъект. Для каждого $t\in N$ на множестве S_t определяется орграф Γ_t, где S_1 и S_2 из S_t соединены дугой $S_1\rightarrow S_2$ тогда и только тогда, когда в случае активизации S_1 возможна активизация S_2. Если субъект S — такой, что для каждого Γ_t в вершину S не входит дуг, то такой субъект называется пользователем. Предполагается, что в системе Σ всего два пользователя: U_1 и U_2. Пользователи считаются активизированными по определению и могут активизировать другие субъекты.

Если в любой момент t в графе Γ_t в вершину S не входят дуги и не выходят дуги, то такие субъекты исключаются из рассмотрения.

Введено обозначение: $S_1\xrightarrow{*}S_2$ $S_1,S_2\in S_t$, процедуры активизации процессом S_1 субъекта S_2.

Далее в рамках модели делаются следующие предположения:

Предположение 1. Если субъект S активизирован в момент t, то существует единственный активизированный субъект S' в S_1, который активизировал S. В момент $t = 0$ активизированы только пользователи.

Лемма 1. Если в данный момент t активизирован субъект S, то существует единственный пользователь U, от имени которого активизирован субъект S, то есть существует цепочка $U \xrightarrow{a} S_1 \xrightarrow{a} S_2 \xrightarrow{a} \cdots \xrightarrow{a} S_k \xrightarrow{a} S$

Предположение1 требует единственности идентификации субъектов. Далее предполагается, что каждый объект в системе имеет уникальное имя.

Кроме активизации в системе Σ существуют и другие виды доступа активизированных субъектов к объектам.

Введено множество всех видов доступов R, $|R| < \infty$. Если $p \subseteq R$, то множество доступов p активизированного субъекта S к объекту O обозначится через $S \xrightarrow{p} O$. Если в некоторый промежуток времени $[t, t+k]$ реализована последовательность доступов

$$U \xrightarrow{a} S_1 \xrightarrow{a} S_2 \xrightarrow{a} \cdots \xrightarrow{a} S_k \xrightarrow{p} S,$$

то считается, что произошел доступ $S \xrightarrow{p} {}^{*}O$ от имени субъекта S к объекту O.

При этом не имеет значения, какую задачу решает система Σ, а лишь моделируется функционирование системы последовательностью доступов.

Предположение 2. Функционирование системы Σ описывается последовательностью доступов множеств субъектов к множествам объектов в каждый момент времени $t \in N$.

Описанный выше орграф Γ_1 обобщается путем добавления дуг $S \to O$, обозначающих возможность любого доступа субъекта S к объекту O в момент t, в случае активизации S.

Введено обозначение: $D_t(S) = \{O \mid S \to {}^{*} O \text{ в момент } t\}$,

где $S \to {}^{*} O$ означает возможность осуществления цепочки доступов

$S \to S_1 \to S_2 \to \cdots S_k \to O$ (возможность доступа к O от имени S). Тогда для любого $t \in N$ в системе определены $D_t(U_1)$ и $D_t(U_2)$.

Считается, что $D = D_t(U_1) \cap D_t(U_2)$ фиксировано для всех t, $O_t = D_t(U_1) \cup D_t(U_2)$, в начальный момент $t = 0$: $O_0 = \{U_1, U_2\} \cup D$

Определение 1. Множество объектов D называется общими ресурсами системы.

Средствами из D пользователь может создавать объекты и уничтожать объекты, не принадлежащие D. Создание и уничтожение каких-либо объектов является доступом в R к некоторым объектам из O (и к уничтожаемым объектам).

Из объектов системы Σ построена некоторая подсистема, которая реализует доступы. Любое обращение субъекта S за доступом p к объекту O в эту подсистему начинается с запроса, который обозначается $S \xrightarrow{p'} O$.

При порождении объекта субъект S обращается к соответствующей процедуре, в результате которой создается объект с уникальным именем. Тогда в силу леммы 1, существует единственный пользователь, от имени которого активизирован субъект, создавший этот объект, т.е. соответствующий пользователь породил данный объект.

Через $O_t(U)$ обозначено множество объектов из O_t, которые породил пользователь U. При этом считается, что $U \in O_t(U)$.

В случае, если в R есть доступы *read* и *write*, рассматривается только опасность утечки информации через каналы по памяти, которые могут возникнуть при доступах к объектам. Таким каналом может быть следующая последовательность доступов при $s < t$

$U_i \xrightarrow{w} O$ в момент s и $U_j \xrightarrow{r} O$ в момент t, $i \neq j$

При определенных условиях может оказаться опасным доступ от имени пользователя:

$U_i \xrightarrow{w} {}^{*} O$ в момент s и $U_j \xrightarrow{r} {}^{*} O$ в момент t, $s < t$, $i \neq j$

С некоторой избыточностью исчерпываются возможные каналы по памяти, если считать неблагоприятными какие-либо доступы $p_1, p_2 \subseteq R$ вида

$$U_i \xrightarrow{p1} {}^{*} O,\ U_j \xrightarrow{p2} {}^{*} O,\ i \neq j \qquad (1)$$

которые и считаются каналами утечки.

Предположение 3. Если $O \in \boldsymbol{D}$, то доступы в (1) при любых p_1 и p_2 не могут создать канал утечки.

Это значит, что предполагается невозможным отразить какую-либо ценную информацию в объектах общего доступа.

Тогда в (1) достаточно ограничиться объектами O, не лежащими в $\boldsymbol{D}$. Это значит, что в одном из доступов в (1) имеется $U_i \xrightarrow{p}{}^{*} O$, где $O \in \boldsymbol{O}_i(U_j)$, $i \neq j$. Таким образом, в системе считаются неблагоприятными доступы вида:

$$\exists t, \exists p \subseteq \boldsymbol{R},\ p \neq \varnothing, \exists U_i, \exists O \in \boldsymbol{O}_i,$$
$$U_i \xrightarrow{p}{}^{*} O,\ O \in \boldsymbol{O}_i(U_j),\ i \neq j \qquad (2)$$

то есть доступы от имени какого-либо пользователя к объекту, созданному другим пользователем. Такие доступы называются утечкой информации.

Предположение 4. Если некоторый субъект $S, S \in \boldsymbol{D}$, активизирован от имени пользователя U_i (т.е. $U_i \xrightarrow{a}{}^{*} S$), в свою очередь субъекту S предоставлены в момент t доступ к объекту O, то либо $O \in \boldsymbol{D}$, либо $O \in \boldsymbol{O}_i(U_i)$, либо система прекращает работу и выключается.

Определена следующая политика безопасности (ПБ):

Если $S \xrightarrow{p'} O$, то при $S, O \in \boldsymbol{O}_i(U)$ доступ $S \xrightarrow{p} O$ разрешается, если $S \in \boldsymbol{O}_i(U_i)$, $O \in \boldsymbol{O}_i(U_j)$, $i \neq j$, то доступ $S \xrightarrow{p} O$ невозможен.

Теорема 1. Пусть в построенной системе выполняются предположения 1 — 4. Если все доступы осуществляются в соответствии с ПБ, то утечка информации (2) невозможна.

Далее построено удобное для реализации множество «услуг» более низкого уровня, поддерживающих ПБ. То есть определено такое множество реализованных в системе Σ условий, что можно доказать теорему о достаточности выполнения этих условий для выполнения правил ПБ.

Условие 1. (Идентификация и аутентификация). Если для любых $t \in \boldsymbol{N}$, $p \subseteq \boldsymbol{R}$, $S, O \in \boldsymbol{O}_t$, $S \xrightarrow{p'} O$, то вычислены функции принадлежности S и O к множествам $\boldsymbol{O}_t(U_1)$, $\boldsymbol{O}_t(U_2)$, $\boldsymbol{D}$.

Условие 2. (Разрешительная подсистема). Если $S \in \boldsymbol{O}_t(U_i)$, $O \in \boldsymbol{O}_t(U_j)$ и $S \xrightarrow{p'}{}^{*} O$ в момент t, то из $i = j$ следует $S \xrightarrow{p} O$, и из $i \neq j$ следует $S \xrightarrow{p} O$ (не разрешается доступ).

Условие 3. (Отсутствие обходных путей политики безопасности). Для любых $t \in \boldsymbol{N}$, $p \subseteq \boldsymbol{R}$, если субъект S, активизированный к моменту t, получил в момент t доступ $S \xrightarrow{p} O$, то в момент t произошел запрос на доступ $S \xrightarrow{p'} O$.

Теорема 2. Если в построенной системе Σ выполняются предположения 1 — 4 и условия 1 — 3, то выполняется политика безопасности.

Теорема означает, что гарантированно выполненные условия 1 — 3 гарантируют выполнение политики безопасности.

Для рассмотрения вопроса о создании системы, в которой можно с достаточной степенью уверенности поддерживать функции 1 — 3, описана следующая архитектура:

1. В каждый момент только один пользователь может работать с системой. Физическое присутствие другого исключено.

2. При смене пользователей системы друг другом уходящий :

— записывает во внешнюю память все объекты, которые он хочет сохранить для дальнейших сеансов;

— выключает питание системы, после чего все содержимое оперативной памяти стирается, остаются записи на внешней памяти и ПЗУ, где хранятся объекты общего доступа.

3. Новый пользователь организует свою работу с включения системы и вызывает свои объекты из внешней памяти, опираясь на объекты общего пользования.

4. На шлюзе внешней памяти стоит шифратор K, который зашифровывает на текущем ключе k всю информацию, записываемую на внешнюю память, включая названия файлов. Наоборот, вся информация, поступающая из внешней памяти, расшифровывается на текущем ключе k. Внешняя память не имеет опции «просмотр директории», а любой запрос на выдачу файла функционирует так, что название запрашиваемого файла шифруется на текущем ключе k. При смене пользователей текущий ключ k автоматически стирается (вместе с содержимым оперативной памяти), а новый пользователь в качестве текущего устанавливает свой ключ.

Функционирование системы данной архитектуры позволяет реализовать все описанные выше свойства и, в частности, выполнить условия теорем 1 и 2.

Предположение 1 и другие допущения в описании системы вполне приемлемы для рассматриваемой архитектуры. Так как неблагоприятные состояния системы и политики безопасности выражены в терминах доступов, то для приемлемости предположения 2 достаточно, чтобы возможные вычислительные процессы однозначно отражались в терминах последовательностей доступов и значений функций принадлежности объектов к множествам $O_i(U_1), O_i(U_2), D$. Если объект только создан и находится в оперативной памяти, то доступ к нему со стороны процессов от имени создавшего пользователя автоматически разрешен и можно считать, что функция принадлежности вычислена. Если объект вызван из внешней памяти, то сам вызов и доступ к информации в объекте возможны, если установлен правильный ключ, что эквивалентно вычислению функции принадлежности к $O_i(U)$. Предположения 3 и 4 выполняются, так как вновь подключенный пользователь работает один и вызывает из ПЗУ функции и объекты D. В системе нет субъекта, реализующего разрешительную систему, она естественно реализована за счет того, что расшифрованная информация читается тогда и только тогда, когда в шифраторе K установлен нужный ключ. Если пользователь или процесс от его имени обращается за доступом к объекту на внешней памяти, то любой доступ разрешен, если ключ зашифрования объекта (ключ создателя объекта) совпадает с ключом текущего пользователя. Наоборот, при несовпадении ключей допуск автоматически не разрешается, так как имя объекта и его содержание не расшифровываются правильно.

Таким образом, автоматически вычисляются функции принадлежности процесса и объекта при обращении через внешнюю память, что обеспечивает выполнение условия 1. Также автоматически выполняется условие 2 о работе разрешительной системы. Условия 1 и 2 не касаются обращений процессов из D и к D. Поэтому вопросы идентификации здесь решаются за счет разделения сеансов пользователей и указанные условия выполняются.

Доступ к объекту возможен лишь при обращении к внешней памяти через шифратор, или в случае, когда объект создан в течение текущего сеанса, или вызван из ПЗУ. Если считать, что доступ к объектам в оперативной памяти автоматически опирается на данное пользователю разрешение на доступ к ним, а активизированными могут быть субъекты от его же имени, то можно считать, что любой доступ в этом случае выполняется в соответствии с условиями 1 и 2.

Что касается условия 3, то невозможность получить доступ минуя разрешительную систему определяется разнесенностью работы пользователей, отсутствием подслушивания, необходимостью расшифровывать информацию для получения доступа к ней. Это не касается объектов из D, или только созданных, где нет проблем из-за разнесенности сеансов.

Также считается, что отсутствует физическое проникновение и модификация системы.

Таким образом гарантии в построенной системе обеспечиваются, если

выполнены следующие требования:

— обеспечена работа только одного пользователя (охрана);

— отключается питание при смене пользователей;

— гарантированы стойкость шифратора K и сохранность в тайне ключей каждого пользователя;

— недопустимо физическое проникновение в аппаратную часть или подслушивание (охрана).

Согласно [5], объект в системе — это конечное множество слов, некоторого гипотетического языка $Я$, одно из которых определяет состояние объекта. Часть объектов является субъектами. Субъект — это описание некоторого преобразования информации в системе. Субъект активизируется активизированным субъектом. Если нет субъекта, который может активизировать субъект S, то S называется пользователем и обозначается U. Пользователь порождает объекты и сам входит во множество объектов, им порожденных (?!). Для каждого объекта, не являющегося общим ресурсом, существует породивший его пользователь.

Доступы к общим ресурсам не могут создать канал утечки (сам автор считает это предположение слишком сильным, и это верно). Утечкой информации являются доступы от имени одного пользователя к объекту, созданному (?) другим пользователем.

Политика безопасности формулируется следующим образом: доступ субъекта к объекту разрешается в том и только в том случае, если и субъект, и объект порождены одним пользователем.

Для такого описания и политики безопасности и построен пример гарантированно защищенной системы обработки информации.

Данное описание, как и любое другое, вполне может быть объектом критики. Так, пользователь является видом субъекта, субъект является видом объекта, а объект — это конечное множество слов некоторого гипотетического языка. При этом субъект — это описание некоторого преобразования информации в системе. Трудно представить себе конструктивное описание пользователя на некотором гипотетическом языке как некоторое преобразование информации в системе. Сложно представить себе и систему, в которой исключены РПВ (требование отсутствия модификации системы).

Тем не менее, для построенного примера гарантированно защищенной системы *действительно* доказаны реализуемость требований формализма и выполнение политики безопасности. Важнейшим здесь является то, что на базе описанного примера можно строить другие варианты гарантированно защищенных систем.

Ниже опишем один из возможных вариантов.

Сохраним архитектуру, предложенную в [5] в части пп. 1 — 3, а именно:

1. В каждый момент только один пользователь может работать с системой. Физическое присутствие другого исключено.

2. При смене пользователей системы друг другом уходящий :

— записывает во внешнюю память все объекты, которые он хочет сохранить для дальнейших сеансов;

— выключает питание системы, после чего все содержимое оперативной памяти стирается, остаются записи на внешней памяти и ПЗУ, где хранятся объекты общего доступа.

3. Новый пользователь организует свою работу с включения системы и вызывает свои объекты из внешней памяти, опираясь на объекты общего пользования.

Пункт 4 заменим с учетом возможностей управления логическими дисками и распределения информации по ним.

4.1. Внешняя память разбивается на логические диски, причем их количество совпадает с количеством пользователей. Каждый логический диск ставится в соответствие одному пользователю.

4.2. На логическом диске, соответствующем пользователю U_i размещаются $O(U_i)$.

4.3. На шлюзе внешней памяти устанавливается аппаратное средство, которое аутентифицирует пользователя U_i при включении питания системы и отключает все логические диски, кроме того, на котором размещены $O(U_i)$. При смене пользователя эта процедура повторяется.

Как в случае [5], функционирование системы данной архитектуры позволяет гарантировать защищенность. Действительно, если пользователь или процесс от его имени обращается за доступом к объекту на внешней памяти, то любой доступ разрешен, если объект размещен на разрешенном логическом диске. С другой стороны, если объект размещен на недоступном для пользователя диске, то и доступ к объекту автоматически не разрешается. Все прочие рассуждения сохраняются в соответствии с [5].

Таким образом, предложенная архитектура обеспечивает в рамках [5] гарантированную защищенность системы обработки информации.

Этот результат является весьма важным для нас, так как описываемый ниже (гл. 2 и 3)«Аккорд АМДЗ» обладает возможностью, описанной в п.4.3 и, следовательно, при предложенной политике безопасности обеспечивает гарантированную защищенность.

Рассматривая особенности предложенной в [5] модели, можно отметить следующие ее особенности:

1. Множество субъектов рассматривается как подмножество объектов, хотя интуитивно ясна их различная природа.

2. Предполагается, что доступ к объектам общего пользования не может привести к утечке. Это возможно только при весьма особой архитектуре системы, а эти особенности могут носить труднореализуемый характер. Как следствие, не рассматриваются вопросы активизации (загрузки) системы.

3. Предполагается, что в каждый момент только один пользователь может работать с системой. Такое ограничение реализуемо в системе на базе изолированной ПЭВМ, но нереализуемо для ЛВС.

4. Отсутствует возможность модификации системы.

5. В приведенной модели рассмотрена лишь одна из множества возможных политик безопасности, хотя по требованиям нормативных документов, система защиты должна давать возможность реализовывать любую разумную непротиворечивую политику безопасности.

Хорошим шагом в развитии моделей стала субъектно-объектная модель, предложенная [11, 21]. Рассмотрим ее основные положения.

4. СУБЪЕКТНО-ОБЪЕКТНАЯ МОДЕЛЬ (СО-МОДЕЛЬ)

Модель произвольной АС рассматривается в виде конечного множества элементов. Указанное множество можно разделить на два подмножества: множество объектов и множество субъектов. Будем считать разделение АС на субъекты и объекты априорным. Также будем считать, что существует априорный безошибочный критерий различения субъектов и объектов в АС (по свойству активности).

Обозначим: $O = \{O_i\}$ — множество объектов АС;

$S = \{S_i\}$ — множество субъектов АС.

При этом полагаем, что в любой дискретный момент времени множество субъектов АС не пусто. Причем в противном случае соответствующие моменты времени исключаются из рассмотрения и рассматриваются отрезки с ненулевой мощностью множества субъектов.

Рассматривая вопросы безопасности информации в АС, будем говорить о защищенности системы как о состоянии, описанном в терминах модели АС. При этом понятие защищенности является для системы внешним, априорно заданным.

Интегральной характеристикой, описывающей свойства защищаемой системы, является *политика безопасности s* как качественное (или качественно-количественное) описание свойств защищенности, выраженное в терминах, описывающих систему.

Описание политики безопасности включает:

1. Множество возможных операций над объектами;

2. Для каждой пары «субъект, объект» (S_i, O_j) назначение множества разрешенных операций, являющееся подмножеством всего множества возможных операций.

Сформулированы аксиомы защищенных АС, имеющие фундаментальное значение для всей теории информационной безопасности.

Аксиома 1. В защищенной АС всегда присутствует активная компонента (субъект), выполняющая контроль операций субъектов над объектами. Данная компонента фактически отвечает за реализацию некоторой политики безопасности.

Аксиома 2. Для выполнения в защищенной АС операций над объектами необходима дополнительная информация (и наличие содержащего ее объекта) о разрешенных и запрещенных операциях субъектов с объектами.

Аксиома 3. Все вопросы безопасности информации в АС описываются доступами субъектов к объектам.

Аксиома 4. Субъекты в АС могут быть порождены только активной компонентой (субъектами) из объектов.

Механизм порождения новых субъектов специфицируется следующим определением.

Определение 1. Объект O_i называется источником для субъекта S_m, если существует субъект S_j, в результате воздействия которого на объект O_i в АС возникает субъект S_m.

Введено обозначение:

$Create(S_j, O_i) \to S_k$ — из объекта O_i порожден субъект S_k при активизирующем воздействии субъекта S_j. *Create* назовем операцией порождения субъектов.

Операция *Create* задает отображение декартова произведения множеств субъектов и объектов на объединение множества субъектов с пустым множеством: $Create: S \times O \to S \cup \{\varnothing\}$

Считается, что если $Create(S_j, O_i) \to \varnothing$, то порождение нового субъекта из объекта O_i при активизирующем воздействии S_j невозможно.

Заметим также, что в рамках рассматриваемой модели в АС действует дискретное время и фактически новый субъект S_k порождается в момент времени $t+1$ относительно момента t, в который произошло воздействие порождающего субъекта на объект-источник.

С любым субъектом связан (или ассоциирован) некоторый объект (объекты), отображающий его состояние.

Определение 2. Объект O_i в момент времени t ассоциирован с субъектом S_m, если состояние объекта O_i повлияло на состояние субъекта в следующий момент времени (т.е. субъект S_m использует информацию, содержащуюся в объекте O_i).

Введем обозначение «множество объектов $\{O_m\}_t$ ассоциировано с субъектом S_i в момент времени t»: $S_i(\{O_m\}_t)$.

Свойство субъекта «быть активным» реализуется и в возможности выполнения действия над объектами. При этом необходимо отметить, что пассивный статус объекта необходимо требует существования потоков информации от объекта к объекту (в противном случае невозможно говорить об изменении объектов), причем данный поток инициируется субъектом.

Определение 3. Потоком информации между объектом O_m и объектом O_i называется произвольная операция над объектом O_i, реализуемая в субъекте S_i и зависящая от O_m.

Обозначим через:

$Stream(S_i, O_m) \to O_i$ — поток информации от объекта O_m к объекту O_i.

Из определения 3 также следует, что поток всегда инициируется (порождается) субъектом.

Определение 4. Доступом субъекта S_i к объекту O_i будем называть порождение потока информации между некоторым объектом (например, ассоциированным с субъектом объектами $S_i(\{O_m\})$) и объектом O_i.

Выделяется все множество потоков P для фиксированной декомпозиции АС на субъекты и объекты во все моменты времени (все множество потоков является объединением потоков по всем моментам дискретного времени) и произвольным образом разбивается на два непересекающихся подмножества: N и L, $P = N \cup L$, $N \cap L = \varnothing$.

Обозначим через:

N — подмножество потоков, характеризующих несанкционированный доступ,

L — подмножество потоков, характеризующих легальный доступ.

Определение 5. Правила разграничения доступа субъектов к объектам есть формально описанные потоки, принадлежащие подмножеству L.

Вводится понятие тождественности:

Определение 6. Объекты O_i и O_j тождественны в момент времени t, если они совпадают как слова, записанные в одном языке.

Например, при представлении в виде байтовых последовательностей объекты $O_1 = (o_{11}, o_{12}, \ldots, o_{1m})$ и $O_2 = (o_{21}, o_{22}, \ldots, o_{2k})$ одинаковы, если $m = k$ и $o_{1i} = o_{2i}$ для всех

$$i = \overline{1, k} \quad (o_{ij} \text{ — байты}).$$

Определение 7. Субъекты S_i и S_j тождественны в момент времени t, если попарно тождественны все ассоциированные с ними объекты.

Следствие (из определений 6 и 7). Порожденные субъекты тождественны, если тождественны порождающие субъекты и объекты-источники.

Модель АС рассматривается как совокупность взаимодействующих субъектов и объектов. При изменении объектов, которые функционально ассоциированы с субъектом реализации политики безопасности, могут возникнуть потоки, принадлежащие множеству N.

В связи с этим введено понятие корректности субъектов.

Определение 8. Субъекты S_i и S_j называются невлияющими друг на друга (или корректными относительно друг друга), если в любой момент времени отсутствует поток (изменяющий состояние объекта) между любыми объектами O_i и O_j ассоциированными, соответственно, с субъектами S_i и S_j. Причем O_j не является ассоциированным объектом с S_i, а O_i не является ассоциированным объектом S_j.

Смысл понятия корректности можно пояснить на примере: существующие в едином пространстве ОП программы не должны иметь функциональных возможностей изменения «чужого» вектора кода и состояния переменных.

Сформулировано и более жесткое определение.

Определение 9. Субъекты S_i и S_j называются абсолютно невлияющими друг на друга (или абсолютно корректными относительно друг друга), если в условиях определения 8 множества ассоциированных объектов указанных субъектов не имеют пересечения.

Ограничим множество порождаемых субъектов.

Определение 10. Монитор безопасности субъектов (МБС) — субъект, который разрешает порождение субъектов только для фиксированного подмножества пар активизирующих субъектов и порождающих объектов.

Воздействие МБС выделяет во всем множестве субъектов S подмножество E разрешенных субъектов.

Сформулируем определение изолированности АС.

Определение 11. Множество субъектов АС называется изолированным (абсолютно изолированным), если в АС активизирован МБС и субъекты из порождаемого множества корректны (абсолютно корректны) относительно друг друга и МБС.

Предположим, что зафиксировано состояние объекта O_m в некоторый момент времени t_0. Будем обозначать состояние объекта O_m в момент времени t как $O_m[t]$.

Определение 12. Операция порождения субъекта $Create\,(S_k, O_m) \to S_i$ называется порождением с контролем неизменности объекта, если для любого момента времени $t > t_0$, в который активизирована операция порождения объекта *Create*, порождение субъекта S_i возможно только при тождественности объектов $O_m[t_0]$ и $O_m[t]$.

Утверждение 1 (базовая теорема ИПС). Если в момент времени t_0 в изолированной АС действует только порождение субъектов с контролем неизменности объекта, и существуют потоки от любого субъекта к любому объекту, не противоречащие условию корректности (абсолютной корректности) субъектов, то в любой момент времени $t > t_0$ АС также остается изолированной (абсолютно изолированной).

Далее, в работах [11,21] сформулирована методология проектирования гарантированно защищенных АС. Приведен метод построения ИПС, а именно, метод субъектно-объектного взаимодействия в рамках ИПС.

Важную роль при проектировании ИПС играет свойство АС, заключающееся в поэтапной активизации субъектов из объектов различного уровня представления информации.

Этот вопрос рассмотрен на примере абстрактной операционной системы (ОС). В таблице 1 представлена иерархия уровней при загрузке ОС.

Таблица 1
Иерархия уровней при загрузке АС

Уровень	Субъект	Локализация	Представление информации	Через какие функции реализуются потоки
0	Субъект аппаратно-программного уровня	ПЗУ (Bios)	Сектора	Через микропрограммы ПЗУ
1	Субъект уровня первичной загрузки	Загрузчик ОС	Сектора	Через Bios или первичный загрузчик
2	Субъект уровня вторичного загрузчика (драйвер)	Драйверы ОС	Сектора	Через Bios или первичный загрузчик

В таблице выделен термин «сектор» для обозначения представления объекта аппаратно-программного уровня. Он обозначает непрерывную последовательность элементов хранения (байт) на материальном носителе, характеризуемую местом расположения.

Термин «файл» обозначает абстрактный объект, построенный по списочной структуре из объектов «сектор». Объекты типа «файл» и «сектор» выделены исключительно исходя из типовой архитектуры объектов АС.

С учетом иерархической структуры представления объектов можно говорить о том, что в начальные этапы активизации АС декомпозиция на субъекты и объекты динамически изменяется. Следовательно, основная теорема ИПС может быть применима только на отдельных интервалах времени, когда уровень представления объектов постоянен и декомпозиция фиксирована.

В АС выделяется конечное число уровней представления объектов $U = \{0,...,R\}$, где R — максимальный уровень представления объекта.

Практическая реализация всех операционных систем позволяет выделить две фазы их работы: активизация субъектов с ростом уровня представления объектов (фаза загрузки или начальная фаза) и фаза стационарного состояния (когда уровень представления объектов не увеличивается).

Вводится понятие последовательности активизации компонент АС.

Обозначим: Z_t последовательность пар $(i, j)_t$ ($t = 0, 1, 2, \ldots, L\text{-}1$ — моменты времени) длины L таких, что $Create\,(S_i, O_j)[t] \to S_m[t+1]$.

Обозначим также:

S_Z — множество всех субъектов, включенных в последовательность Z_L,

O_Z — множество всех объектов, включенных в последовательность Z_L.

Для многопотоковых АС можно рассматривать несколько (возможно, зависимых друг от друга) последовательностей Z_t и соответственно множеств S_Z и O_Z.

Утверждение 2 (условие одинакового состояния АС). Состояние АС в моменты времени $t_x^{(1)}$ и $t_x^{(2)}$ ($t_x^{(1)}$ и $t_x^{(2)}$ исчисляются для двух отрезков активности АС от нулевого момента активизации АС $t_0^{(1)}$ и $t_0^{(2)}$ — например, включения питания аппаратной части) одинаково, если:

1. $t_x^{(1)} = t_x^{(2)}$;
2. Тождественны субъекты $S_i[t_0^{(1)}]$ и $S_i[t_0^{(2)}]$;
3. Неизменны все объекты из множества O_Z;
4. Неизменна последовательность Z_L.

Необходимо заметить, что последовательность Z_t локализуется в некотором объекте, либо совокупности объектов (например, для *DOS* пос-

ледовательность активизации субъектов предопределена содержанием файлов *AUTOEXEC.BAT* и *CONFIG.SYS*) и неизменность последовательности Z_L тождественна неизменности указанных объектов .

При этом рассматриваемая модель предполагает выполнение следующих условий для контроля и управления элементами АС, включенными в последовательность активизируемых компонент:

1. Отсутствие возможности управления субъектами, принадлежащими множеству S_Z со стороны пользователя (в противном случае последовательность активизации субъектов может быть изменена).

2. Доступность для контроля неизменности всех объектов из множества O_Z.

3. Уровень представления информации не должен возрастать с некоторого момента времени (в данном случае имеется в виду, что существует такой момент времени t_x, когда для любого $t > t_x$ объект-аргумент O_i операции $Stream(S_i, O_i)_t$ принадлежит одному уровню представления).

Пусть в последовательности Z_L можно выделить z_i такое, что для любого z_k, $k > i$, отображения *Create* и *Stream* используют только объекты уровня R. Другими словами, с момента времени i наступает стационарная фаза функционирования АС.

В этих условиях, а также если:

— выполнено условие попарной корректности субъектов;

— активизирован МБС с контролем неизменности объектов-источников на уровне R

справедливо:

Утверждение 3 (достаточное условие ИПС при ступенчатой загрузке). При условии неизменности Z_L и неизменности объектов из O_Z в АС с момента времени установления неизменности Z_L и O_Z действует изолированная программная среда.

Таким образом, в рамках модели на временной оси можно выделить несколько точек, имеющих принципиальное значение, а именно:

$t = 0$ — момент включения питания аппаратной части;

$t = i$ — момент времени, в который наступает стационарная фаза функционирования АС;

$t = m > i$ — момент времени, в который начинает действовать изолированная программная среда;

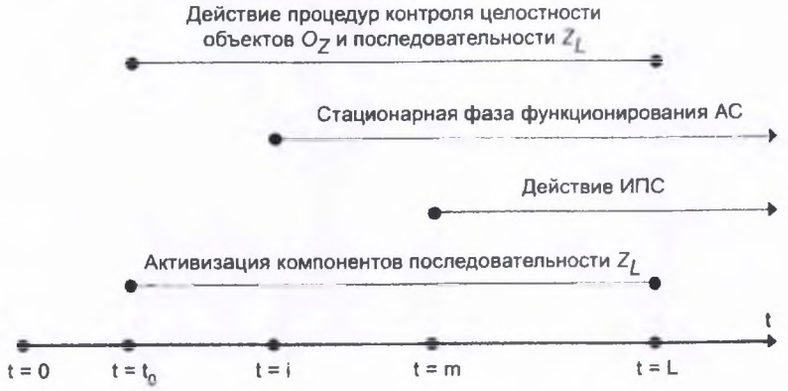
$t = L$ — момент времени, когда завершена активизация всех компонент АС, содержащихся в последовательности Z_L.

В рассматриваемой модели отмечено, что субъект контроля неизменности объектов, входящих в процедуры активизации АС и объектов, описывающих последовательность активизации компонент, должен быть активен уже на этапе работы субъектов аппаратно-программного уровня, но

его объект-источник технически не может быть проверен на неизменность. В связи с этим формулируется

Аксиома 5. Генерация ИПС рассматривается в условиях неизменности конфигурации тех субъектов АС, которые активизируются до старта процедур контроля целостности объектов O_Z и последовательности Z_L. Неизменность данных субъектов обеспечивается внешними по отношению к самой АС методами и средствами. При анализе или синтезе защитных механизмов свойства указанных субъектов являются априорно заданными.

С учетом аксиомы 5 схематическое представление этапов функционирования АС примет следующий вид



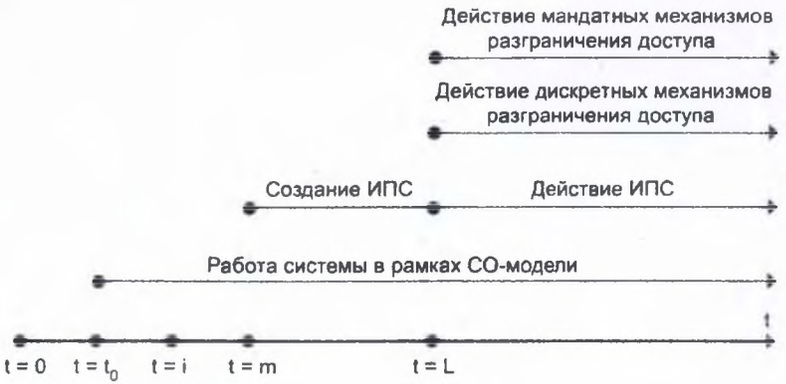
Таким образом, из рассмотрения СО-модели можно сделать следующие выводы.

Генерация ИПС в рамках модели возможна только на отрезке $[t_0, L]$. При этом неизменность субъектов, активизируемых на отрезке $[0, t_0]$, гарантируется априорно, вернее даже декларируется.

В СО-модели сделан по сравнению с другими моделями значительный шаг вперед, так как рассмотрены механизмы и методы создания ИПС как безопасных начальных состояний. Действительно, в условия всех основных моделей входит некоторое безопасное начальное состояние, но в рамках моделей не рассматриваются конструктивные механизмы его достижения.

СО-модель отвечает на этот вопрос, дополняя, а не отвергая другие модели. Формируется иерархия моделей, где выделяются этапы генерации ИПС, поддержка ИПС и разграничение доступа как с помощью мандатных так и дискреционных механизмов.

Схематически это можно изобразить следующим образом:



При этом СО-модели присущ и ряд серьезных недостатков. Так, например, очень сильным представляется требование аксиомы 5. Действительно, невыполнение требований не позволит создать ИПС, а как их реализовать, и что вообще означает «внешние априорно заданные средства», какими они должны быть?

На наш взгляд, ограниченность как СО-модели, так и других связана также и с некоторыми упрощениями, не всегда обоснованными. Так, в СО-модели отмечается, что «все вопросы безопасности в АС описываются доступами субъектов к объектам» (аксиома 3). Такая формулировка вызывает серьезные вопросы — например, разве не связаны с безопасностью состав и структура системы? В модели [5] отмечается, что «нас не интересует какую задачу решает система, мы лишь моделируем ее функционирование системы последовательностью доступов».

Рассмотрим надежность системы с точки зрения ее структуры и организации.

5. МУЛЬТИПЛИКАТИВНОСТЬ ЗАЩИТНЫХ СВОЙСТВ

Надежность системы может быть охарактеризована некоторым числовым параметром, который носит вероятностный характер. Обозначим его **P**. Параметр принимает значения от 0 до 1.

Рассмотрим систему, как множество занумерованных объектов $O = \{o_i\}_{i=1}^{N}$.

Будем полагать, что каждый объект системы $o_i \in O$ подвержен атакам злоумышленников $A_i = \{a_{ik}\}_{k=1}^{n}$.

Определим множество A атак на систему как совокупность атак на все объекты системы: $A = \bigcup_{i=1}^{N} A_i$.

Рассмотрим конечные вероятностные пространства:

$$(\Omega_1, Z_1, P_1), \ldots, (\Omega_N, Z_N, P_N) \qquad (1)$$

где $\Omega_i = \{\omega_{ik}\}_{k=1}^{n}$ — множество элементарных событий, каждое из которых есть срабатывание системы защиты АС в случае атаки a_{ik} на объект системы $o_i \in O$;

Z_i — множество всех подмножеств Ω_i;

$Z \in Z_i$, событие Z есть срабатывание системы защиты АС в случае атак на объект системы $o_i \in O$;

$P_i(Z) = \sum_{\omega_{ik} \in Z} p_i(\omega_{ik})$ — вероятность события Z, которая задается с помощью вероятностей элементарных событий $p_i(\omega_{ik})$, $\omega_{ik} \in \Omega_i$.

Построим вероятностное пространство (Ω, Z, P) как прямое произведение вероятностных пространств (1).

Тогда $\Omega = \Omega_1 \times \Omega_2 \times \ldots \times \Omega_N$ — пространство элементарных событий, такое что

$\omega \in \Omega \Leftrightarrow \omega = (\omega_1, \omega_2, \ldots, \omega_N)$, где $\omega_i \in \Omega_i$, $i = \overline{1,N}$.

Каждое из элементарных событий w есть срабатывание системы защиты АС в случае атак a_{ik_i}, $1 \leqslant k_i \leqslant n_i$ на соответствующие объекты системы $o_i \in O$, $i = \overline{1,N}$ (или атаки на систему).

Вероятность события w есть $p(\omega) = p_1(\omega_1) \cdot \ldots \cdot p_N(\omega_N)$.

$Z = Z_1 \times Z_2 \times \ldots \times Z_N$ — алгебра всех подмножеств Ω;

Событие $Z = Z_1 \times Z_2 \times \ldots \times Z_N$, $Z_i \in Z_i$, $i = \overline{1,N}$ — срабатывание системы защиты АС в случае атак на систему.

Для построенного таким образом класса событий Z, согласно [14], вероятность P является прямым произведением вероятностей P_i: $P = P_1 \times \ldots \times P_N$ и определяется по следующей формуле:

$$P(Z) = \sum_{\omega \in Z} p(\omega) = \sum_{\omega_1 \in Z_1} p_1(\omega_1) \cdot \ldots \cdot \sum_{\omega_N \in Z_N} p_N(\omega_N) = \prod_{k=1}^{N} P_k(Z_k) \qquad (2)$$

Таким образом, вероятность срабатывания системы защиты в случае атак на АС есть произведение вероятностей срабатывания системы защиты при атаках на объекты АС $o_i \in O$, а параметр **P**, характеризующий

надежность системы, определим как вероятность срабатывания системы защиты АС при атаках на систему, т.е. $\mathbf{P} = P(Z)$.

Из (2) видно, что надежность система обладает мультипликативным свойством, т.е. в случае, если хотя бы один объект системы $o_i \in O$ окажется незащищенным, говорить о надежности системы не имеет смысла.

Из этого факта следует Мультипликативная парадигма защиты — для построения защищенной системы нужно обеспечить защиту всех ее элементов. Свойство мультипликативности относится к наиболее общим закономерностям в среде безопасности, а именно: «степень безопасности системы определяется степенью безопасности ее самого «слабого» элемента» [18] или «итоговая прочность защищенного контура определяется его слабейшим звеном» [9].

Как отмечалось [3, 8], программных механизмов защиты информации явно недостаточно. В качестве некоторых исходных положений по созданию системы защиты принимаются выводы [3, 22] о том, что реализация функции защиты должна быть преимущественно аппаратная, и должно быть строго доказано обеспечение задаваемого уровня защиты. Хотя требование аппаратной реализации функции защиты здесь и сформулировано, оно само нуждается в строгом доказательстве, так как до сих пор даже широко распространенные на рынке средств защиты в лучшем случае используют аппаратную компоненту, но не базируются на ней. В [3] отмечено, что основной причиной неудачи является то, что вопросы защиты информации рассматриваются без органической связи с проектированием автоматизированных систем, т.е. вопросы защиты должны отражаться как в архитектуре, так и в технологии проектирования АС. С учетом этого тем более ограниченными выглядят утверждения вида : «Все вопросы безопасности описываются доступами субъектов к объектам».

Как было показано, с точки зрения архитектуры и технологии функционирования АС наиболее развитой является СО-модель. Однако и СО-модель нуждается в развитии, как это следует из мультипликативной парадигмы.

6. РАСШИРЕНИЕ СО-МОДЕЛИ

Ниже нас будет интересовать в первую очередь этап от включения системы до активизации механизмов, предусмотренных СО-моделью, так как с силу мультипликативности защитных свойств оставлять этот этап без защиты нельзя, и декларирования свойств недостаточно. В этом смысле предлагаемая модель является развитием СО-модели и позволяет установить ряд важных фактов, связанных с аппаратурной реализацией функций защиты.

Рассмотрим систему, имеющую линейную структуру.

Занумеруем объекты системы в порядке их инициализации: $o_1,\ldots,o_N$.

Множество объектов системы $O = \{ o_i \}_{i=1}^{N}$.

Определение 1. Связь между любой парой объектов $(o_i,\ o_{i+1})$ при проверке целостности системы будем описывать $(n_i + 1)$ — местной функцией проверки целостности объекта:

$$f_i = f_i(o_i) = f_i(o_i,\ p_{i1},\ldots,p_{in_i}) = o_{i+1},\ i = \overline{1, N-1},$$

где $p_{i1},\ldots,p_{in_i}$ — параметры объекта o_{i+1}. Функция $f_i,\ i = \overline{1, N-1}$ устанавливает целостность объекта o_{i+1}, если целостность объекта o_i зафиксирована. При этом функции $f_i(o_i) = o_{i+1},\ i = \overline{1, N-1}$ являются функциями следования согласно [8].

Определение 2. Будем говорить, что система является целостной, если установлена целостность каждого из ее объектов $o_i \in O,\ i = \overline{1, N}$

Утверждение (об установлении целостности системы). Целостность системы установлена тогда и только тогда, когда установлена целостность объекта o_N.

Доказательство.

1. Пусть система целостна, тогда из определения 2. следует, что установлена целостность всех объектов системы $o_i \in O,\ i = \overline{1, N}$. Следовательно, установлена целостность объекта o_N.

2. Пусть установлена целостность объекта o_N, т.е. известно значение функции проверки целостности $f_{N-1}(p_{N-1\,1},\ldots,p_{N-1\,n_{N-1}},\ o_{N-1}) = o_N$, но функция f_{N-1} определена только в том случае, когда установлена целостность объекта o_{N-1} и т.д.

Из целостности объекта o_2 следует, что целостность объекта o_1 зафиксирована.

Таким образом, зафиксирована целостность объектов системы $o_i \in O,\ i = \overline{1, N}$. Следовательно, по определению 2, система является целостной. Что и требовалось доказать.

Таким образом, задача установления целостности системы эквивалентна задаче установления целостности объекта o_N.

Пусть подмножество $M \subseteq O$ есть множество объектов системы, целостность которых установлена.

Определение 3. Множество M разрешимо, если существует алгоритм A_M, который по любому объекту o_i дает ответ, принадлежит o_i множеству M или не принадлежит.

Определение 3'. Множество M разрешимо, если оно обладает вычислимой всюду определенной (общерекурсивной) функцией χ_M, такой, что

$$\chi_M(o_i) = \begin{cases} 1, \text{ если } o_i \in M \\ 0, \text{ если } o_i \notin M \end{cases}$$

Определение 4. Множество M называется перечислимым, если оно является областью значений некоторой общерекурсивной функции, т.е. существует общерекурсивная функция $\psi_M(x)$, такая, что $o_i \in M$, если и только если для некоторого $x \in N$ $o_i = \psi_M(x)$.

Функция ψ_M называется перечисляющей для множества M.

Из [8] известна

Теорема (о разрешимости). Множество M разрешимо, если и только если M и $\overline{M}$ перечислимы.

Определение 2'. Будем говорить, что система является целостной, если M – разрешимое множество и M совпадает с O: $M = O$.

Теперь может быть сформулирована

Теорема об использовании компонента безопасности (ИКБ). Задача контроля целостности системы разрешима только при использовании специализированного аппаратного резидентного компонента безопасности (РКБ).

Доказательство.

1. Покажем, что без дополнительного компонента построение системы с установлением целостности невозможно.

Построим перечисляющую функцию для множества M. Она должна быть общерекурсивной, т.е. всюду определенной на O частично-рекурсивной функцией.

Определим функцию ψ_M следующим образом:

$$\psi_M(i) = o_i \qquad (1)$$

Тогда ее можно представить через функции проверки целостности объектов f_i (см. опр. 1). Для этого определим семейство операторов суперпозиции $\left\{S^{n_i+1}_{n_{i-1}+1}\right\}$:

$$S^{n_i+1}_{n_{i-1}+1}(f_{i+1},\ f_i) = f_{i+1}(o_i,\ f_i(o_i)) \qquad (2)$$

В этом случае перечисляющая функция примет вид:

$$\psi_M(i) = S^{n_i+1}_{n_{i-1}+1}(f_{i+1},\ f_i)$$

При этом ψ_M является частично-рекурсивной функцией, так как построена из функций f_i, которые являются элементарными как функции следования [8], путем последовательного к ним применения операторов суперпозиции $S^{n_i+1}_{n_{i-1}+1}$.

Однако, ψ_M не является всюду определенной, т.к. значение функции f_i в точке o_i не определено. Это объясняется тем, что не определена (согласно определению 1) связь f_0, которая устанавливала бы целостность объекта o_i. Причиной неопределенности функции f_0 является отсутствие объекта, который являлся бы областью определения функции f_0 и целостность которого была бы зафиксирована.

Следовательно, нельзя построить перечисляющую функцию множества M и множество M не является перечислимым.

В таком случае, согласно теореме о разрешимости, множество M не является разрешимым, и, по определению 2', система не является целостной.

2. Покажем, что при использовании РКБ построение целостной АС возможно.

Добавим в систему элемент o_0, целостность которого достоверно определена. Рассмотрим систему с множеством объектов $O^0 = O \cup \{o_0\}$.

Определим связь f_0, устанавливающую целостность объекта o_1:

$$f_0(o_0) = f_0(o_0, p_{11}, \ldots, p_{1n_1}) = o_1$$

Тогда перечисляющая функция ψ_M множества M будет задана согласно (1) следующим образом:

$$\begin{cases} \psi_M(0) = f_0(o_0) = o_1 \\ \psi_M(i) = S^{n_{i+1}+1}_{n_{i+1}+1}(f_{i+1},\ f_i) = o_{i+1}, \end{cases}$$

т.е. функция является частично-рекурсивной и определена на всем множестве O. Следовательно множество M – перечислимо.

Определим $\overline{M} = O^0 \setminus M = \{o_0\}$.

Перечисляющая функция $\psi_{\overline{M}}$ для множества $\overline{M}$ может быть определена как функция аутентификации для объекта o_0.

$$\begin{cases} \psi_{\overline{M}}(0) = o_0, \text{ т.е. } o_0 \in \overline{M} \\ \psi_{\overline{M}}(i) = o_i,\ \{o_i\}_{i=1}^{N} = O, \text{ т.е. } o_i \notin \overline{M} \end{cases}$$

В этом случае $\psi_{\overline{M}}$ является всюду определенной, вычислимой, т.е. общерекурсивной функцией, следовательно множество $\overline{M}$ является перечислимым.

Тогда, по теореме о разрешимости, M есть разрешимое множество, следовательно, в силу определения $2'$, целостность системы может быть установлена. Что и требовалось доказать.

Из доказанной теоремы ИКБ следует:

Следствие 1 (принцип Архимеда). Установление целостности возможно только при расширении АС специализированным резидентным компонентом безопасности (РКБ).

Следствие 2 (синдром Мюнгхаузена). Установление целостности АС не возможно только за счет программных средств без использования резидентного компонента безопасности (РКБ).

Рассмотрим теперь вопрос о размещении компонентов безопасности (РКБ) в системе, описываемой на рассматриваемом этапе линейной структурой.

Будем полагать, что КБ внедрен в систему как объект o_0.

Утверждение (о размещении РКБ в системе линейной структуры). Компоненты безопасности (РКБ) могут быть размещены в системе линейной структуры произвольным образом, при условии, что в системе присутствует объект o_0.

Доказательство.

Рассмотрим вопрос об установлении целостности объекта системы o_k, $k \leqslant N$.

1. Пусть $k = 1$. В этом случае установление целостности объекта o_1 осуществляется на основании априорно определенной целостности объекта o_0. Для этого, с помощью вычислимой функции f_0 определяется связь, устанавливающая целостность объекта o_1: $f(o_0) = o_1$

2. Пусть $2 \leqslant k \leqslant N$. Тогда, имея в системе объект o_0, можно установить целостность интересующего объекта o_k с помощью связей $f_i(o_i) = o_{i+1}$, $i = \overline{1,k-1}$ и рекурсивно определенной функции

$$\begin{cases} \psi(0) = f_0(o_0) = o_1 \\ \psi(i) = S^{n_{i+1}+1}_{n_{i+1}+1}(f_{i+1},\ f_i) = o_{i+1},\ i = \overline{1,k-1} \end{cases}$$

где $S^{n_{i+1}+1}_{n_{i+1}+1}(f_{i+1},\ f_i) = f_{i+1}(o_i,\ f_i(o_i))$ — оператор суперпозиции.

Выберем произвольным образом объект системы o_i, $1 < i < k$. Выделим из множества объектов системы $O = \{o_i\}_{i=1}^{N}$ два следующих подмножества:

$$O' = \{o_1, \ldots, o_i\} \text{ и } O'' = \{o_{i+1}, \ldots, o_k\}$$

На множестве O' рассмотрим задачу установления целостности объекта o_i.

Аналогично задаче для множества объектов системы $O=\{o_i\}_{i=1}^{N}$ на множестве O' можно определить связи $f_j'(o_j)=o_{j+1}$, $j=\overline{1,i-1}$ и функцию

$$\begin{cases}\psi'(0)=f_0'(o_0)=o_1\\ \psi'(i)=S_{n-i+1}^{n-i+1}\left(f_{i+1}',\ f_i'\right)=o_{i+1},\ i=\overline{1,k-1}\end{cases}$$

устанавливающую целостность объекта o_i.

Это возможно, если определена связь f_0, т.е. когда в системе присутствует объект o_i с достоверно установленной целостностью.

Будем считать, что целостность объекта o_i установлена.

На множестве O'' рассмотрим задачу об установлении целостности объекта o_k. В данном случае определения связей $f_j''(o_j)=o_{j+1}$, $j=\overline{i+1,k}$ и функции $\psi''(j)$, $j=\overline{i+1,k-1}$ недостаточно для установления целостности объекта o_k. Необходимо присутствие элемента, целостность которого установлена достоверно. В качестве такого элемента может быть выбран объект системы o_i, в силу того, что его целостность является результатом решения задачи для множества O'.

Итак, у нас имеется возможность вычислить функции ψ' и ψ'', т.е. можно построить машины тьюринга T' и T'' с соответствующими функциональными возможностями.

Из [16] известна

Теорема (о программировании последовательной композиции). Каковы бы ни были тьюринговы программы A и B, может быть эффективно построена тьюрингова программа C такая, что для всех рассматриваемых слов P:

$$C(P)=B(A(P));$$

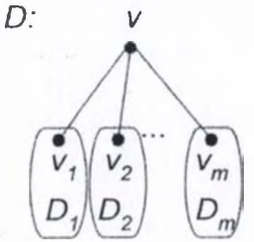
согласно которой может быть построена последовательная композиция машин T' и T'', а именно, машина T, такая, что

$$T(o_0)=T''(T'(o_i))=o_k$$

Таким образом, можно установить целостность элемента o_k, опираясь при этом на гарантированную целостность объекта o_i, если только возможно установить целостность отрезка $[o_1,\ o_i]$, или, что тоже самое, опираясь на РКБ, помещенный в точке i, если возможно установить целостность объектов $\{o_i\}_{i=1}^{i}$.

В силу произвольности выбора объекта o_i, можно утверждать, что полученный результат справедлив для любого i, $1<i<k$. Что и требовалось доказать.

Рассмотрим теперь систему, имеющую нелинейную структуру, а именно такую, которая может быть представлена в виде упорядоченного корневого дерева [13].



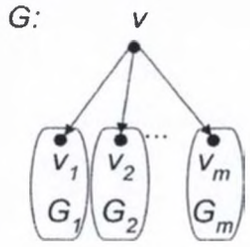
где D_i, $i=\overline{1,m}$ — упорядоченные корневые деревья:

При этом компонентам системы соответствуют вершины дерева D. Возможным связям между компонентами системы соответствуют ребра дерева.

Компонента системы считается активизированной, если существует ребро, исходящее из соответствующей этой компоненте вершины дерева.

Задача установления целостности рассматриваемой системы эквивалентна задаче установления целостности всех висячих (концевых) вершин дерева D.

1. Процесс последовательной по уровням иерархии инициализации системы описывается ориентированным графом G, который построен из описанного выше дерева D путем приписывания всем дугам направления от нижестоящей вершины к вышестоящей.



Сначала активизируется компонента системы, соответствующая корневой вершине v графа G, затем компоненты, принадлежащие второму уровню иерархии и т.д. При этом инициализация каждой ветви системы, соответствующей одному из подграфов G_i $i=\overline{1,m}$, осуществляется независимо ото всех остальных ветвей системы, которым соответствуют подграфы G_j, $j=\overline{1,m}$, $j\neq i$.

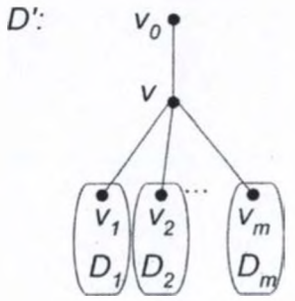
Выберем произвольным образом подграф G_i.

Выберем любую концевую вершину из подграфа G_i. Обозначим ее v_K.

Рассмотрим маршрут, а именно, простую цепь от корневой вершины v_i подграфа G_i к выбранной концевой вершине. Очевидно, что простая цепь является единственной. Любой другой маршрут от v_i к v_K будет содержать хотя бы одну из вершин и хотя бы одно из ребер не менее 2 раз, то есть такой маршрут не будет являться простой цепью.

Тогда рассматриваемая задача установления целостности элемента, соответствующего концевой вершине $v_K \in G_i$, эквивалентна задаче установления целостности системы линейной структуры. Эта задача была рассмотрена выше.

Таким образом, согласно теореме ИКБ, для установления целостности элемента $v_K \in G_i$ необходимо внедрение в систему специализированного резидентного компонента безопасности (РКБ). Причем для рассматриваемой системы КБ будет определен как вершина v_0, и тогда система со всеми возможными связями может быть представлена в виде упорядоченного корневого дерева D':



При этом после последовательной по уровням иерархии инициализации целостность системы будет проконтролирована.

2. Рассмотрим теперь случай, когда инициализация системы выполняется в произвольном порядке. Процесс такой инициализации представляется графом G, который содержит все вершины, присутствующие в дереве D. При этом, в силу произвольного порядка инициализации системы, говорить о наличии связей между всеми компонентами системы невозможно. Следовательно, в общем случае утверждение относительно наличия или отсутствия любого из ребер графа G невозможно, т.е. граф G будем считать несвязным.

Не ограничивая общности предположим, что в системе активизированы несколько (n) подветвей. В графе G им соответствуют компоненты связности, т.е. не соединенные друг с другом части. Обозначим их g_i, $i=\overline{1,n}$. Эти компоненты являются связными графами, а именно имеют структуру упорядоченного корневого дерева. Инициализация ветвей системы, соответствующих компонентам связности g_i, $i=\overline{1,n}$, осуществляется последовательно по уровням иерархии.

В таком случае задача контроля целостности всей системы будет эквивалентна задаче контроля целостности всех активизированных компонент системы, соответствующих концевым вершинам графа G, т.е. концевым вершинам компонент связности $g_i \in G$, $i=\overline{1,n}$

Выберем произвольным образом одну из компонент связности g_i. Пусть v_0^i — корневая вершина для графа g_i. Тогда согласно случаю, рассмотренному в п.1., для контроля целостности подветви системы, соответству-

ющей рассматриваемому графу g_i, необходимо разместить специализированный РКБ в вершине v_0^i. Обозначим его РКБ$_i$. Аналогично, для контроля целостности всех остальных компонент связности $g_i \in G$, $i = \overline{1,n}$ необходимо размещение специализированных КБ в их корневых вершинах v_0^i, $i = \overline{1,n}$.

В силу произвольного выбора компонент связности графа G справедливо следующее

Утверждение (о размещении РКБ в системе произвольной структуры). Для контроля целостности всей системы в целом необходимо размещение РКБ во всех вершинах графа, кратность которых больше 2, т.е. имеющих по крайней мере 2 ребра, связывающие их с вершинами следующего уровня иерархии.

Относительно структуры компонента безопасности можно утверждать следующее.

Утверждение (о структуре РКБ). Структура компонента безопасности (КБ) должна быть переменной для обеспечения функционирования АС на всех этапах жизненного цикла.

Выделим два основных этапа работы АС:

— этап формирования политики безопасности или этап управления;

— этап работы АС в пользовательском режиме.

Предполагаем, что описание политики безопасности, согласно которой функционирует АС, зафиксирована в АС в некоторой базе данных в виде правил разграничения доступа дискреционного механизма, меток конфиденциальности мандатного механизма и т.п.

1. Предположим, что РКБ обладает функциональными возможностями R (чтение) и W (запись). Функционирование АС на этапе управления при такой структуре РКБ обеспечивается.

В случае пользовательского режима работы АС, злоумышленник имеет возможность успешно внедрить РПВ.

Приведем пример такого РПВ.

Если злоумышленник, является легальным пользователем АС, то используя функциональные возможности РКБ, он может сначала выяснить свои права доступа к интересующему его объекту (с помощью возможности R), а затем заменить свои права на интересующие (с помощью возможности W). Очевидно, что можно построить машину Тьюринга, которая будет реализовывать данный механизм.

2. Предположим, что РКБ обладает только возможностью R.

В этом случае функционирование АС в пользовательском режиме происходит согласно политике безопасности, т.е. контролируется доступ пользователя к интересующему его объекту. При этом доступ либо разрешается, либо

запрещается в соответствии с информацией, прочитанной из базы данных, содержащей правила функционирования АС.

Рассмотрим работу АС на этапе управления. Предположим, что возникнет необходимость санкционированного изменения политики безопасности, зафиксированной в АС. Предполагаемые изменения никаким образом не могут быть внесены в АС, так как РКБ не имеет функциональной возможности запись.

В результате, при предполагаемой структуре РКБ, обеспечивается функционирование АС на этапе работы в пользовательском режиме и не обеспечивается функционирование АС на этапе управления.

3. Предположим, что РКБ обладает только возможностью W.

В этом случае обеспечивается функционирование АС на этапе управления, т.е. имеется возможность легально формировать новую политику безопасности путем внесения изменений (записи) в базу данных.

Работу АС, не противоречащую принятой политике безопасности в пользовательском режиме, при данной структуре РКБ обеспечить невозможно, так как РКБ не имеет функциональной возможности R для выяснения прав пользователя на интересующий его объект. Кроме того, имеющаяся у РКБ возможность W делает возможным несанкционированное изменение базы данных, т.е. дает злоумышленнику возможность путем записи в базу данных получить необходимые ему права на интересующий его объект.

В результате, рассмотренная структура РКБ обеспечивает функционирование АС на этапе управления и не обеспечивает работу АС в пользовательском режиме.

Таким образом из пп. 1 — 3 видно, что никакая фиксированная структура РКБ не может на обоих рассмотренных этапах обеспечить функционирование АС, не противоречащее политике безопасности. Можно сделать вывод о том, что структура РКБ должна быть переменной.

Изменение структуры РКБ определяется завершением этапа управления и переходом к этапу работы АС в пользовательском режиме.

Структура РКБ может изменяться под действием внутреннего (программного) или внешнего воздействий.

Предположим, что РКБ меняет свою архитектуру после сигнала, получаемого от программного элемента. Но в таком случае возможно злоумышленное вмешательство путем подмены программного элемента, посылающего сигнал.

Пусть на этапе управления действие РКБ описывается тьюринговой программой T_y, а на этапе работы АС в пользовательском режиме — тьюринговой программой $T_п$. Тогда согласно теореме о программировании последовательной композиции [16] может быть построена тьюрингова программа T, работающая таким образом, что нарушитель имеет возможность узнать свои права доступа к интересующему его объекту, внедрить РВП, подающее сигнал об изменении архитектуры РКБ, и, получив доступ к функциональной возможности W РКБ, несанкционированно изменить свои права

на интересующий его объект. Такая ситуация эквивалентна структуре РКБ, рассмотренной в п.1.

В результате, можно сделать следующий вывод:

Структура РКБ должна быть переменной, причем изменение структуры должно быть внешним относительно АС, в которую внедрен РКБ, т.е. необходимо изменение самого РКБ.

Внедрение с систему компонента безопасности связано с определением перечисляющей функции как функции аутентификации, применение которой должно установить подлинность РКБ (объекта o_0), так как его целостность зафиксирована технически. Это означает, что на основании выработанного РКБ сигнала (маркера) должна быть установлена:

— подлинность o_0 и подтверждение того, что маркер аутентификации был действительно сгенерирован o_0 ;

— целостность и актуальность переданного маркера аутентификации;

— подлинность объекта o_1, аутентифицирующего o_0 и подтверждение того, что маркер аутентификации был действительно предназначен для o_1.

Этот подход к аутентификации основан на механизме демонстрации обладания личным ключом, и относится к процедуре однонаправленной аутентификации, определенной [4]. В нашем случае при однонаправленной аутентификации выполняются следующие шаги:

1. РКБ создает r (неповторяющийся номер), который используется для обнаружения повторений и предотвращения подделок.

2. РКБ посылает o_1 следующее сообщение (маркер):

$$\text{РКБ}\{t, r, o_1^t\},$$

где:

$A\{I\}$ — информация I с электронно-цифровой подписью (ЭЦП) на ключе абонента A;

t — отметка времени создания маркера;

o_1^t — идентификатор адресата (в нашем случае — объекта o_1).

3. Получатель (o_1) выполняет следующие действия:

— проверяет ЭЦП на открытом ключе РКБ, и тем самым целостность и подлинность информации;

— проверяет, что маркер адресован ему;

— проверяет актуальность t и новизну r.

При успешном завершении проверок o_0 считается аутентифицированным.

Один из основных принципов строгой аутентификации состоит в том, что секретный ключ отправителя остается защищенным. Отсюда следует, что РКБ должен обладать средствами надежного хранения и применения ключа. С другой стороны, получатель маркера (объект, относительно которого выполняется аутентификация — в нашем случае o_1) должен иметь память для хранения ключа общего пользования.

Необходимость выработки для аутентификации ЭЦП отправителя маркера диктует также следующее:

— РКБ должен обладать ресурсами, позволяющими вырабатывать ЭЦП;

— для выработки ЭЦП необходимо применять хэш-функцию.

Вопросы выработки ЭЦП здесь не рассматриваются. Достаточно подробно они описаны в [7, 20]. Отметим, что хорошим механизмом для выработки r является генерация случайных чисел.

Строгая хэш-функция должна удовлетворять следующим требованиям [4]:

а) она должна быть однонаправленной, то есть при любом заданном результате хэширования вычислительным способом должно быть невозможно построить входное сообщение, хэширование которого дало бы такой же результат;

б) она должна быть свободна от конфликтов, то есть вычислительным способом должно быть невозможно построить два различных входных сообщения, хэширование которых дало бы одинаковый результат.

Известен [30] метод аутентификации IP-потоков, получивший название «*MD5* с ключом». Суть его состоит в том, что для аутентификации вычисляется маркер по следующему механизму:

$$MD5(key, datagram, key),$$

т.е. в качестве маркера используется значение хэш-функции, вычисляемой по алгоритму *MD5* [31] от конкатенации секретного ключа, данных и снова секретного ключа. В нашем случае маркер мог бы выглядеть так:

$$h(Key, r, Key),$$

где $h(I)$ — хэш-функция (не обязательно *MD5*) от данных I.

Практически одновременно с [30] нами был предложен и реализован практически метод кодов аутентификации (КА), где КА вычисляется как

$$\text{КА} = h(datagram, key)$$

или для нашего случая

$$\text{КА} = h(r, Key).$$

Механизм КА описан в Главе 2, здесь же нас интересует лишь возможность применения для построения функции аутентификации механизма хэш-функции с ключом.

Действительно, применение для выработки маркера механизма КА целесообразно, так как требования к ресурсам РКБ при этом только снижаются — остаются требования по надежному хранению ключей, вычислению хэш-функции, генерации случайного числа, снимаются требования по выработке и проверке ЭЦП — а это весьма сложные с точки зрения затрат ресурсов операции.

На основе приведенного выше анализа можно сформулировать требования к компоненту безопасности, а именно следующие:

1. РКБ — активный элемент, имеющий перестраиваемую структуру.
2. РКБ должен иметь возможность надежного хранения и применения ключей.
3. РКБ должен иметь аппаратный датчик случайных чисел.
4. РКБ должен иметь возможность вычисления надежной хэш-функции.
5. РКБ должен иметь возможность выполнения контрольных процедур до загрузки ОС.

Ниже будут рассмотрены дополнительные требования к РКБ и вопросы его проектирования.

Литература

1. Абрамов В.А. «Введение в теорию систем детерминированного, стохастического и нечеткого типа». МИЭТ, Москва, 1980.

2. Аккорд 1.95. Описание применения. 1143195.4012-003 31, ОКБ САПР, Москва, 1997.

3. Герасименко В.А. «Проблемы защиты данных в системах их обработки». Зарубежная радиоэлектроника, №12, 1989

4. ГОСТ Р ИСО/МЭК 9594-8-98. Информационная технология. Взаимосвязь открытых систем. Справочник. Часть 8. Основы аутентификации. ГОССТАНДАРТ РОССИИ, Москва, 1998.

5. Грушо А.А., Тимонина Е.Е. «Теоретические основы защиты информации». Москва, Яхтсмен, 1996.

6. Зегжда П.Д., Зегжда Д.П., Семьянов П.В., Корт С.С., Кузьмич В.М.,Медведовский И.Д., Ивашко А.М., Баранов А.П. «Теория и практика обеспечения информационной безопасности». Москва, Яхтсмен, 1996.

7. Зима В.М., Молдавян А.А., Молдавян Н.А. «Компьютерные сети и защита передаваемой информации». Санкт-Петербург, СпбГУ, 1998.

8. Кузнецов О.П., Адельсон-Вельский Г.М. «Дискретная математика для инженеров». Москва, Энергоатомиздат, 1988.

9. Мельников В.В. «Защита информации в компьютерных системах». Москва, Финансы и статистика, 1997.

10. «Программно-аппаратный комплекс защиты компьютера от несанкционированного доступа Dallas Lock 4.0». Санкт-Петербург, Конфидент, 1997.

11. Прокофьев И.В., Шрамков И.Г., Щербаков А.Ю. «Введение в теоретические основы компьютерной безопасности». Москва, МИФИ, 1998.

12. «Средства вычислительной техники. Защита от несанкционированного доступа и информации. Показатели защищенности от несанкционированного доступа к информации». Руководящий документ. Гостехкомиссия России. Москва,Военное издательство, 1992.

13. Сапоженко А.А., Рыбко А.И. «Элементы теории графов и схем». Методическая разработка. Москва, МГУ, 1991.

14. Севастьянов Б.А. «Курс теории вероятностей и математической статистики». Москва, Наука, 1982.

15. «Система разграничения доступа Secret Net v.1.10». Руководство. SafeWare Group.

16. Трахтенброт Б.А. «Алгоритмы и вычислительные автоматы». Москва, Советское радио, 1974.

17. Ухлинов Л.М. «Управление безопасностью информации в автоматизированных системах». Москва, МИФИ, 1996.

18. Ходаковский Е.А. «Системология безопасности». Безопасность. Информационный сборник фонда национальной и международной безопасности. №7-9(39) 1997. Москва, стр. 178-185

19. Хоффман Л.Дж. «Современные методы защиты информации». Москва, Советское радио, 1980.

20. Романец Ю.В., Тимофеев П.А., Шаньгин В.Ф. «Защита информации в компьютерных системах и сетях». Москва, Радио и связь, 1999.

21. Щербаков А.Ю. «Методы и модели проектирования средств обеспечения безопасности в распределенных компьютерных системах на основе создания изолированной программной среды». Автореферат на соискание степени доктора технических наук. Москва, 1997, на правах рукописи.

22. Вулф А. «Аппаратные средства, обеспечивающие защиту информации ЭВМ». Электроника. Т.58, №18, 1985.

23. Bell D.E., La Padulla J. «Security Computer System: A Mathematical Model». Bedford, Massachusetts: Mitre Corp., 1973, №11.

24. Benson G., Appelbe W., Akyldiz I. «The hierarhicical model of distributed system security. IEEE Symposium on Security and Privacy». Oakland 1989.

25. Clements D., Hoffman L.J. «Computer Assisted Security System Design». ERL Memo M-468, Electronics Research Laboratory, University of California, Berkeley, Nov. 1974.

26. Jeremy J. Security Specifications. IEEE Symposium on Security and Privacy. Oakland 1988.

27. Harrison M.A., Ruzzo W.L., J. Security Specifications. IEEE Symposium on Security and Privacy. Oakland 1988.

28. Landwerh C., Heitmeyer C., McLean J. «A security model for military message». ACM Transactions on Computer System, 1984. V.2. №3.

29. McLean J. «Reasoning About Security Models». Proceedings IEEE Symposium on Privacy and Security, Oakland, CA April 1987, IEEE Computer Society Press, 1987.

30. Metzger P., Simpson W. «IP Authentication using Keyed MD5». RFC-1828, August 1995.

31. Rivest.R. «The MD5 Message-Digest Algoritm». RFC 1321, MIT and RSA Data Security, Inc., April 1992.

32. Shandhu R. «The Schematic Protection Model: It's Definition and Analisis for Acyclic Attenuating Scheme». Journal of ACM 1988 v.35 №2.

33. Shandhu R. «Transformation of access right». IEEE Symposium on Security and Privacy. Oakland 1989.

34. Wisoman S., Terry D. «A new security policy mode». IEEE Symposium on Security and Privacy. Oakland 1989.

35. Zadeh L.A. «The Concept of a Linguistic Variable and Its Application to Approximate Reasoning». Memo ERL-M411, Electronics Research Laboratory, University of California, Berkeley, Oct. 15, 1973.

Глава 2.
МЕТОДЫ И МЕХАНИЗМЫ АППАРАТНОЙ ЗАЩИТЫ

Еще не повсеместно развеяно одно из наиболее значительных заблуждений в сфере информационной безопасности, заключающееся в том, что обеспечить достаточный уровень защиты якобы можно и без применения специализированных аппаратных средств.

Основные идеи аппаратной защиты состоят в следующем:

— признании *мультипликативной парадигмы защиты*, и, как следствие, равное внимание реализации контрольных процедур на всех этапах работы АС (цветная иллюстрация **А**);

— «материалистическом» решении *«основного вопроса» информационной безопасности: «Что первично — hard или soft?»* (цветная иллюстрация **В**);

— последовательном отказе от программных методов контроля, как очевидно ненадежных (преодоление *«синдрома Мюнхгаузена»,* цветная иллюстрация **Г**) и перенос наиболее критичных контрольных процедур на аппаратный уровень (*«принцип Архимеда»,* цветная иллюстрация **Д**);

— максимально возможном разделении условно-постоянных (программы) и условно-переменных (данные) элементов контрольных операций (принцип *«отчуждай и властвуй»,* цветная иллюстрация **Е**).

Электронный документооборот в АС основан на следующей взаимосвязи субъектов и объектов:

оператор
за **ПЭВМ** *в* **ЛВС**
с **ОС**,
используя **ППО**
и **данные**,
формирует **ЭлД**,
передаваемый в **АС**,
обрабатываемый в **АС**,
хранимый в **АС**,
исполняемый в **АС** .

Это означает, что применяемые технические средства должны обеспечивать достаточный уровень информационной безопасности на следующих этапах:

— идентификация/аутентификация (ИА) пользователей;
— контроль целостности технического состава ПЭВМ и ЛВС;
— контроль целостности ОС;
— контроль целостности ППО и данных;
— аутентификация документа при его создании;
— защита документа при его передаче;
— аутентификация документа при обработке, хранении и исполнении документа;
— защита документа при доступе к нему из внешней среды.

Понимая, что абсолютно надежных средств защиты не существует, и полагая, что злоумышленник в состоянии преодолеть защиту на любом этапе, если только не доказано обратное (гарантированная защита), в дальнейшем будем придерживаться не аддитивной, а мультипликативной парадигмы защиты, а именно — уровень информационной безопасности в АС не выше уровня обеспечиваемого самым слабым звеном защиты.

Ниже приведены требования к реализации этапов защиты. В дальнейшем эти требования будут обсуждаться на качественном уровне.

1. ТРЕБОВАНИЯ К РЕАЛИЗАЦИИ ЭТАПОВ ЗАЩИТЫ

1.1. Идентификация/аутентификация пользователей

ИА должна выполняться аппаратно до этапа загрузки ОС. Базы данных ИА должны хранится в энергонезависимой памяти СЗИ, организованной так, чтобы доступ к ней средствами ПЭВМ был невозможен. Программное обеспечение контроллера должно храниться в памяти контроллера, защищенной от несанкционированных модификаций. Целостность ПО контроллера СЗИ должна обеспечиваться технологией его изготовления.

Идентификация должна осуществляться с применением отчуждаемого носителя информации.

Стойкость системы защиты связана с длиной пароля. При этом длина пароля может привести к трудностям при его запоминании. Для преодоления этих трудностей рекомендуется использовать фразы, облегчающие запоминание пароля.

Для генерации пароля следует применять аппаратный датчик случайных чисел.

1.2. Контроль целостности технического состава ПЭВМ и ЛВС

Контроль целостности технического состава ПЭВМ должен выполняться контроллером СЗИ до загрузки ОС. При этом должны контролироваться:

— центральный процессор,
— системный BIOS,
— дополнительный BIOS,
— вектора прерываний INT 13 и INT 40,
— CMOS, в том числе гибких дисков, жестких дисков и CD-ROM.

Контроль целостности технического состава ЛВС.

Целостность технического состава ЛВС должна обеспечиваться процедурой усиленной аутентификации сети. Процедура должна выполняться на этапе подключения проверенной ПЭВМ к сети и далее через заранее определенные администратором безопасности интервалы времени.

Усиленная аутентификация должна выполняться с применением рекомендованного варианта аппаратного датчика случайных чисел. Качество работы датчика должно контролироваться системой рекомендованных тестов.

1.3. Контроль целостности ОС

Контроль целостности системных областей и файлов ОС должен выполняться контроллером до загрузки ОС, чем обеспечивается механизм чтения реальных данных. Так как в электронном документообороте могут использоваться различные ОС, то встроенное в контроллер ПО должно обеспечивать разбор наиболее популярных файловых систем, а именно:

— FAT 12, FAT 16, FAT32 (Dos, Win 3x, Win 95/98),
— NTFS (Win NT),
— HPFS (OS/2),
— FreeBSd (Unix).

Целостность данного ПО должна гарантироваться технологией изготовления контроллеров СЗИ.

Защита ПО от несанкционированных модификаций должна обеспечиваться аппаратными средствами контроллера.

Для контроля целостности должна применяться известная (опубликованная) хэш-функция, эталонное значение которой должно храниться в энергонезависимой памяти контроллера, защищенной аппаратно от доступа из ПЭВМ.

1.4. Контроль целостности ППО и данных

Контроль целостности ППО и данных может выполняться как аппаратной компонентой, так и программной компонентой СЗИ в том слу-

чае, если ее целостность была зафиксирована аппаратно на предыдущем этапе. Для контроля целостности должна применяться известная(опубликованная) хэш-функция, эталонное значение которой должно аутентифицироваться с помощью отчуждаемого технического носителя информации (идентификатора).

1.5. Аутентификация документа при его создании

Для аутентификации документа при его создании должен аппаратно вырабатываться код аутентификации (КА). При этом до начала выработки КА должна быть обеспечена изолированность программной среды (ИПС) — см. пункты 1.1.—1.4. Запись копии электронного документа на внешние носители до выработки КА должна быть исключена. Если ЭлД порождается оператором, то КА должен вырабатываться с привязкой к оператору. Если ЭлД порождается программной компонентной АС, то КА должен вырабатываться с привязкой к данной программной компоненте.

1.6. Защита документа при его передаче

Защита документа при его передаче по внешним (открытым) каналам связи должна выполняться на основе применения сертифицированных криптографических средств, в том числе с использованием электронно-цифровой подписи (ЭЦП) для каждого передаваемого документа.

1.7. Аутентификация документа при обработке, хранении и исполнении документа

На этих этапах защита документа осуществляется применением двух КА — входного и выходного для каждого этапа. При этом КА должны вырабатываться аппаратно с привязкой КА к процедуре обработки. Для поступившего документа (с КА и ЭЦП) вырабатывается $КА_2$ и только затем снимается ЭЦП.

Далее: на следующем этапе (n) вырабатывается $КА_{n+1}$ и снимается $КА_{n-1}$. Таким образом, в любой момент времени документ защищен двумя КА - $КА_n$ и $КА_{n+1}$. КА должны вырабатываться и проверяться для документа, размещенного в оперативной памяти ЭВМ, в которой создана и поддерживается ИПС. Снятие $КА_{n-1}$ выполняется после установки $КА_{n+1}$.

1.8. Защита документа при доступе к нему из внешней среды

Принятие решения о доступе субъекта к объекту связано с характеристиками субъекта и объекта, но очень слабо связано с тем, откуда субъект запрашивает сведения об объекте и с помощью каких средств он это делает. В этой связи применение различного рода брандмауэров для этих целей бессмысленно. Достижение необходимого уровня безопасности возможно при реализации концепции функционально-распределенного межсетевого экрана с поддержкой семантического анализа данных на основе мандатного механизма.

2. ОСНОВЫ ЗАЩИТЫ ИНФОРМАЦИИ НА ПЭВМ

2.1. Обоснование создания изолированной программной среды

Пусть имеется компьютерная система (КС), в состав которой входит также некоторый набор исполняемых задач (программ) P1, P2,..., Pm (в том числе командные интерпретаторы, необходимый набор драйверов внешних устройств и сетевое ПО). Предположим, что в КС работают N субъектов-пользователей, каждый i-й из которых характеризуется некоторой персональной информацией Ki, не известной другим пользователям и хранящейся на некотором материальном носителе. Существует также выделенный субъект- администратор системы, который знает все Ki. Администратор КС присваивает i-му пользователю полномочия, заключающиеся в возможности исполнения им только заданного подмножества программ Ti={Pi1, Pi2, ... Pit}.

Положим, что в ПЗУ (BIOS) и операционной среде (в том числе и в сетевом ПО) отсутствуют специально интегрированные в них возможности НСД. Пусть пользователь работает с программой, в которой также исключено наличие каких-либо скрытых возможностей (проверенные программы). Потенциально злоумышленные действия могут быть такими:

1) проверенные программы будут использованы на другой ПЭВМ с другим BIOS и в этих условиях использоваться некорректно;

2) проверенные программы будут использованы в аналогичной, но не проверенной операционной среде, в которой они также могут использоваться некорректно;

3) проверенные программы используются на проверенной ПЭВМ и в проверенной операционной среде, но запускаются еще и не проверенные программы, потенциально несущие в себе возможности НСД.

Тогда, НСД в КС гарантировано невозможен, если выполняются следующие условия.

У1. На ПЭВМ с проверенным BIOS установлена проверенная операционная среда.

У2. Достоверно установлена неизменность DOS и BIOS для данного сеанса работы.

У3. Кроме проверенных программ в данной программно-аппаратной среде не запускалось и не запускается никаких иных программ, проверенные программы перед запуском контролируются на целостность.

У4. Исключен запуск проверенных программ в какой-либо иной ситуации, т.е. вне проверенной среды.

У5. Условия У1 — 4 выполняются в любой момент времени для всех пользователей, аутентифицированных защитным механизмом.

При выполнении перечисленных условий программная среда называется изолированной (далее будем использовать термин ИПС-изолированная программная среда).

Функционирование программ в ИПС существенно ослабляет требования к базовому ПО. В самом деле, ИПС контролирует активизацию процессов через операционную среду, контролирует целостность исполняемых модулей перед их запуском и разрешает инициирование процесса только при одновременном выполнении двух условий-принадлежности к разрешенным и неизменности. В таком случае от базового ПО требуется только:

1) невозможность запуска программ помимо контролируемых ИПС событий;

2) отсутствие в базовом ПО возможностей влиять на среду функционирования уже запущенных программ (фактически это требование невозможности редактирования оперативной памяти).

Все прочие действия, являющиеся нарушением Условий 1—3, в оставшейся их части будут выявляться и блокироваться. Таким образом, ИПС существенно снижает требования к ПО в части наличия скрытых возможностей.

Основным элементом поддержания изолированности среды является контроль целостности. При этом возникает проблема чтения реальных данных, так как контроль целостности всегда сопряжен с чтением данных (по секторам, по файлам и т.д.). В процессе чтения внедренное в систему разрушающее программное воздействие (РПВ) может навязывать вместо одного сектора другой или редактировать непосредственно буфер памяти. С другой стороны, даже контроль самого BIOS может происходить «под наблюдением» какой-либо дополнительной программы («теневой BIOS») и не показать его изменения. Аналогичные эффекты могут возникать и при обработке файла. Таким образом, внедренное в систему РПВ может влиять на процесс чтения-записи данных на уровне файлов или на уровне секторов и предъявлять системе контроля некоторые другие вместо реально существующих данных. Этот механизм неоднократно реализовывался в STELS-вирусах. Тем не менее, очевидно, что если программный модуль, обслуживающий процесс чтения данных, не содержал РПВ и целостность его зафиксирована, то при его последующей неизменности

чтение с использованием этого программного модуля будет чтением реальных данных.

Данный механизм можно реализовать с помощью следующего алгоритма (алгоритм ступенчатого контроля для создания ИПС) (цветная иллюстрация **Б**).

При включении питания ПЭВМ происходит тестирование ОП, инициализация таблицы прерываний и поиск расширений BIOS. При их наличии управление передается на них. После отработки расширений BIOS в память считывается первый сектор дискеты или винчестера (загрузчик) и управление передается на него, код загрузчика считывает драйверы DOS, далее выполняются файлы конфигурации, подгружается командный интерпретатор и выполняется файл автозапуска.

Таким образом, для реализации ИПС предварительно фиксируется неизменность программ в основном и расширенных BIOS, далее, используя функцию чтения в BIOS (для DOS int13h), читаются программы обслуживания чтения (драйверы DOS), рассматриваемые как последовательность секторов и фиксируется их целостность. Далее, используя уже файловые операции, читаются необходимые для контроля исполняемые модули (командный интерпретатор, драйверы дополнительных устройств, .EXE и .COM-модули и т.д.).

При запуске ИПС таким же образом и в той же последовательности выполняется контроль целостности.

В случае описанного механизма загрузки процесс аутентификации необходимо проводить в одном из расширений BIOS (чтобы минимизировать число ранее запущенных программ), а контроль запуска программ включать уже после загрузки DOS (иначе DOS определяет эту функцию на себя). При реализации ИПС на нее должна быть возложена функция контроля за запуском программ и контроля целостности.

Очевидно, что первый шаг алгоритма ступенчатого контроля для создания ИПС может быть основан на применении некоторой неизменяемой (немодифицируемой) процедуры, выполняющей роль точки опоры для следующих этапов.

Необходимость «твердой» точки опоры позволяет решить «основной вопрос» информационной безопасности — что первично «hard» или «soft»? (цветная иллюстрация **В**) Вывод можно сформулировать так — *создание и поддержка ИПС возможна только на основе применения специализированных аппаратных средств,* целостность которых обеспечивается технологией производства и периодическими проверками. Естественно, не любой контроллер в состоянии обеспечить все необходимые защитные функции — необходимо обеспечить хотя бы минимальный ресурс, с использованием которого можно обеспечить выполнение пошагового алгоритма контроля целостности и формирования ИПС. Эти вопросы будут затронуты ниже.

На основании проведенного анализа можно сформулировать требования к СЗИ, выполнение которых является обязательным для достиже-

ния высоких уровней защищенности АС (1В и выше).

Т1. Выполнение идентификации и аутентификации с гарантированной защитой от РПВ (например, как описано выше).

Т2. Контроль целостности программно-аппаратной среды АС.

Т3. Контроль чтения реальных данных.

Т4. Контроль доступа ко всем объектам файловой системы.

Т5. Контроль запуска задач.

Т6. Поддержание ИПС.

Эти требования не противоречат требованиям нормативных документов Гостехкомиссии России, а являются их необходимым дополнением, тем более, что изложенные в [1,2] требования характеризуют каждый класс только минимальной совокупностью требований по защите.

Для того, чтобы определить пригодность тех или иных СЗИ для применения в составе АС как средства обеспечения информационной безопасности, можно было бы рассматривать не только реализацию требований [1,2], но и выполнение требований Т1—Т6.

2.2. Механизмы создания ИПС

2.2.1. Выполнение идентификации и аутентификации с гарантированной защитой от РПВ (Т1)

В соответствии с нормативными документами, комплекс средств защиты (КСЗ) должен требовать от пользователей идентифицировать себя при запросах на доступ. КСЗ должен подвергать проверке подлинность идентификатора субъекта — осуществлять аутентификацию. КСЗ должен располагать необходимыми данными для идентификации и аутентификации. КСЗ должен препятствовать входу в СВТ и доступу к защищаемым ресурсам неидентифицированного пользователя или пользователя, чья подлинность при аутентификации не подтвердилась. КСЗ должен обладать способностью надежно связывать полученные результаты идентификации со всеми действиями данного пользователя.

Обсуждение

Мы считаем, что реализация этих требований связана с выполнением условий создания изолированной программной среды, в том числе:

1) идентификация должна выполняться труднокопируемым уникальным идентификатором до загрузки операционной системы;

2) аутентификация должна выполняться с обеспечением защиты от раскрытия пароля—по крайней мере, пароль должен быть достаточной длины и проверяться он должен также до загрузки операционной системы (ОС);

3) должен обеспечиваться контроль целостности программ и данных и на этой основе защита от несанкционированных модификаций программ и данных (с осуществлением основных контрольных функций до ОС);

4) в составе средств защиты от НСД должны быть средства, позволяющие обеспечить контроль запуска задач и на этой основе функциональное замыкание информационных систем (создание и поддержание изолированной программной среды) с исключением возможности несанкционированного выхода в ОС.

Особенностью (и, несомненно, преимуществом) комплекса «Аккорд» является проведение процедур идентификации и аутентификации до загрузки операционной системы.

Это обеспечивается при помощи ПЗУ, установленного на плате контроллера «Аккорд». Это ПЗУ получает управление во время так называемой процедуры ROM-Scan.

Суть данной процедуры в следующем.

В процессе начального старта после проверки основного оборудования BIOS компьютера начинает поиск внешних ПЗУ в диапазоне от С 800:0000 до Е000:0000 с шагом в 2К. Признаком наличия ПЗУ является наличие слова АА55Н в первом слове проверяемого интервала. Если данный признак обнаружен, то в следующем байте содержится длина ПЗУ в страницах по 512 байт. Затем вычисляется контрольная сумма всего ПЗУ, и если она корректна — будет произведен вызов процедуры, расположенной в ПЗУ со смещением 3. Такая процедура обычно используется для инициализации.

В данном случае в этой процедуре проводится идентификация и аутентификация пользователя, и при ошибке возврат из процедуры не происходит, т.е. загрузка выполняться не будет.

Стойкость процедур идентификации и аутентификации определяется и длиной пароля.

Оценим требуемую длину пароля, используемого при аутентификации. Эта оценка важна для того, чтобы правильно выбрать период смены паролей из предположения, что идентификатор пользователя может быть утрачен, а пользователь по тем или иным причинам не поставит об этом в известность администратора безопасности информации.

Пусть требуемая вероятность подбора пароля в результате трехмесячного регулярного тестирования должна быть не выше 0,001.

По формуле Андерсона [3]

$$4{,}32 \times 10^{4} \times k\,(M/P) \leqslant A^{S},$$

где

k — количество попыток в мин,

M — период времени тестирования в месяцах,

P — вероятность,

A — число символов в алфавите,
S — длина пароля.

Время на одну попытку при использовании комплекса «Аккорд» — не менее 7 сек., т.е.

$$k = 60/7 = 8{,}57$$

Для английского алфавита $(A = 26)$ и $S = 7$

$$1{,}11 \times 10^9 \leqslant 8{,}03 \times 10^9$$

т.е. пароля длиной 7 символов достаточно.
Для $S = 6$

$$3{,}7 \times 10^8 \times M \leqslant 3{,}089 \times 10^8,$$

или

$$M \leqslant 0{,}83$$

т.е. при длине пароля 6 символов и регулярном тестировании в течении 25 дней вероятность подбора пароля составит не более 0,001.

Отметим, что в СЗИ «Аккорд» используются и некоторые дополнительные механизмы защиты от НСД к компьютеру. Так, в частности, для пользователя администратор БИ может установить:

— время жизни пароля и его минимальную длину — например, исходя из расчетов, приведенных выше;

— временные ограничения — время по дням недели (с дискретностью 30 мин), в которое разрешено начало работы для данного субъекта;

— параметры управления экраном — гашение экрана через заранее определенный интервал времени (в случае, если в течение указанного интервала действия оператором не выполнялись), подача соответствующих звуковых и визуальных сигналов.

2.2.2. Контроль целостности программно-аппаратной среды АС (Т2)

В СВТ должны быть предусмотрены средства периодического контроля за целостностью программной и информационной части КСЗ.

Программы КСЗ должны выполняться в отдельной части оперативной памяти.

Обсуждение

Обычно для контроля целостности используется вычисление контрольной суммы. Вычисленное значение сравнивается с эталонным, и результат сравнения позволяет судить о состоянии программ и данных - остались ли они неизменными, были ли несанкционированные модификации.

Казалось бы, — все нормально, но как убедиться в целостности процедур контроля? Ведь можно модифицировать и процедуру расчета контрольных сумм, и процедуру контроля процедуры расчета контрольных сумм и т.д. Кажется, что выхода из этого круга нет,—невозможно обеспечить целостность программных средств за счет других программных средств, как невозможно вытащить самого себя за волосы из болота (но попытки были, а в области информационной безопасности они продолжаются и до сих пор, — что позволяет данную некорректность в проведении контроля целостности квалифицировать как «*синдром Мюнхгаузена*») (цветная иллюстрация **Г**).С этой неясностью можно покончить только одним способом — вынести процедуры контроля за пределы компьютера, т.е. выполнять их аппаратно (или с использованием аппаратных средств) — иными словами, опять-таки нужна некоторая «кочка», что-то твердое, на что можно опереться, точка опоры — этот принцип обозначим как *«принцип Архимеда»* (цветная иллюстрация **Д**).

Еще одна проблема — это качество самих процедур контроля. Рассмотрим следующий случай — для контроля применяется контрольная сумма, вычисляемая по алгоритму CRC (именно так обычно и происходит). Для наглядности предположим, что используется CRC-3 (реально, естественно, используется от CRC-8 до CRC-32, но усложнение при этом носит чисто технический характер). Для CRC-3 количество значений—8 (от 0 до 7). Допустим, контрольная сумма исходного (правильного) набора документов C1=3. К этому пакету злоумышленник добавляет еще несколько, таких, что их контрольная сумма C2=2. Таким образом, контрольная сумма модифицированного пакета документов Cm=(C1 + C2)mod8 = 5. Теперь для того, чтобы эта модификация не была обнаружена, злоумышленник вполне может к уже модифицированному пакету добавить еще несколько документов, таких, что их контрольная сумма C3=6. Контрольная сумма модифицированного пакета становится Cm=(C1 + C2 + C3)mod8 = (3 + 2 + 6)mod8 = 3, т.е. Cm = C1, и модификация не будет замечена контрольными процедурами. В общем случае, достаточно подобрать такой пакет, для которого (C2 + C3)mod8 = 0, и информационная атака может быть успешной. Общий источник возможности таких атак состоит в том, что значения применяемых алгоритмов вычисления контрольных сумм распределены равномерно (или просто известным образом). Соответственно, хорошей защитой может служить применение алгоритмов с существенно нелинейным распределением значений. В частности, этими свойствами обычно обладают хэш-функции.

Контроль целостности в СЗИ «Аккорд» выполняется следующим образом.

Целостность самой контрольной процедуры обеспечивается тем, что хранится она не на диске, а в ПЗУ контроллера «Аккорд», будучи тем самым защищена от модификаций. При нормальном осуществлении процедур защиты от НСД (идентификации и аутентификации) копия контрольных процедур помещается в специально отведенную область ОЗУ и затем запускается на исполнение.

Администратор БИ для каждого субъекта-пользователя системы определяет опции контроля, перечень файлов, целостность которых должна контролироваться системой.

Дополнительно в СЗИ «Аккорд» предусмотрен динамический контроль целостности собственно монитора разграничения доступа. Этот контроль выполняется периодически и обеспечивает дополнительный уровень защиты от случайных или преднамеренных покушений на отключение СЗИ. Как и на других этапах контроля целостности, здесь применяется контроль с использованием хэш-функции.

Выбор идентификатора

Мы уже говорили о важности корректного выполнения процедур контроля целостности, но при этом в тени остался еще один вопрос — а где, собственно, нужно хранить эталонное значение контрольной суммы, которое и будет сравниваться с вычисленным в текущем сеансе работы? Вариантов всего два — или в долговременной памяти (на диске) компьютера, или вне его, на некотором отчуждаемом носителе. Очевидно, что уровень безопасности выше, если эталонное значение контрольной суммы отчужденно от компьютера — *«отчуждай и властвуй»* (цветная иллюстрация **Е**). Значит, должен быть носитель, на котором это значение должно быть записано. Этот факт является решающим в споре о типе идентификатора — использовать ли для идентификации технические устройства или биометрические параметры. Действительно, отпечатки пальцев и рисунок на сетчатке глаза уникальны, но — и на палец, и на глаз трудно записать значение контрольной суммы для контроля целостности. Следовательно, остается использовать для этих целей технические устройства.

Изучая вопрос носителя информации, еще в 1991г. мы пришли к выводу, что хорошим выбором является touch-memory компании «Dallas semiconductor». Первое — это высокая надежность, вывести touch-memory из строя достаточно трудно. Приемлемы и массо-габаритные характеристики — таблетка диаметром 16 мм и толщиной 3-5 мм, очень подходит для таких применений. Уникальность — каждая таблетка обладает уникальным номером, который формируется технологически и подделать который невозможно. Вполне приемлемой при использовании touch-memory является и цена. Это устройство может быть отнесено к категории труднокопируемых устройств, а с учетом того, что также может быть использован уникальный номер каждой таблетки, можно говорить, что это устройство некопируемое. Все эти качества и определяют выбор touch-memory как наиболее удобного носителя информации. В качестве альтернативного варианта могут применяться smart-карты. Этот подход реализован в версии «Аккорд ПК».

Подводя итог, можно отметить, что подавляющее число преступных воздействий на информационные системы совершаются с помощью несанкционированных модификаций программ и данных — с помощью программ-закладок, вирусов, криптовирусов и др. Общим элементом при этих воздействиях является этап, на котором разрушающее программное воздействие (вредоносная программа) внедряется в информационную систему. Стало быть, система защиты должна обеспечивать блокировку этой возможности.

2.2.3. Контроль чтения реальных данных (ТЗ)

Как уже отмечалось, для адекватного контроля чтения реальных данных необходимо обеспечить целостность системных областей и системных файлов. К ним относятся:

— MBR drive C,
— TM hardware base adress 0340,
— RAM size,
— DOS bootrecord,
— MSDOS.SYS,
— IO.SYS,
— AUTOEXEC.BAT,
— CONFIG.SYS,
— COMMAND.COM,

и для WIN 95:

— MSDOC.W40,
— WINBOOT.SYS,
— AUTOEXEC.W40,
— CONFIG.W40,
— COMMAND.W40.

Заметим, что некоторые из этих объектов являются условно-постоянными (например, MBR), некоторые условно-переменными (например, autoexec и config). Условно-переменные объекты на ПЭВМ, входящих в состав АС, могут изменяться — так, autoexec и config настраиваются администратором безопасности информации. Изменение этих файлов приводит к изменению эталонного значения хэш-функции, используемого при контроле целостности. Если это значение будет хранится на идентификаторе пользователя, то периодически будет возникать ситуация, при которой администратор безопасности информации будет вынужден собирать всех пользователей, работающих на некоторой ПЭВМ, и записывать на их идентификаторы новое эталонное значение хэш-функции для контроля целостности системных областей и системных файлов. Очевидно, что существуют организационные проблемы для реализации такого подхода. Отсюда следует, что значение хэш-функции системных областей целесообразно хранить не в идентификаторе пользователя, а в специальной энергонезависимой памяти контроллера, максимально защищенной от несакционированного доступа со стороны потенциальных злоумышленников.

2.2.4. Контроль доступа ко всем объектам файловой системы (Т4)

Требование реализации дискреционного механизма разграничения доступа.

КСЗ должен контролировать доступ наименованных субъектов (пользователей) к наименованным объектам (файлам, программам, томам и т.д.).

Контроль доступа должен быть применим к каждому объекту и каждому субъекту (индивиду или группе равноправных индивидов).

Для каждой пары (субъект—объект) в СВТ должно быть задано явное и недвусмысленное перечисление допустимых типов доступа (читать, писать и т.д.), т.е. тех типов доступа, которые являются санкционированными для данного субъекта (индивида или группы индивидов) к данному ресурсу СВТ (объекту).

КСЗ должен содержать механизм, претворяющий в жизнь дискреционные правила разрешения доступа (ПРД), как для явных действий пользователя, так и для скрытых. Под «явными» здесь подразумеваются действия, осуществляемые с использованием системных средств — системных макрокоманд, инструкций языков высокого уровня и т.д., а под «скрытыми» — иные действия, в том числе с использованием собственных программ работы с устройствами.

Механизм, реализующий дискреционный принцип контроля доступа, должен предусматривать возможности санкционированного изменения ПРД, в том числе возможность санкционированного изменения списка пользователей СВТ и списка защищаемых объектов. Права изменять ПРД должны предоставляться выделенным субъектам (администрации, службе безопасности и т.д.). Должны быть предусмотрены средства управления, ограничивающие распространение прав на доступ.

Обсуждение

Для реализации дискреционного механизма разграничения доступа необходимо по крайней мере конкретизировать термины, используемые в формальном описании. При этом целесообразно исходить из того, что полученная модель должна, с одной стороны, быть понятна пользователю, с другой—не ограничивать пользователя в реализации процедур разграничения доступа и как можно ближе соответствовать особенностям архитектуры технических средств компьютера и особенностям операционной системы. С этой точки зрения необходимо определить, что целесообразно выбрать в качестве объектов разграничения доступа, и какие допустимые типы доступа целесообразно использовать.

Обсуждая этот вопрос, отметим, что в качестве объектов в ОС используются:

— диски;
— каталоги;
— файлы (задачи).

Конечно, использование файлов (задач) в качестве основного типа объектов разграничения доступа также нецелесообразно, т.к. в этом случае объем и сложность описаний ПРД были бы неоправданно увеличены.

Нужно отметить, что выбор базового уровня объектов разграничения доступа зависит от уровня развития средств вычислительной техники и политики информационной безопасности, принятой в организации. Даже из общих соображений ясно, что количество объектов, которыми может управлять оператор, конечно. С ростом количества объектов существенно возрастает трудоемкость управления защитой — в частности, резко растут описания правил разграничения доступа, усложняются процедуры анализа корректности этих описаний и т.д. Рано или поздно целесообразным становится переход на более высокий уровень иерархии объектов — от записей к файлам, от файлов к каталогам, от каталогов к логическим дискам.

Переход к очередному (более высокому) уровню иерархии снижает эффективность использования внешней памяти, но так же значительно снижает сложность управления защитой. В настоящее время снижение эффективности использования внешней памяти вряд ли можно считать критичным, так как объемы накопителей резко возросли, а их цены резко снизились. Стандартный на сегодня накопитель 10 Gb можно разбить на 5—10 логических дисков, и управлять безопасностью, определяя возможность доступа пользователя к тому или иному логическому диску.

Тем не менее, основным объектом иерархии в ОС являются каталоги, и, таким образом, естественным выбором в качестве основного объекта доступа является выбор каталогов.

Выбор типов доступа целесообразно связать с функциями DOS, посредством которых осуществляется доступ к ресурсам. Перехват вызовов этих функций позволит реализовать ПРД для явных действий пользователя.

Реализация ПРД для скрытых действий пользователя может быть осуществлена за счет ограничения перечня задач, которые пользователь имеет право запускать. Это означает, что средства ПРД должны содержать возможность явного и недвусмысленного описания перечня задач, запуск которых разрешен пользователю, и средства контроля за использованием этих задач. Формирование перечня должно осуществляться администратором БИ в порядке, предусмотренном для формирования и изменения ПРД.

Разграничение доступа в СЗИ «Аккорд» (разграничение ресурсов DOS) реализовано при помощи резидентной программы, которая перехватывает на себя обработку функций DOS (в основном это прерывание int 21h, а также int 25/26, и int 13).

Смысл работы данного резидентного модуля в том, что при получении от пользовательской программы запроса, например, на удаление файла — вначале производится проверка наличия таких полномочий у пользователя. Если такие полномочия есть — управление передается обычному обработчику DOS для исполнения операции. Если таких полномочий нет—имитируется выход с ошибкой.

В обычных системах разграничения доступа в качестве атрибутов используется триада(*R, W, X*), где

R – разрешение на чтение объекта (диска, каталога, файла),

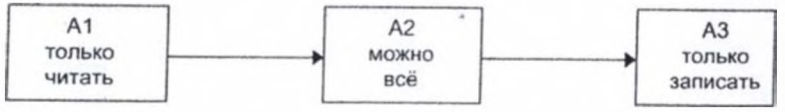
W – разрешение на модернизацию объекта,

X – разрешение на запуск задач.

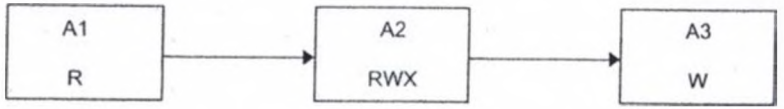
Достаточно ли этих атрибутов, чтобы на их основе реализовать «любую разумную непротиворечивую политику безопасности» предприятия (организации)?

Очевидно, что нет. Действительно, рассмотрим обычную процедуру конфиденциального делопроизводства. Движение документов может в этом случае быть таким:

Исходный документ (файл) может быть только прочитан пользователем из каталога *A1* и переписан в каталог *A2*. В каталоге *A2* документ обрабатывается по произвольной технологии, а после завершения записывается в каталог *A3*, после чего документ уже не может быть модифицирован (и даже прочитан) этим пользователем.



С использованием триады(*R,W,X*) права доступа могут быть назначены следующим образом.



Обратим внимание — при этом итоговый документ, записанный в каталог *A3*, может быть модифицирован оператором, так как атрибут <*W*> разрешает не только запись, но и модификацию (переименование, удаление, изменение) объекта.

Даже этот пример показывает, что трех атрибутов недостаточно для описания политики безопасности организации.

В СЗИ «Аккорд» используется полная система атрибутов, позволяющая реализовать любую разумную политику безопасности.

Для описания взаимодействия <субъект-объект> в СЗИ «Аккорд» применяются следующие атрибуты:

R – открытие файлов для чтения;

W – открытие файлов для записи;

O – подмена атрибута *R* атрибутами *RW* на этапе открытия файла;

C – создание файлов;

D – удаление файлов;

N – переименование файлов и подкаталогов;

V – видимость файлов;

M – создание подкаталогов;

E – удаление подкаталогов;

G – доступность данного каталога (т.е. переход к нему);

X – исполнение задач;

S – наследование подкаталогами атрибутов каталога;

В СЗИ «Аккорд» поддерживается также два списка файлов, применение которых повышает адаптивность системы. Это список «Файлы» («белый» список) и список «Скрытые файлы» («черный» список).

Права доступа к отдельным файлам описываются в разделе «Файлы» — эти права будут обеспечиваться в безусловном порядке, даже если файл расположен в каталоге, доступа к которому данный пользователь не имеет. Предусмотрено определение следующих прав:

— открытие файлов для записи;

— открытие файлов для чтения;

— создание файлов;

— удаление файлов;

— переименование файлов;

— видимость файлов.

Предусмотрена также возможность установки разрешения на исполнение задач, размещенных в данном файле.

Существует также и «черный список» (в терминах, принятых в описании СЗИ «Аккорд» — «Скрытые файлы»). Файлы, описанные в «черном списке», становятся недоступными пользователю, даже если они расположены в каталогах, к которым пользователь имеет доступ. В «черный список» можно включать также логические имена устройств и драйверы устройств. Эти объекты после такого описания становятся недоступны пользователю. Таким образом осуществляется сопоставление пользователя и доступных ему устройств.

С помощью описанных механизмов назначаются ПРД для каждого пользователя. Для удобства описания сложных ПРД в систему введен некоторый «суперпользователь» — SYSTEM. Все параметры определяемые данному пользователю, распространяются и на других пользователей ПЭВМ.

Еще одним фактором является дисциплина взаимодействия атрибутов. В обычных системах по отношению к иерархическим элементам файловой системы (логический диск-каталог-файл) действует правило, по которому право доступа к объектам нижнего уровня определяются минимальными правами доступа ко всем объектам вышележащих уровней. В СЗИ «Аккорд» такое правило действует на уровне «логический диск—остальные объекты», но на всех других уровнях (катало—подкаталоги—файлы) атрибуты независимы, если не используется специально введенный атрибут наследования «S» — наследование подкаталогами атрибутов каталога.

Очевидно, что задача конфиденциального делопроизводства легко решается в системе атрибутов СЗИ «Аккорд».

3. АУДИТ

Регистрация действий пользователей на рабочих станциях является одной из важнейших функций, реализованных в программно-аппаратном комплексе средств защиты информации от несанкционированного доступа «Аккорд. Благодаря мощной системе атрибутов администратор безопасности информации (АБИ) может очень чётко отслеживать все действия пользователей не только на рабочих станциях, но и все запросы пользователей к ресурсам любого файлового сервера (к примеру, файлового сервера Novell NetWare). При этом регистрация событий выполняется в локальном журнале на рабочей станции. Для адекватного представления о работе пользователя АБИ достаточно просмотреть этот журнал. Регистрация запросов пользователей к сетевым ресурсам не создаёт никаких дополнительных затрат для файлового сервера. Однако, при всех положительных чертах ведения журналов на рабочих станциях существует один существенный недостаток: все попытки несанкционированного доступа (НСД) можно отследить только на основании отложенного (off-line) анализа (АБИ собирает и анализирует все журналы с некоторой периодичностью, к примеру, раз в день), что позволяет выявить НСД, но не позволяет его предотвратить. Именно поэтому АБИ нуждается в инструменте оперативного (on-line) наблюдения и управления работой пользователей на рабочих станциях.

Основной функцией наблюдения следует признать оперативный анализ журналов регистрации событий на каждой конкретной станции. АБИ должен получать оперативную информацию обо всех попытках НСД. Кроме того, часто вызывает интерес сам факт доступа к некоторому ресурсу, расположенному как на рабочей станции, так и на файловом сервере. Для более детального анализа деятельности пользователя необходимо предоставить возможность получения АБИ локальных журналов с рабочих станций.

Во время работы с журналами для предотвращения попыток НСД АБИ должен иметь возможность блокировать средства ввода и вывода рабочей станции. И в качестве последней меры воздействия возможность удалённой перезагрузки рабочей станции.

Все вышеперечисленные функции реализованы в программном продукте «Автоматизированное рабочее место администратора безопасности информации» (АРМ АБИ) в СЗИ «Аккорд». Для получения более подробной информации у АБИ есть дополнительная возможность просмотра экрана рабочей станции. Кроме того, АБИ может работать в режиме эмуляции терминала. Особенностью оперативного анализа журналов является

мощная система фильтров в сочетании с возможностью вывода событий выделенной станции в отдельное окно наблюдения.

В качестве протоколов передачи данных АРМ АБИ может использовать как ipx, так и netbios, обеспечивая тем самым аппаратную независимость от типа используемых сетевых карт и возможность работы в любой ЛВС. Корректная работа АРМ АБИ возможна как в ЛВС с выделенным сервером (к примеру, Novell NetWare), так и в одноранговой сети типа Windows. Более того, для оперативного наблюдения и управления работой пользователей АБИ достаточно обеспечить полноценное функционирование протоколов ipx или netbios на каждой рабочей станции без подключения к серверам.

Так как действия, выполняемые при удалённом управлении и наблюдении, носят критичный характер, моменты установления соединения между рабочей станцией и АРМ АБИ проводятся с использованием протокола усиленной аутентификации, что гарантирует подлинность как рабочей станции, так и АРМ АБИ. Поэтому, в качестве сервисных функций АРМ АБИ добавлены функции усиленной аутентификации. Кроме того, все команды, посылаемые на рабочие станции, снабжаются кодом аутентификации, что гарантирует корректность данной команды.

4. УСИЛЕННАЯ АУТЕНТИФИКАЦИЯ

Существующая в ЛВС Novell NetWare система идентификации и аутентификации пользователей ориентирована в основном на подтверждение подлинности пользователей в момент запроса доступа к ресурсам файлового сервера. В качестве базовой системы аутентификации используется схема с простым паролем. В то же время подлинности рабочих станций сети уделяется недостаточное внимание. Единственная проверка заключается в сравнении номера сетевой карты рабочей станции, с которой идёт запрос доступа, со списком разрешённых сетевых карт для данного пользователя. Все проверки, осуществляемые при диалоге пользователя и файлового сервера, решают одну задачу— проверку подлинности пользователя.

Проверка номера сетевой карты при запросе пользователем доступа к ресурсам файлового сервера не является достаточно сильным средством пресечения попыток несанкционированного доступа. Внеся дополнительную строку в файл настройки сетевого программного обеспечения рабочей станции, пользователь может задать любой номер сетевой карты и легко получить доступ к ресурсам файлового сервера с любой рабочей станции ЛВС, при этом, с точки зрения сервера, данное соединение будет корректным. Для предотвращения описанной ситуации необходимо ограничивать доступ пользователей к файлам конфигурации сетевого программного обеспечения или контролировать их целостность. Для рабочих станций, работающих под управлением ка-

ких-либо средств защиты от НСД, недопустимо изменение номеров сетевых карт. Неконтролируемое изменение данных параметров может разрушить любой план защиты.

Другим путём для получения несанкционированного доступа к ресурсам файлового сервера является подключение посторонней станции к сети. Принцип работы ЛВС EtherNet таков, что злоумышленник может незаметно подключить свою ЭВМ к кабелям ЛВС и работать как легальная станция.

При этом (в случае дискредитации пароля пользователя) нет никаких преград для несанкционированного доступа к ресурсам файлового сервера с любой рабочей станции в ЛВС. Источник проблемы заключается в том, что существующие подсистемы аутентификации ориентированны на установление подлинности пользователя-субъекта, а в ЛВС пользователем является вовсе не оператор, а оператор вместе со всей рабочей станцией. Критичным с точки зрения НСД является как подмена оператора, так и подмена рабочей станции. Следовательно, необходимо применять подсистему усиленной аутентификации, осуществляющую проверку подлинности файловых серверов и рабочих станций после того, как пользователь признан корректным с точки зрения Novell NetWare. Это взаимодействие всякий раз должно содержать уникальные данные,т.к., прослушав один раз такой диалог, злоумышленник может полностью воспроизвести его и в результате получить доступ к ресурсам файлового сервера. В каждом сеансе работы системы усиленной аутентификации необходимо использовать случайные данные, которые должны при этом ещё и обеспечивать проверку подлинности обеих сторон. Идеальным решением данной проблемы было бы применение механизма подтверждения подлинности, основанного на коде аутентификации (КА), функционально аналогичного электронной цифровой подписи (ЭЦП). Однако в этом случае необходимо решить проблему хранения секретных ключей станций.

Предлагаемая система усиленной аутентификации предоставляет дополнительный механизм проверки подлинности рабочих станций в момент запроса доступа к ресурсам файлового сервера с учётом всех перечисленных требований. Секретный ключ станции храниться в закодированном виде, причём кодируется он на секретном ключе пользователя, который, в свою очередь, хранится вне ЭВМ в Touch Memory пользователя. Даже в случае полного доступа к рабочей станции у злоумышленника нет никакой возможности получить доступ к секретному ключу станции. Доступ к секретному ключу файлового сервера также затруднён для пользователей, так как он хранится не на жёстком диске, а во внутренней памяти платы «Аккорд». Уникальность данных для каждого диалога обеспечивается использованием аппаратного датчика случайных чисел (ДСЧ) контроллера «Аккорд».

Механизм усиленной аутентификации в этом случае выглядит примерно так:

— на сервере АС выбирается случайное число с использованием ДСЧ «Аккорд»;

— это случайное число подписывается подписью сервера и рассылается по всем станциям;

—каждая станция проверяет подпись сервера. Если подлинность подписи не подтверждается—станция посылает сообщение на АРМ АБИ и отключается, считая север подмененным;

— при подтверждении подлинности подписи каждая станция подписывает пакет своей подписью и направляет его на сервер;

— сервер анализирует полученные от станций пакеты аутентификации и в случае, если подлинность подписи станции не подтверждена, отключает её и передает сообщение на АРМ АБИ.

Обмен информацией при сеансе усиленной аутентификации показан на рис. 2.1.

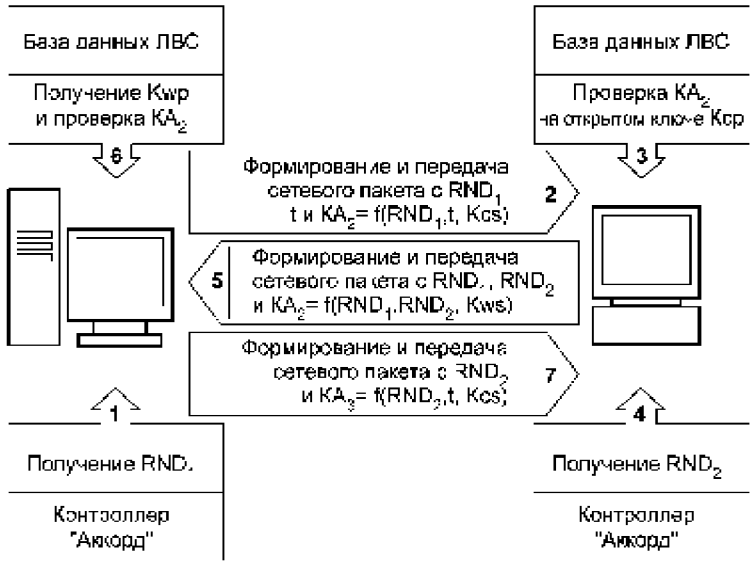


Рисунок 2.1

Естественно, применять ЭЦП как таковую для целей усиленной аутентификации необходимости нет. Этот термин был выбран для простоты пояснений. Реально используется механизм, близкий к механизму КА.

Система усиленной аутентификации не требует от пользователя существенных затрат времени и сил, являясь в тоже время мощным инструментом проверки подлинности рабочих станций ЛВС. Усиленная аутентификация в гетерогенной сети иллюстрируется рис. 2.2.

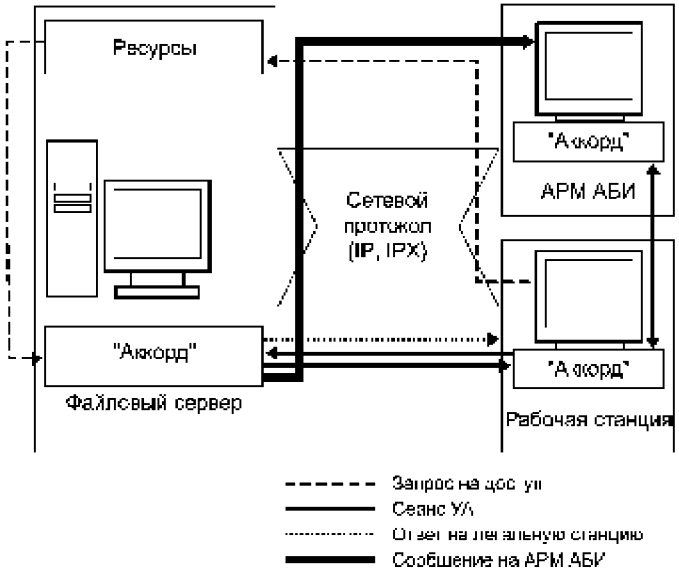


Рисунок 2.2

5. АУТЕНТИФИКАЦИЯ ДОКУМЕНТОВ

В процессе обработки, хранения, передачи ЭлД в АС может быть искажен как в результате действий злоумышленника, так и в результате активизации РПВ, да и просто ошибок в программах. Это означает, что аутентификация ЭлД должна осуществляться на каждом этапе обработки.

Применение для указанных целей механизма ЭЦП по ГОСТ 34.10-11.94 неэффективно, так как сопряжено с добавлением к каждому ЭлД блоков контрольной информации в 64 байта, что создает порядка 150% накладных расходов при передаче и хранении информации, не считая затрат на время обработки контрольной информации. Данное обстоятельство является вынужденным следствием заложенного в ГОСТ 34.10-11.94 асимметричного алгоритма ЭЦП, стойкость которого непосредственно связана с длиной проверочного блока данных. В то же время известны другие (симметричные) алгоритмы ЭЦП, в которых длина проверочного блока данных может быть существенно уменьшена, а скорость выработки и проверки ЭЦП — существенно увеличена. Недостаток симметричного алгоритма состоит в том, что секретный элемент (ключ), на сохранении в тайне которого держится стойкость ЭЦП, находится в ведении как удостоверяющей, так и проверяющей стороны, что не позволяет применять данный алгоритм при удостоверении ЭлД с возможностью последующего арбитражного разбора спорных случаев в связи с отказом от передачи или приема данного ЭлД. Однако, в том случае, если один из экземпляров секретного ключа, на сохранении в тайне которого держится стойкость ЭЦП, хранится в недоступной оператору ПЭВМ аппаратурной части системы защиты, то данный, симметричный по-существу вариант алгоритма ЭЦП, превращается в асимметричный вариант ЭЦП. Более того, если оба экземпляра ключа, на сохранении в тайне которого держится стойкость ЭЦП, хранятся в недоступной операторам ПЭВМ аппаратурной части системы защиты, то в этом случае о создаваемой системе контроля целостности и подтверждения достоверности ЭлД можно говорить уже не как о системе электронной цифровой подписи, а как о системе раннего обнаружения несанкционированного вмешательства в электронный документооборот с последующим разбирательством административными (неарбитражными) методами. Данная система относится к внутриобъектовой технологии, она не отменяет, а дополняет применяемые системы ЭЦП для пакетов документов.

Код аутентификации (КА) электронных документов предлагается вырабатывать в виде хэш-функции от контролируемой информации, индивидуального идентификационного кода операции, и некоторых служебных данных. Такой выбор базируется на фундаментальном математическом свойстве хэш-функции, которое может быть сформулировано следующим образом.

Пусть X и Y — некоторые алфавиты(конечные множества элементов), $X(n)=\{(x(1), x(2), \ldots, x(n))\}$, $Y(m)=\{(y(1), y(2), \ldots, y(m))\}$ — множества всех последовательностей (сообщений) в указанных алфавитах длиной n и m, соответственно, n и m — целые числа.

Заданное однозначное отображение $h(*):X(n) => Y(m)$, ставящее в соответствие каждой последовательности из $X(n)$ некоторую последовательность из $Y(m)$, называют хэш-функцией, если для любой последовательности $\underline{x} = (x(1), x(2), \ldots, x(n))$ практически невозможно найти отличную от нее последовательность $\underline{x}' = (x'(1), x'(2), \ldots, x'(n))$, такую, что

$$h(\underline{x}) = h(\underline{x}').$$

Под практической невозможностью решить предыдущее уравнение относительно $\underline{x}'$ понимается значительная вычислительная сложность этой задачи, в силу которой получить хотя бы одно ее решение можно только за очень длительное время, перебирая значения $\underline{x}'$ или производя какие-то другие вычисления.

Функция $h(*)$, описанная в стандарте ГОСТ Р 34.11-94, этому требованию удовлетворяет, как, впрочем, и функция, используемая в СЗИ «Аккорд».

Таким образом, если электронный документ (ЭлД) снабжен проверочным блоком в виде значения его хэш-функции, то у злоумышленника (РПВ) не оказывается возможности изменить документ каким-либо образом с сохранением проверочного блока. Однако, если не предпринять дополнительных мер, он может создать нужный ему документ, вычислить для него проверочный блок (способ вычисления хэш-функции считается известным) и сформировать таким образом ложный электронный документ.

Для предотвращения такой угрозы можно снабдить ЭлД проверочным блоком вида

$$h(\text{ЭлД}, \text{КОД}),$$

где КОД — конфиденциальный код, сохраняемый в тайне и добавляемый к ЭлД при хэшировании (код хэширования). Поскольку КОД не доступен злоумышленнику, он не сможет сформировать ложный электронный документ вместе с его проверочным блоком. Однако, учитывая архитектуру взаимодействия между ПЭВМ в ЛВС, с одной стороны, и сервером безопасности (сервером кода аутентификации), с другой стороны, более рационально использовать несколько иной вид проверочного блока, а именно

$$h\,(h(\text{ЭлД}), \text{КОД}\,).$$

Такой выбор проверочного блока позволяет снизить нагрузку в канале передачи данных на сервер безопасности, путем вычисления h(ЭлД) на ПЭВМ АС, а контрольного значения КА — на сервере безопасности, который получает не ЭлД целиком, а его свертку размером 32 байта. Отметим одновременно, что защите с помощью проверочного блока может подвергаться не весь электронный документ, а его отдельные части (поля).

В связи с вышеизложенным код аутентификации ЭлД предлагается формировать отбором достаточного количества байт из $h(h(\text{ЭлД}),\text{КОД})$. Достаточность длины проверочного блока данных определяется вероятностью неповторения проверочных блоков в течение времени действия используемых значений КОДов.

В схеме равновероятной и независимой выборки k проверочных блоков их бесповторное появление будет иметь вероятность

$$P = (1 - k^2/2N),$$

где N — количество различных значений проверочных блоков.
Если m — длина проверочного блока в байтах, то всего можно выработать

$$N = 2^{8m}$$

различных значений проверочных блоков.

Пусть ежедневно в АС обрабатываются 10000 ЭлД, причем для каждого ЭлД в процессе обработки необходимо выработать 10 КА. Если КОДы действуют в течении года, то

$$k = 365 \times 10^6 \sim 2^{28}.$$

Пусть $m = 12$, т.е. $N = 2^{8\times12}$. Тогда

$$1 - P = k^2/2N \sim 8\times10^{-9}.$$

Отсюда очевидно, что для формирования КА достаточно использовать 12 байт из $h(h(\text{ЭлД}), \text{КОД})$.

Используемые на серверах безопасности коды хэширования предлагается хранить в виде таблиц (далее будет использоваться термин «таблица достоверности», сокращенно ТД) в логически недоступной с прикладного уровня программ памяти сопроцессора «Аккорд СБ» и таким образом, чтобы в таблицах, используемых серверами безопасности, содержались только необходимые для их работы значения КОД.

Логическая недоступность ТД как легальным операторам, так и злоумышленникам предотвращает возможность создавать и проверять КА, не обращаясь к серверу безопасности в соответствии с реализованным интерфейсом. Корректность обращений к серверу безопасности должна обеспечиваться проверкой программного обеспечения, устанавливаемого на ПЭВМ АС, и его периодическим тестированием.

Для выяснения соответствия ЭлД его проверочному коду последний должен содержать информацию о том, на какой ПЭВМ из АС был создан этот проверочный код. Для реализации этой потребности предлагается ввести в формат кода аутентификации условный номер ПЭВМ (NC).

Как уже упоминалось выше, таблицы достоверности должны периодически (например, раз в год) заменяться. При переходе на новые варианты ТД могут возникать нештатные ситуации, если на какой-либо ПЭВМ замена произведена не вовремя. Для раннего отсева и облегчения разбора таких ситуаций предлагается ввести в формат кода аутентификации условный номер таблицы достоверности (NТД).

Для привязки кода аутентификации к технологической операции, для которой он был выработан в формат кода аутентификации предлагается ввести номер технологической операции (NТО).

Таким образом, предлагается следующий формат кода аутентификации (КА) электронных документов, суммарная длина которого будет составлять 16 байт.

$$\text{КА} = (\text{NТД}, \text{NC}, \text{NTO}, \text{12 байт из } h(h(\text{ЭлД}),\text{КОД})).$$

Созданная система основана на применении аппаратного средства защиты — устройства «Аккорд СБ/КА», объединяющего в своей архитектуре аппаратную часть устройств класса «Аккорд», обеспечивающую защиту от несанкционированного доступа к ресурсам ПЭВМ, а также энергонезависимую память для хранения КОД. Эта память и данные, хранящиеся в ней, являются недоступными с уровня пользовательских задач ПЭВМ. Кроме этого, устройство содержит высокопроизводительный процессор ADSP2181 для реализации функций КА. Взаимодействие компонент устройства реализуется применением трех специализированных (полузаказных) СБИС, объединенных в единую систему магистралью процессора. Вырабатываемый данным устройством КА представляет собой описанный блок данных (16 байт), при этом достигнута производительность более 20000 КА/с.

На основе КА можно построить подсистему технологической защиты ЭлД как средство обеспечения электронного документооборота.

6. ТЕХНОЛОГИЧЕСКАЯ ЗАЩИТА ЭЛЕКТРОННЫХ ДОКУМЕНТОВ

Известные в теории защиты информации модели и системы рассматривают, как правило, проблемы доступа субъекта (оператора) к объекту (информации). В этой связи и средства защиты ориентированы лишь на аутентификацию субъекта и (в лучшем случае) создание изолированной программной среды (ИПС) функционирования АС. При этом хорошо известно, а для большинства моделей и доказано, что проблема разграничения доступа неразрешима в рамках этих моделей (известный тезис: «100 % защиты не бывает»). Для того чтобы преодолеть это противоречие, либо делаются нереализуемые допущения (типа «В АС в любой момент времени используются только проверенные программы, не влияющие одна на другую»), либо предлагается решать все возникающие трудности за счет усиления организационных мер защиты. Как один, так и другой вариант представляются нам совершенно неадекватными.

Действительно, что означает «проверенность» программы и отсутствие взаимовлияния программ? Видимо, единственной возможностью установить это является применение некоторой другой программы, которая и должна установить «проверенность» исследуемой программы. Но, как обычно, возникает следующий круг — как установить «проверенность» проверочной программы — и т.д. Представляется, что рассматриваемая проблема сводится к проблеме распознавания самоприменимости программ — которая, как известно, неразрешима.

Таким образом, если обеспечить гарантированную(доказательно) «правильность» обработки информации в АС невозможно, то не стоит ли попытаться создать контролируемую технологию обработки документа в АС? Очевидно, что контролю должны подлежать в данном случае не изолированность программной среды, не неизменность (целостность)

программного обеспечения, а состояние обрабатываемой информации — электронного документа. Такой контроль — контроль «внутри» технологии обработки ЭлД — будем называть технологической защитой, а совокупность средств технологического контроля — подсистемой технологической защиты (ПСТЗ).

В настоящее время невозможно с высокой степенью полноты описать модели, механизмы, опыт технологической защиты — всего этого еще нет. Есть только первые попытки создания и осмысления ПСТЗ. Ниже попытаемся описать некоторые аспекты этой проблемы.

6.1. Можно ли обойтись без технологической защиты?

Да, ведь до сих пор обходились, а мир остался трехмерным, однородным. Правда, до сих пор никому не удалось добиться надежной защиты своих АС, хотя применялись сертифицированные СЗИ НСД и СКЗИ. В случаях обнаружения НСД к информации обычно говорят о том, что средства применялись неправильно, что владелец АС пренебрегал оргмерами, что ключи не сохранялись в тайне — и все это абсолютно верно, так же, как и абсолютно скучно. Действительно, если все это говорят, не значит ли это, что утечка сведений, их модификация и т.д. были все-таки обнаружены? Если да, то почему так поздно, что пришлось все это говорить? Не лучше ли обнаруживать возникающие проблемы не спустя недели, а непосредственно тогда, когда они возникают (как говорят, в реальном масштабе времени)? Вот именно это и призвана делать подсистема технологической защиты.

6.2. Почему нужна технологическая защита ЭлД?

Потому, что сегодня невозможно создать абсолютно надежных СЗИ. Желающим убедиться в этом можно рекомендовать разобраться с известными моделями защиты информации, а затем поэкспериментировать с фундаментальными результатами теории алгоритмов — в частности, с результатами Геделя, Черча, Клини и других.

6.3. Что является объектом защиты в подсистеме технологической защиты?

Наверное, это основной вопрос. Для ответа на него, возможно, потребовался бы специальный труд, посвященный взаимосвязи понятий «информация», «сведения», «документ». Без подробного обсуждения этих гло-

бальных вопросов мы зафиксируем объект защиты как электронный документ, имея в виду имеющийся у каждого опыт работы с обычными документами.

Садясь за свой рабочий стол и разбирая подготовленные для работы материалы, каждый из нас без всяких усилий понимает: это — *копия* приказа по Гостелекому, это — выписка из протокола ГМЭК (документ, т.к. содержит подпись, на бланке и т.д.).

Действительно, мы легко отличаем оригинал и копию документа, причем способ доставки материалов при этом не имеет значения. Будет ли документом текст без каких-либо реквизитов, подписей, печатей, даже если его (текст) доставили по закрытому каналу связи или фельдъегерской почтой? Что делает акцию или вексель документом? Зафиксированная на бумаге информация в виде текста и изображения, или возможность доказать подлинность и таким образом использовать в документообороте?

Похоже, что содержание документа и собственно документ — это весьма разные понятия.

Электронный документ представляет собой не просто совокупность сведений, а, как и его бумажный аналог, должен обладать рядом специфических качеств. Так, например, для обычного документа практически не существует проблемы «копия—оригинал». Почти всегда, проведя визуальный, приборный или криминалистический контроль, можно с высокой степенью вероятности установить, оригинал или копия у Вас в руках. Обычно это достигается за счет внедрения в документ (а иногда и извлечения из документа) некоторых элементов, несущих в себе информацию о подлинности объекта — их можно назвать трейлерами безопасности. От качества этих трейлеров зависит сложность подделки документов — так, есть хорошо защищенные документы (например, доллар США — хотя известны и подделки), есть документы со слабой защитой (например, гарантийный талон на СЗИ НСД «Аккорд» — подделки еще не встречались).

В отличие от бумажного, в электронном документообороте проблема «оригинал—копия» еще не решена. Дело, видимо, в том, что ввести трейлеры безопасности для электронного документа до сих пор пытаются в полной аналогии с бумажным документом, а именно — трейлер позволяет установить целостность документа и подтвердить/опровергнуть авторство.

Нам кажется, что этого недостаточно для электронного документа. Действительно, вряд ли инструмент, посредством которого создается бумажный документ, может стать источником искажения оригинала. А вот инструмент электронного документооборота вполне может быть источником искажений, уж хотя бы в силу своей сложности и связанным с этим несовершенством. В этой связи представляется мало осмысленным ставить на сведения в электронном виде трейлер безопасности уже после того, как обработка сведений завершена — в это время сведения, в общем случае, уже искажены. Нам представляется, что электронным документом вполне можно считать сведения в электронном виде вместе со всей совокупностью трейлеров безопасности, фиксирующих целостность тех или иных свойств на протяжении технологического процесса обработки этих сведений.

6.4. Как снизить требуемые ресурсы?

Этот вопрос связан с тем, не слишком ли много времени займет выработка трейлеров безопасности, и не слишком ли много места займут они в составе ЭлД?

Затраты времени связаны с вычислительной сложностью процедур, и их целесообразно выбирать из условий «реального времени» для каждого из этапов (в частности, выработка и проверка трейлеров могут быть разделены). Включать же в состав ЭлД все трейлеры тоже нет необходимости, на каждом этапе обработки достаточно иметь всего два трейлера, характеризующих ЭлД перед началом операции и после ее завершения. В этом смысле возможна следующая технология:

Защита документа осуществляется применением двух КА — входного и выходного для каждого этапа. При этом КА должны вырабатываться аппаратно с привязкой КА к процедуре обработки. Для поступившего документа (с КА и ЭЦП) вырабатывается $КА_2$ и только затем снимается ЭЦП.

Далее:

на следующем этапе (n) вырабатывается $КА_{n+1}$ и снимается $КА_{n-1}$. Таким образом, в любой момент времени документ защищен двумя КА - $КА_n$ и $КА_{n+1}$. КА должны вырабатываться и проверяться для документа, размещенного в оперативной памяти ЭВМ, в которой создана и поддерживается ИПС. Снятие $КА_{n-1}$ выполняется после установки $КА_{n+1}$.

Возникает вопрос, «почему же нельзя сначала выработать ЭлД, а затем его проверить и лишь после этого подписать?».

В силу «проклятия времени». Действительно, при обычной «бумажной» технологии время создания документа $T_б$ значительно больше времени его проверки $T_п$

$$T_б >> T_п$$

В этом случае временем проверки в общих затратах можно пренебречь

$$T_о = T_б + T_п \cong T_б$$

При электронном документообороте время создания документа $T_э$ значительно меньше, так что

$$T_э << T_б$$

и, более того,

$$T_э << T_п$$

В этом случае, при ручной проверке

$$T_о' = T_э' + T_п \approx T_п$$

т.е. время проверки становится решающим фактором производительности АС. Отсюда ясно, что время на процесс проверки нужно максимально сокращать, повышая тем самым общую производительность АС.

Хорошим механизмом для повышения производительности может быть технология серверов безопасности, где сервер безопасности — некоторое внешнее активное средство, позволяющее решать вопросы безопасности за счет собственных ресурсов, с минимальным использованием ресурсов АС.

6.5. Как же отличить копию ЭлД от оригинала?

По большому счету, как в обычном, так и в электронном документообороте копию от оригинала отличает технология изготовления.

Вспомните, как проверяют ценные бумаги: есть ли элементы, светящиеся в ультрафиолете? Есть ли водяные знаки? Нет ли нарушений рисунка? Что это, если не попытка на основе простейших оценок восстановить характерные этапы технологического процесса! Отметим, однако, что каждый из этих и многих других трейлеров безопасности (именно так и нужно относиться к средствам защиты документа) может быть, а может и не быть на документе (за неимением гербовой пишем на простой). Тем не менее, для ряда важных документов набор трейлеров безопасности, позволяющих восстановить с той или иной точностью характерные этапы технологического процесса, фиксирован нормативными документами. Для документов попроще трейлеры фиксируются традиционно — например, письмо предприятия должно быть на бланке (первый трейлер) и содержать подпись ответственного лица (второй трейлер). Гарантийное письмо должно быть на бланке, содержать две подписи и оттиск мастичной печати. Обратите внимание — без трейлеров безопасности текст (информация, сведения) представляет собой всего лишь черновик, копию и т.д. Сведения становятся документом только при наличии трейлеров безопасности.

Представляется, что именно такой подход должен быть применен и в электронном документообороте.

Разница лишь в том, что для бумажного документа нарушение технологии определяется известными методами контроля (визуальный, приборный, криминалистический), а вот методы контроля для ЭлД следует разработать. Однако, если на основе известных методов для бумажного документа можно установить нарушение технологии (например, другой состав носителя или наличие лишнего этапа ксерокопирования), то и анализируя совокупность трейлеров безопасности, можно восстановить состав операций изготовления ЭлД. Состав операций (технологического процесса) можно анализировать на соответствие базовому и на этой основе делать вывод об аутентичности документа или копии.

Действительно, определим некоторый алфавит A как конечную систему попарно различимых знаков (букв алфавита). Пусть каждая буква соответствует той или иной операции технологического процесса. Обозначим R слово в данном алфавите, соответствующее зафиксированной (правильной, эталонной) для данной АС и данного типа ЭлД технологии обработки. В реальной жизни некоторые отклонения от данной технологии могут быть вполне допустимы. При этом восстановленная по трейлерам безопасности технология изготовления будет описываться словом S, в общем случае отличным от R. Такой формализм соответствует теории ассоциативных исчислений, если рассматривать совокупность всех слов во введенном алфавите вместе с какой-либо конечной системой допустимых подстановок.

В соответствии с [4] будем говорить о вхождении слова L в слово M, в том случае, если L является частью слова M. Преобразования одних слов в другие задаются посредством некоторых допустимых подстановок в виде $P \to Q$ или $P — Q$, где P и Q — слова в алфавите A.

Применение *ориентированной подстановки* $P \to Q$ к слову R возможно в том случае, когда в нем имеется хотя бы одно вхождение левой части P; оно заключается в замене любого одного такого вхождения соответствующей правой частью Q. Применение же *неориентированной подстановки* $P — Q$ допускает как замену вхождения левой части правой, так и замену вхождения правой части левой. Будем рассматривать преимущественно неориентированные подстановки, называя их просто подстановками.

Если слово R может быть преобразовано в слово S посредством однократного применения допустимой подстановки, то и S может быть преобразовано в R таким же путем; в таком случае R и S будем называть *смежными* словами. Последовательность слов $R_1, R_2, \ldots, R_{n-1}, R_n$, таких что R_1 и R_2 смежны, R_2 и R_3 смежны, ... R_{n-1} и R_n смежны, будем называть *дедуктивной цепочкой*, ведущей от R_1 к R_n. Если существует дедуктивная цепочка, ведущая от слова R к слову S, то, очевидно, существует и дедуктивная цепочка, ведущая от S к R; в таком случае слова будем называть эквивалентными и обозначать это так: $R \sim S$. Ясно так же, что если $S \sim R$ и $R \sim T$, то $S \sim T$.

Лемма. *Пусть* P~Q*; тогда, если в каком-либо слове* R *имеется вхождение* P*, то в результате подстановки вместо него* Q*, получается слово, эквивалентное* R.

При таком описании распознавание оригинала и копии сводится к «ограниченной проблеме слов», которая заключается в следующем:

Для любых двух слов R и T в данном ассоциативном исчислении требуется узнать, можно ли преобразовать одно в другое последовательным применением не более чем k разных допустимых подстановок (k — произвольное, но фиксированное число).

Допустимыми подстановками для установления эквивалентности R и S в нашем случае будут являться подстановки, соответствующие разрешенным отклонениям от эталонного технологического процесса.

Таким образом, анализируя с помощью рассмотренных механизмов совокупность трейлеров безопасности как некоторое слово, описывающее технологический процесс изготовления ЭлД, можно на основе понятия эквивалентности установить аутентичность документа, несмотря на отличия реального технологического процесса от эталонного.

Конструктивность механизмов ассоциативных исчислений (работы А.А. Маркова, П.С. Новикова, Г.С. Цейтина, их учеников и др.) позволяет предлагаемый механизм распознавания «копия—оригинал» считать также конструктивным.

Естественным требованием здесь является также то, что по значению трейлера безопасности должна быть возможность установить субъекта, выработавшего данный трейлер.

Сегодня в нормативной базе отсутствует определение ЭлД, что значительно затрудняет создание действительно защищенного документооборота. Известны лишь некоторые (неполные, на наш взгляд) определения документа. Одно из них приведено в статье 1 Федерального закона «Об обязательном экземпляре документов», а именно: «Документ — материальный объект с зафиксированной на нем информацией в виде текста, звукозаписи или изображения, предназначенный для передачи во времени и пространстве в целях хранения и общественного использования». Это определение не представляется нам конструктивным, так как в полном соответствии с ним один и тот же материальный носитель с одной и той же информацией может являться документом (если он предназначен для общественного пользования), а может и не являться — в другом случае. В принципе это означает, что для определения, документ ли перед нами, нужно знание целей его создателей. А как проанализировать цели? Как их вообще можно узнать?

Более совершенным является определение, данное в Федеральном законе «Об информации, информатизации и защите информации». Это определение гласит: «Документированная информация (документ) — зафиксированная на материальном носителе информация с реквизитами, позволяющими ее идентифицировать».

Это значительный шаг вперед, но не окончательный. На основе реквизитов можно идентифицировать информацию, но нельзя аутентифицировать сведения, и, тем более, установить, копия или оригинал перед нами.

Определяя ЭлД как сведения, зафиксированные на материальном носителе с реквизитами, позволяющими его (документ) идентифицировать, и трейлерами безопасности, позволяющими их (сведения) аутентифицировать, можно решить и проблему «оригинал—копия», и важнейшую задачу расследования компьютерных преступлений. Совершенно очевидно, что это возможно лишь для ЭлД, определенных выше, и только для защищенных АС. В создании соответствующей нормативной базы видится важнейшая роль государственного регулирования в сфере информатизации.

6.6. От каких же параметров должен зависеть трейлер безопасности в системе технологического контроля?

Если нам нужно по набору трейлеров восстановить технологию обработки ЭлД (т.е. состав и последовательность операций), то трейлер безопасности должен отображать тип (номер) операции, и их последовательность—последовательность операций. Так как нужно установить и связь последовательности трейлеров с обработкой не любого, а конкретного ЭлД, естественно, что трейлер должен основываться и на информации об ЭлД. Было бы полезно также иметь возможность установить, на каком рабочем месте, кто и когда выполнял ту или иную технологическую операцию по обработке документа. В этой связи трейлер безопасности можно описать как

$$ТБ = ТБ\,(\,K_{оп},\ ЭлД,\ |\bar{x}|),$$

где:

$K_{оп}$ — ключ операции

$|\bar{x}|$ — множество векторов вспомогательных параметров

Очевидно, что в качестве ТБ может использоваться КА.

6.7. Какой может быть структура подсистемы технологической защиты?

Такая подсистема должна содержать:

1) сервер кодов аутентификации — как активный элемент, вырабатывающий (проверяющий) КА как вариант ТБ;

2) АРМ администрирования и разбора конфликтов — смысл его понятен из названия;

3) АРМ ключей — его функциональность — выработка всех ключей, используемых в подсистеме;

4) АРМ персонализации — это средство, которое позволяет снабдить универсальные средства КА индивидуальными признаками (персонифицировать их).

7. О ЗАЩИТЕ ИНФОРМАЦИИ В ЛВС ОТ НСД ИЗ ВНЕШНЕЙ СРЕДЫ

Ниже рассмотрим новый подход к *защите информации в локальных сетях (LAN), связанных с глобальными сетями (WAN).* Мы считаем, что актуальна именно эта проблема, т.к.:

1) известно, как обеспечить нужный уровень безопасности информации в LAN;

2) известно, как обеспечить нужный уровень безопасности информации в корпоративной сети;

3) известно, как обеспечить нужный уровень безопасности информации при передаче конфиденциальных данных;

4) ясно, что изменить технологии WAN в целом невозможно.

Таким образом актуальной является следующая постановка:

— организация безопасного доступа из незащищенной среды к открытой информации в ЛВС, обрабатывающей информацию различных уровней конфиденциальности.

Здесь следует обратить внимание на принципиальное отличие терминов «информационный ресурс» и «информация». Первый из них связан со статическим состоянием сведений, а второй — с динамическим. Действительно, *сведения приобретают функциональность информации* только в процессе «движения» — при изготовлении «копии» сведений — т.е. *при отображении сведений на материальный объект, который может служить их носителем.* Это отличие является принципиальным, именно с точки зрения защиты, в той мере, в которой защита от несанкционированного доступа к объекту отличается от защиты потоков информации. С учетом этих отличий построено дальнейшее рассмотрение.

Согласно [5], «Межсетевой Экран (МЭ) — это локальное (однокомпонентное) или функционально-распределенное программное (программно-аппаратное) средство (комплекс), реализующее контроль за информацией, поступающей в АС и/или выходящей из АС. МЭ обеспечивает защиту АС посредством фильтрации информации, т.е. ее анализа по совокупности критериев и принятия решения о ее распространении в (из) АС на основе заданных правил, проводя таким образом разграничение доступа субъектов из одной АС к объектам другой АС. Каждое правило запрещает или разрешает передачу информации определенного вида между субъектами и объектами. Как следствие, субъекты из одной АС получают доступ только к разрешенным информационным объектам из другой АС. Интерпретация набора правил выполняется последовательностью фильтров, которые разрешают или запрещают передачу данных (пакетов) на следующей фильтр или уровень протокола».

Это определение позволяет ввести следующую классификацию МЭ.

По типу реализации:

— локальные или функционально-распределенные программные или программно-аппаратные средства.

По типу контроля:

— разграничение доступа субъектов одной АС к объектам другой АС.

Очевидно, что данная классификация ограничена — отсутствует всякое упоминание контроля потоков информации, т.е. предполагается наличие дискреционного механизма разграничения доступа, но отсутствует мандатный механизм. Это странно, т.к. хорошо известны ограничения дискреционного механизма — его целесообразно применять для защиты локальных ресурсов, но эффективность его проблематична для глобальных сетей.

Коротко рассмотрим известные типы МЭ.

1. МЭ на основе фильтрации пакетов (ФП) анализируют значения полей «адрес» и «порт» в заголовке IP-пакета и на основании заранее определенных правил пропускают либо отклоняют пакет. Ясно, что на основе МЭ ФП ничего напоминающего мандатный механизм реализовать невозможно, хотя простота, дешевизна и незначительное влияние на производительность сети позволяет найти место для приложения экранов этого типа.

2. МЭ на основе технологии контекстной проверки (КП) просматривают пакеты на сетевом уровне и анализируют некоторые данные в пакете по отношению к применяемым сервисам. В политике информационной безопасности (ПИБ), реализуемой с применением этого типа МЭ, обычно требование доступности является более значимым, чем целостность и конфиденциальность, и в этой связи для жестких мер защиты информации МЭ КП не всегда применимы.

3. МЭ на основе технологии «посредника» являются промежуточным звеном передачи пакетов между сервером и клиентом. При этом явно должно быть определено допустимое подмножество команд для конкретного протокола — все прочие будут отклонены. При этом МЭ с посредником в общем случае не проверяют содержащиеся в пакете данные. Таким образом, в этом случае наиболее последовательно реализуется дискреционная модель, но даже не предусмотрена возможность реализации мандатной модели.

В целом можно сделать следующий вывод — все известные *МЭ являются локальными средствами*, с той или иной степенью полноты реализующими дискреционный механизм разграничения доступа. Как результат, при применении МЭ фиксируется «место» принятия решения о доступе, но невозможно зафиксировать «место» порождения данных, их семантику и, следовательно, отсутствует возможность анализа потоков. Для преодоления выявленных ограничений *следует изучить возможность реализации функционально-распределенного программно-аппаратного средства с поддержкой мандатного механизма управления потоками.*

Действительно, пусть дискреционным механизмом определено, что субъект S_1 имеет доступ к объектам F_1 и F_2, а субъект S_2 имеет доступ только к объекту F_2. В этом случае ничто не помешает S_1 скопировать сведения из F_1 в F_2. При этом S_2, обладая доступом к F_2, фактически получает доступ к сведениям из F_1. Таким образом, защита информации посредством дискреци-

онного механизма не является полной, необходимо с учетом возможности доступа S_2 контролировать также поток от F_1 к F_2, организованный S_1, т.е. необходимо введение мандатного механизма.

Отсутствие мандатного механизма является, на наш взгляд, основной допускаемой в настоящее время ошибкой, от преодоления которой зависит эффективность защиты. *Действительно, принятие решения о доступе субъекта к объекту связано с характеристиками субъекта и объекта, но очень слабо связано с тем, откуда субъект запрашивает сведения из объекта и с помощью каких средств он это делает.*

Известны механизмы управления потоками на основе меток безопасности, включающих как иерархические, так и неиерархические признаки. Данные метки ассоциируются с объектами и субъектами АС, и решение о возможности доступа (организации потока) принимается на основе анализа меток. Проиллюстрировать данный механизм можно следующим примером.

Рассмотрим описание типа

ID (I; P_1, P_2,...P_n), где

ID — идентификатор или субъекта (S_i), или объекта (F_i);

I — иерархический признак;

P_i — неиерархический признак.

Пусть иерархические признаки описывают:

Для объекта гриф	Для субъекта допуск
1 — красный (к)	1 — (к, ж, з, б)
2 — желтый (ж)	2 — (ж, з, б)
3 — зеленый (з)	3 — (з, б)
4 — белый (б)	4 — (б)

Неиерархические признаки описывают семантику (тематику) сведений, содержащихся в объекте:

1 — манго
2 — яблоки
3 — груши

Пусть:

S_1 (2; 1, 2, 3)
S_2 (1; 2, 3)
F_1 (3; 2, 3)
F_2 (4; 1, 2),

интерпретация:

— субъекту S_2 доступны сведения о яблоках и грушах (неиерархические признаки 2 и 3) всех степеней зрелости — белых, зеленых, желтых и красных (иерархический признак 1), но недоступны никакие сведения о манго, даже самых незрелых;

— субъекту S_1, напротив, доступны сведения о всех фруктах, информация о которых имеется в системе, но лишь в той мере, пока они зреют. Сведения о созревших («красных» — иерархический признак 1) фруктах субъекту S_1 не доступны.

При этом S_1 имеет доступ и к F_1, и к F_2. Для S_2 доступен F_1, но недоступен F_2, хотя допуск S_2 значительно выше грифа F_2. Дело в том, что по совокупности неиерархических признаков S_2 имеет право доступа к сведениям, касающимся яблок и груш, но не имеет права доступа к сведениям о манго, которые содержатся в F_2.

Пусть S_1 в процессе обработки F_1 и F_2 создает объект F_3, который при этом может быть описан как:

F_3 (3; 1, 2, 3).

Таким образом, хотя допуск S_2 позволяет иметь доступ к сведениям, имеющим гриф как у F_3, но S_2 не получит доступа к F_3, т.к. множество неиерархических признаков S_2 уже аналогичного для F_3.

Очевидно, что такой механизм позволит контролировать потоки. Необходимым условием при этом является наличие меток безопасности и средств их обработки.

Ставя перед собой задачу обеспечения необходимого уровня защиты информации в защищенной ЛВС при доступе в том числе и из открытой внешней сети, необходимо разработать механизмы, позволяющие осуществлять:

— назначение меток субъектам и объектам АС;

— хранение (привязку) меток в ассоциированном с субъектами и объектами виде;

— сопровождение (обработку и преобразование) меток в жизненном цикле информации.

Основой этих механизмов является:

— политика информационной безопасности и процедуры ее описания;

— файловая система для объектов и база полномочий (account) субъектов;

— драйвер файловой системы и менеджер памяти.

По нашему мнению, наиболее перспективными ОС для реализации данной концепции являются Windows NT 5.0 и Windows 98, причем в качестве сервера предпочтительнее Windows NT, а в качестве клиента — Windows 98. Ориентация на Novell NetWare будет возможна только при открытии разработчиком исходного текста ряда процедур.

Подводя итог, отметим, что *достижение необходимого уровня безопасности возможно при реализации концепции функционально-распределенного МЭ с поддержкой семантического анализа данных на основе мандатного механизма.* Очевидно, что в качестве меток ЭлД могут использоваться КА, ассоциированные через ключи с субъектами и объектами.

8. ПРОЕКТИРОВАНИЕ АППАРАТНЫХ СРЕДСТВ СЗИ

В предыдущем разделе были сформулированы основные требования к функциональности и составу аппаратных средств СЗИ. Это позволяет определить перечень функциональных узлов, которые должны быть реализованы в контроллере СЗИ, взаимосвязи между ними и сформулировать некоторые требования к их реализации.

Как было показано, минимальный состав должен содержать следующие функциональные узлы.

1.Touch-memory интерфейс (ТМИ).
2. Энерго-независимая память(ЭНП).
3. Датчик случайных чисел (ДСЧ).
4. ROM BIOS.

Кроме этого, обязательно должен быть организован

5. Интерфейс связи с ЭВМ (ИВВ)

Примерная структура контроллера при этом может выглядеть следующим образом (рис. 2.3)

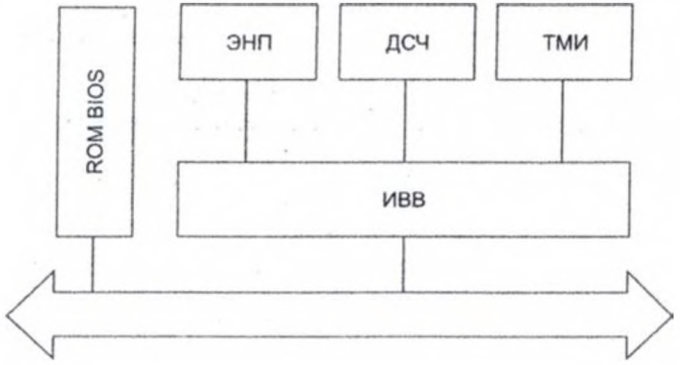


Рисунок 2.3

При этом в ЭНП хранится (как минимум) эталонная информация для процедур идентификации/аутентификации пользователя, ссылки на полномочия пользователя и контрольные суммы для контроля целостности системных областей и системных файлов.

По ТМИ в контроллер СЗИ поступает информация о пользователе и (в общем случае) контрольная сумма прикладных задач и данных.

Введем множество пользователей системы U, $|U|= m$, $U=\{U_i\}_{i=\overline{1,m}}$, и множество прав пользователей R, $|R|= p$, т.е. $R = \{r_1,...,r_p\}$.

Рассмотрим подмножество $R_H \subseteq R$, $|R_H|= p_H$, содержащее элементы $r_{H1},...,r_{Hp_H}$, а так же подмножества вида $R_i \subseteq R$, $|R_i|= p_i$, $(r_{i1},...,r_{ip_i}) \in R$.

Тогда определим множество всевозможных подмножеств R_i множества R : $R_i \in \{R_i\}_{i=\overline{1,m}}$. Элементы этого множества есть права доступа пользователей системы $U_i \in U$.

В ЭНП для i-ого пользователя зафиксирована $\{I_i, P_i, G_i, R_i\}$, где

I_i — идентификатор пользователя U_i, $I_i \in \{I_i\}_{i=\overline{1,m}}$

P_i — пароль пользователя U_i, $P_i \in \{P_i\}_{i=\overline{1,m}}$

G_i — контрольная сумма, $G_i \in \{G_i\}_{i=\overline{1,m}}$

R_i — права пользователя U_i, $R_i \in \{R_i\}_{i=\overline{1,m}}$

В ТМ, принадлежащей пользователю U_i, при этом хранится пара $\{I_i, G_i\}$

Рассмотрим данную схему, принимая во внимание особенности потоков информации между узлами устройства, и на этой базе определим для них основные особенности (в том числе ресурсные).

Исходя из мультипликативной парадигмы защиты, будем полагать, что на данном этапе злоумышленник обладает возможностью использовать все ресурсы ПЭВМ.

Как отмечалось выше, на ТМ должна храниться не только идентифицирующая пользователя информация, но и отчужденные данные для аутентификации и контроля целостности, ключи шифрования (или их части). Утечка (перехват) этих данных вполне может привести к дискредитации защиты именно в силу нарушения принципа «отчуждай и властвуй». Действительно, так как изменение данных в ПЭВМ может сопровождаться изменениями данных в ТМ, контроллер должен обеспечивать и чтение, и запись. При этом логика действий нарушителя могла бы быть такой (в предположении, что нарушитель является легальным пользователем АС):

Подготовить разрушающее программное воздействие (РПВ), выполняющее следующие функции:

а) фиксируются тройки $\{I_i, P_i, G_i\}$, и соответствующие им права R_i при легальном входе пользователей за период времени T;

б) анализируется информация о правах $\{R_i\}_{i=\overline{1,k}} \subset \{R_i\}_{i=\overline{1,m}}$, где $k \leqslant m$, k — число различных пользователей, зафиксированных РПВ и $R_j \subseteq \{R\}_{i=\overline{1,k}}$ выбирается в качестве прав, которые необходимы нарушителю,

в) при загрузке нарушителем U_n с $\{I_n, P_n, G_n\}$ эта совокупность заменяется на $\{I_j, P_j, G_j\}$, что позволяет пользователю U_n получить права R_j.

Определим предикат P следующим образом:

$$P(r_{H1},...,r_{Hp_H},i)=1 \Leftrightarrow (r_{H1},...,r_{Hp_H})\in R_i \qquad (*)$$

Анализ зафиксированных прав $\{R\}_{i=1}^{k}$ заключается в вычислении предиката $P(r_{H1},...,r_{Hp_H},i)$, $\forall i=\overline{1,m}$. Если $P(r_{H1},...,r_{Hp_H},i)=1$, т.е. $(r_{H1},...,r_{Hp_H})\in R_i$, то право $R_i\subseteq\{R\}_{i=1}^{k}$, выбирается как необходимое нарушителю.

Опишем машину Тьюринга, реализующую данный механизм.
Внешний алфавит машины:

$$\{U_1,...,U_i,...,U_m,U_H,r_1,...,r_p\}$$

Внутренний алфавит машины:

$$\{Л,П,Н,Λ,!\}\cup\{q_i\},\ |\{q_i\}|=\sum_{j=1}^{m}p_j+mp_H+m+1$$

Входное слово: информация о пользователе U_i, поступающая по ТМИ.
Выходное слово: право, полученное нарушителем при входе в систему.

Функциональная схема машины представлена на рис. 2.4. При этом часть машины, обозначенная на рисунке звездочкой (*), вычисляет предикат (*) и изображена на рис. 2.5.

Принципиальная осуществимость атак такого рода возможна лишь потому, что:

а) носящие конфиденциальный характер данные попадают в оперативную память ЭВМ, где к ним может получить доступ РПВ;

б) обработка данных выполняется процессором ЭВМ, который не различает источника данных — отсюда следует возможность подмены данных.

Очевидно, что для того, чтобы противостоять атакам этого типа, необходимо изменить структуру контроллера (организовать каналы передачи данных, минуя RAM ЭВМ, и ввести микропроцессорное устройство управления (MCU), на основе которого будет осуществляться обработка данных, носящих конфиденциальный характер. В этом случае примерная структура контроллера может выглядеть, как на рис. 2.6.

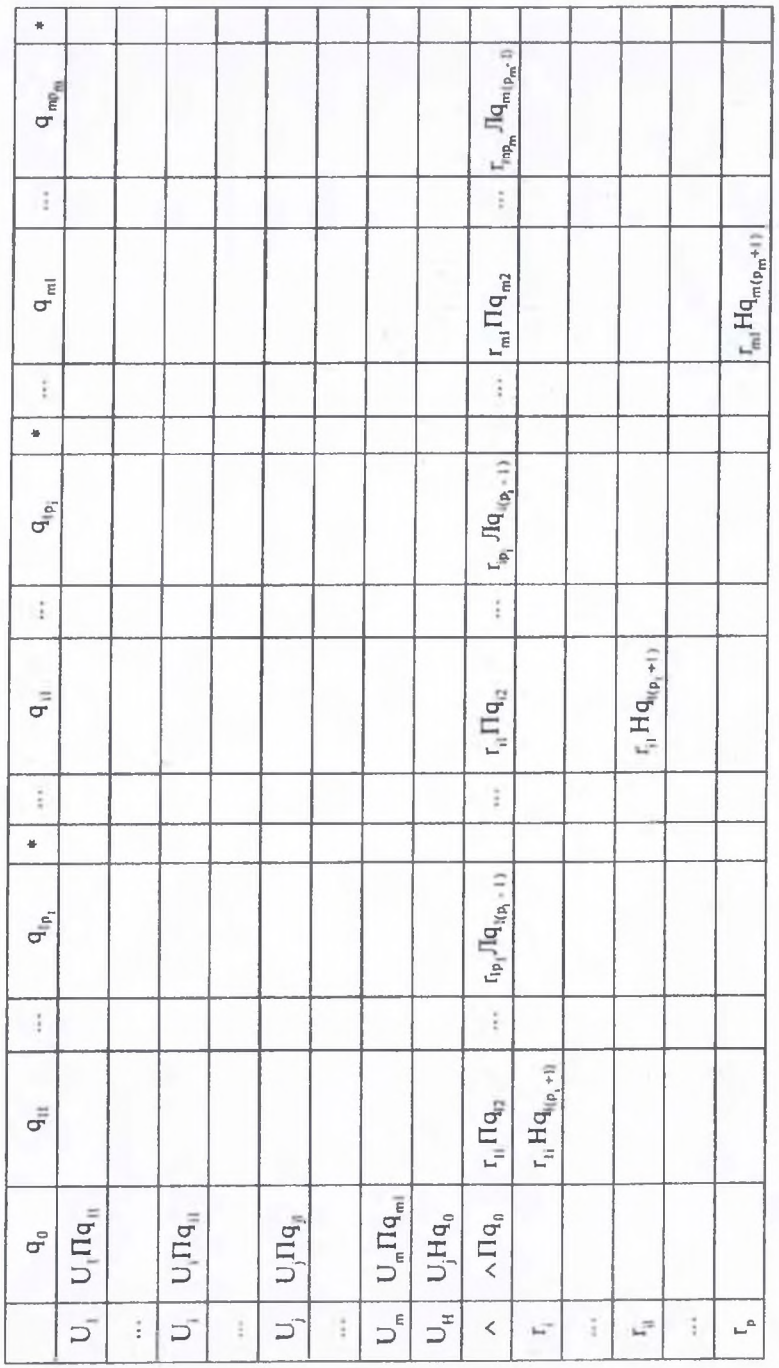


Рисунок 2.4

	q_0	...	$q_{i(p_1+1)}$	$q_{i(p_1+2)}$	...	$q_{i(p_1+i)}$	...	$q_{i(p_1+p_H)}$	$q_{i(p_1+p_H+1)}$
r_1			$r_1 Пq_{i(p_1+1)}$		...		...		
r_2			$r_2 Пq_{i(p_1+1)}$		...		...		
...			...	...	...	...	...		
r_{H1-1}			$r_{H1-1} Пq_{i(p_1+1)}$		...		...		
r_{H1}			$r_{H1} Пq_{i(p_1+2)}$		...		...		
r_{H1+1}			$r_{H1+1} Пq_0$	$r_{H1+1} Пq_{i(p_1+2)}$	...		...		
...			...	...	...	...	...		
r_{H2-1}	$r_{H2-1} Пq_0$			$r_{H2-1} Пq_{i(p_1+2)}$	...		...		
r_{H2}	$r_{H2} Пq_0$			$r_{H2} Пq_{i(p_1+3)}$	...		...		
r_{H2+1}	$r_{H2+1} Пq_0$			$r_{H2+1} Пq_{0\ 1}$	...		...		
...					...		...		
r_{Hi-1}	$r_{Hi-1} Пq_0$				...	$r_{Hi-1} Пq_{i(p_1+i)}$	...		
r_{Hi}	$r_{Hi} Пq_0$				...	$r_{Hi-1} Пq_{i(p_1+i+1)}$	...		
r_{Hi+1}	$r_{Hi+1} Пq_0$				...	$r_{Hi+1} Пq_0$	...		
...			...	...	...	...	...		
$r_{H(p_H-1)}$					...		...	$r_{H(p_H-1)} Пq_{i(p_1+p_H)}$	
r_{HP_H}					...		...	$r_{HP_H} Пq_{i(p_1+p_H+1)}$	
$r_{H(p_H+1)}$					...		...		
...			...	...	...	...	...		
$r_{p\,1}$	$r_{p\,1} Пq_0$				...		...		
r_p	$r_p Пq_0$				...		...		
U_H					...		...		$U_1 Hq_0$

(*)

Рисунок 2.5

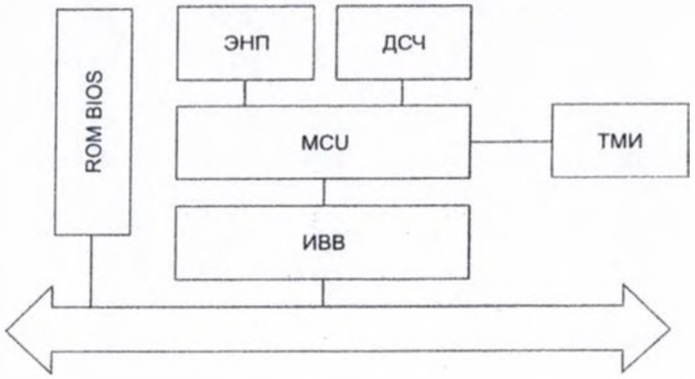


Рисунок 2.6

Такое устройство вполне работоспособно, и необходимо лишь доопределить его дополнительными функциональными блоками и рассмотреть особенности связей и функциональность MCU.

Важнейшей особенностью любого устройства является адаптивность. Так, например, до сих пор не существует стандарта на шину ISA — стало быть, вероятна необходимость адаптации ИВВ. Дополнение контроллера функциональными блоками может привести к необходимости изменения ПО MCU. Это означает, что функциональность основных узлов контроллера должна определяться уже после изготовления контроллера, путем программирования с использованием специализированного программатора, который, естественно, тоже должен входить в состав контроллера.

Для того чтобы снять ограничения по подключению к контроллеру дополнительных устройств, целесообразно использовать магистрально-модульный принцип организации вычислительных устройств. При этом полезно организовать поддержку наиболее применимых интерфейсов и шин, и в дальнейшем их можно будет использовать для расширения функциональных возможностей устройства. Такими шинами и интерфейсами могли бы быть RS232C, I²C, SPI. С точки зрения дополнительных устройств, структура устройства может быть такой, как показано на рис. 2.7.

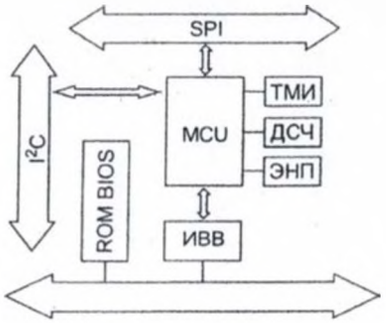


Рисунок 2.7

Рассматривая этот вариант структурной схемы, нетрудно заметить присутствующие противоречия. Так, очевидно, что программное обеспечение, записанное в ROM BIOS, с одной стороны, не должно меняться для гарантированности стабильности свойств, но с другой стороны должна быть предусмотрена возможность его модификации для обеспечения адаптивности. Это же касается и других программируемых узлов устройства. Данное противоречие может быть разрешено путем введения различных режимов функционирования устройства: рабочего(основного), в котором функциональность исполняется, но модификации невозможны; специального и технологического, в которых исполняются только функции программирования узлов устройства. Переход от одного к другому режиму должен реализовываться физическими (контролируемыми), а не программными методами.

Теперь структура СЗИ приобретает реальные очертания СЗИ «Аккорд 4++», особенностями которого являются (рис. 2.8):

— модульная архитектура со спецканалами;
— программируемость всех узлов (ISP-технология);
— возможность расширения путем подключения дополнительного оборудования по стандартным каналам;
— функциональная достаточность резидентного ПО.

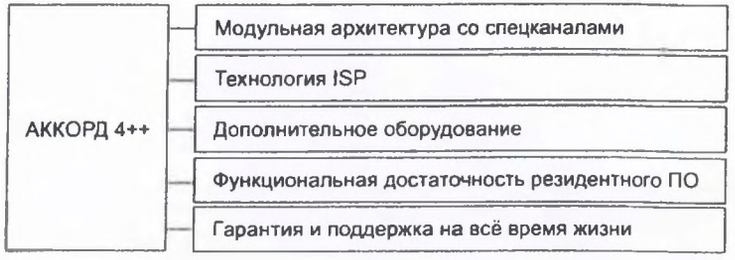


Рисунок 2.8

8.1. Контроллер «Аккорд 4++»

Контроллер «Аккорд» предназначен для работы в IBM PC совместимых компьютерах, имеющих слоты плат расширения ISA (XT или AT), в составе программно-аппаратных комплексов защиты информации от несанкционированного доступа

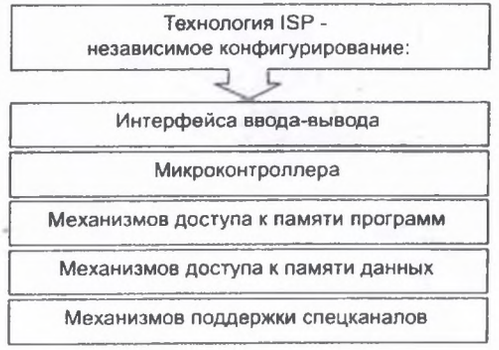


Рисунок 2.9

Плата контроллера выполнена как полностью программируемое устройство (In-System-Programmable - ISP), рис. 2.9. То есть функциональное назначение контроллера определяется программированием его составных частей (рис. 2.10):

— интерфейса шины (BusInt)
— постоянного запоминающего устройства типа Flash (ROM)
— микроконтроллера (MCU).

Программирование и модификация BusInt производится в *технологическом* режиме, а ROM и MCU — в *специальном*. Конструкция контроллера обеспечивает его работу в этих режимах только при вскрытии корпуса компьютера, извлечения платы из слота *motherboard* и отсоединений крепежного кронштейна, что служит защитой от проведения такой операции несанкционированно или случайно.

Контроллер, при подсоединении специального программатора, работающего под управлением IBM PC совместимого компьютера, переходит в *технологический* режим, который обеспечивает программирование интерфейса (рис. 2.11). Возможность перепрограммирования блокируется при установке платы в слот расширения ЭВМ и/или при установке крепежного кронштейна (см. рис. 2.10)

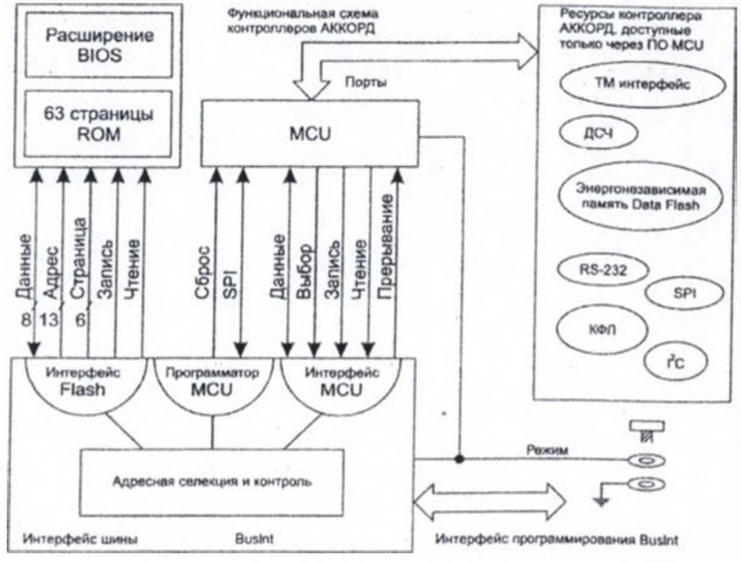


Рисунок 2.10

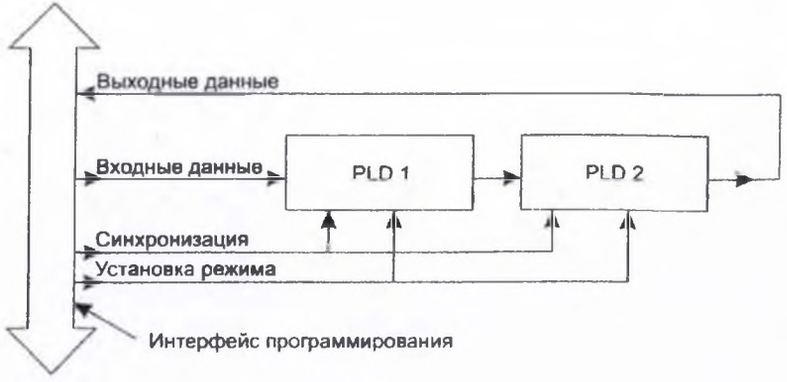


Рисунок 2.11

Интерфейс шины в технологическом режиме

Если же плата без крепежного кронштейна (при этом интерфейс шины уже должен быть запрограммирован) установлена в слот расширения компьютера, то она работает в *специальном* режиме, который обеспечивает программирование содержимого ROM и встроенного программного обеспечения (ПО) MCU (рис. 2.12) при помощи соответствующих утилит. Возможность перепрограммирования блокируется при установке крепежного кронштейна.

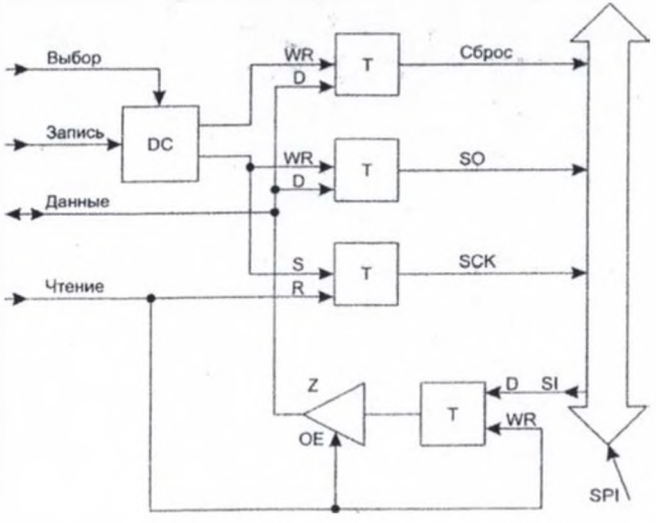


Рисунок 2.12

Интерфейс шины

Программатор микроконтроллера. Доступен только в *специальном* режиме.

После процедур ISP плата в сборе с крепежным кронштейном устанавливается в слот расширения и функционирует в основном режиме.

Отметим, что конструкция программируемых микросхем BusInt и MCU обеспечивает защиту от модификации и чтения записанной в них информации в любом режиме.

Интерфейс шины контроллера обеспечивает доступ центрального процессора ЭВМ к ROM и MCU (рис 2.13).

Интерфейс микроконтроллера в специальном ирабочем режимах

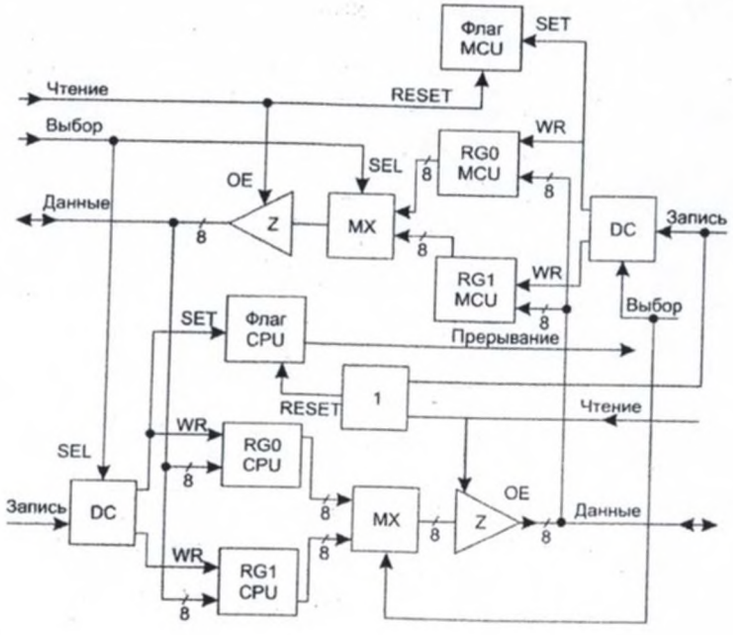


Рисунок 2.13

Адресная селекция и контроль

Для работы контроллера используется область верхней памяти ЭВМ (UMB) размером 16 Кбайт. Ее базовый адрес определяется состоянием джамперов на плате в соответствии с таблицей 2.1. Состояние 0 — замкнуто, 1 — разомкнуто.

Выбранная область разделяется на три окна — окно расширения BIOS (wBIOS), окно регистров управления контроллером (wREG) и окно страничного доступа (wFLASH). В окне wREG размещаются регистры.

Таблица 2.1
Установка джамперов для выбора базового адреса контроллера Доступ к ROM

J3	J2	J1	Базовый адрес
0	0	0	0xD0000
0	0	1	0xD0000
0	1	0	0xCC000
0	1	1	0xC8000
1	0	0	0xDC000
1	0	1	0xD8000
1	1	0	0xD4000
1	1	1	0xD0000

Одна из основных функций контроллера — выполнение кода расширения BIOS (ПО BIOS) во время процедуры POST до загрузки операционной системы. Обычно, для плат расширения этот код располагается в ПЗУ, отображаемом на часть верхней памяти ЭВМ (UMB), и его размер ограничен из- за наличия других плат и устройств — SYSTEM, VGA, SCSI, LAN и т.п. С другой стороны, объем кода определяет как функциональные возможности ПО, так и сложность его разработки. Для уменьшения размера области занимаемой контроллером в UMB и увеличения размера кода ПО BIOS используется следующий механизм.

Окно wBIOS обеспечивает инициализацию ПО BIOS во время процедуры POST , при этом через окно wFLASH предоставляется страничный доступ к большому объему данных. Другими словами, небольшая программа (до 8 Кб) из окна wBIOS считывает через окно wFLASH большую программу (до 500 Кб), размещает ее в памяти ЭВМ и передает ей управление. Такой способ позволяет принципиально расширить функции ПО BIOS, при этом дает возможность разрабатывать его на языках высокого уровня.

В качестве ROM используется микросхема объемом 512 Кбайт, который разделен на 64 страницы по 8 Кбайт каждая. Страница 0 доступна через окно wBIOS, а остальные — через окно wFLASH. При этом номер требуемой страницы предварительно записывается в 6-ти разрядный регистр номера страниц (rgPAGE). Запись в него производится по адресу

rgPAGE_WR, а чтение — rgPAGE_RD. При этом значащими являются 6 младших бит байта шины. В зависимости от режима и содержимого rgPAGE определяется тип доступа к окнам (рис. 2.14).

Отметим, что содержимое rgPAGE устанавливается в 0 при сбросе компьютера. Таким образом, код расширения BIOS во время процедуры POST в специальном режиме не исполняется, что важно при отладке ПО или если при его записи возникла ошибка.

Программирование ROM

Для программирования ROM контроллер должен работать в специальном режиме.

В качестве ROM используется микросхема Flash-памяти, которая программируется с помощью записи командных последовательностей. Эти последовательности формируются утилитой в соответствии с таблицей 2.2. Возможность перепрограммирования ROM блокируется при установке крепежного кронштейна.

Таблица 2.2.
Тип доступа к окнам ROM

Специальный режим

rgPAGE	wBIOS	wFLASH
= 0	denied	denied
> 0	R/W	R/W

Рабочий режим

rgPAGE	wBIOS	wFLASH
= 0	read-only	denied
> 0	read-only	read-only

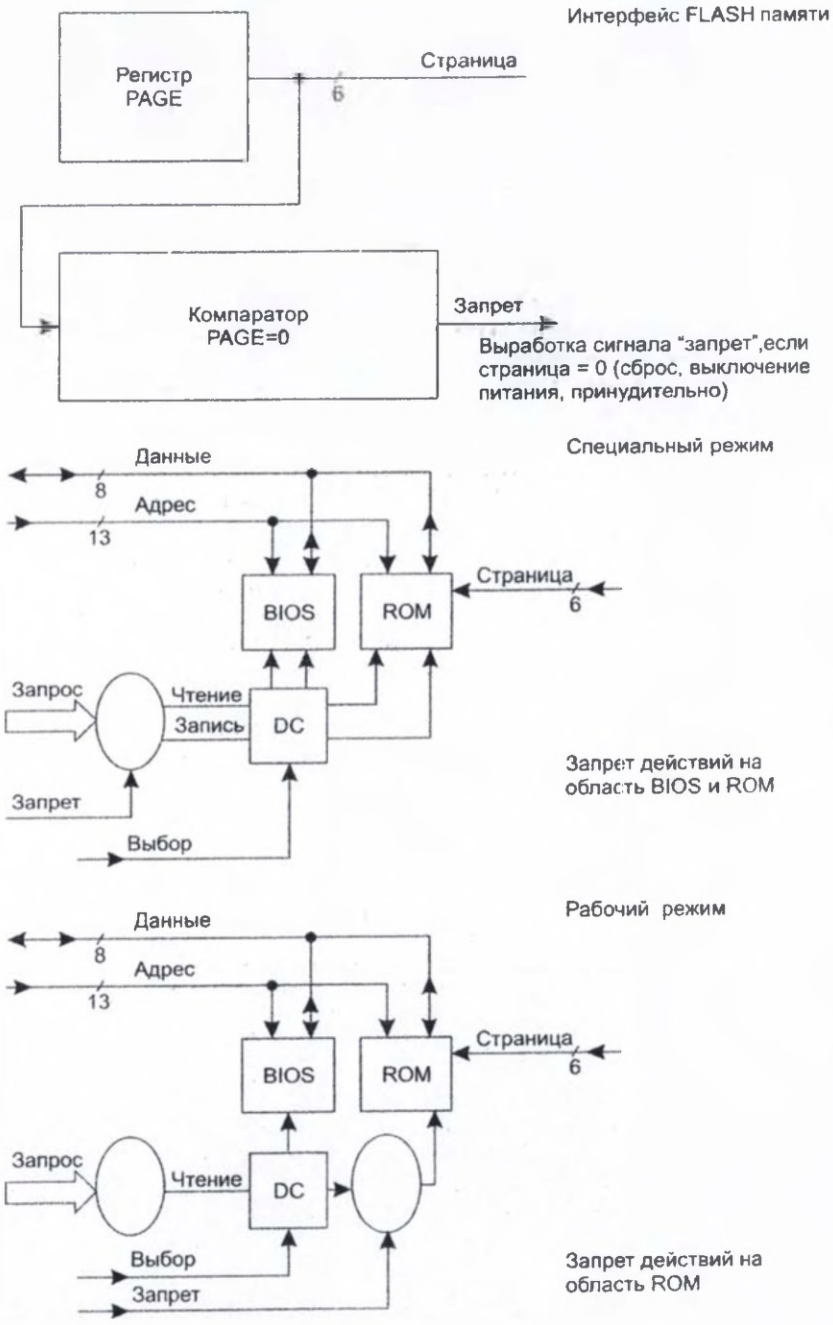


Рисунок 2.14

Интерфейс с MCU

MCU работает под управлением собственного встроенного ПО и выполняет функцию «электронного замка» для доступа пользователей к таким ресурсам как:

— вычислительные функции микроконтроллера;
— встроенная энергонезависимая память микроконтроллера (EEPROM);
— энергонезависимая память большого объема (DataFlash);
— внешний интерфейс (TM или RS232);
— датчик случайных чисел;
— возможность физического отключения устройств;
— расширение интерфейса SPI.

Поэтому для того, чтобы MCU выполнял свои функции, его необходимо запрограммировать.

Программирование MCU

Для программирования MCU контроллер должен работать в специальном режиме, а содержимое rgPAGE должно быть равно 0.

В качестве MCU используется микросхема, которая программируется с помощью командных последовательностей по интерфейсу SPI, при этом сигнал сброс на MCU должен быть установлен активным. Для управления сигналом сброса MCU и обмена по SPI используется только один бит из байта данных шины ISA, и два адреса в окне wBIOS. Специальная команда по выделенному адресу устанавливает сброс MCU и разрешает обмен по SPI. Интерфейс SPI обеспечивает синхронный последовательный обмен между устройствами — Master и Slave — используя протокол приведенный ниже.

Протокол SPI

Сигналы SPI показаны в таблице 2.3.

Таблица 2.3

Сигнал	Тип для Master	Тип для Slave
SCK	Выход	Вход
SO	Выход	Вход
SI	Вход	Выход

Обмен производится байтами старшим битом вперед. Один обмен производится за 8 циклов.

Цикл SPI

Устройство *Master*

1. Устанавливает сигнал SO - бит данных для устройства *Slave.*
2. Формирует передний фронт сигнала SCK.
3. Формирует задний фронт сигнала SCK и по нему принимает бит данных с линии SI.
4. Переходит к следующему циклу обмена.

Устройство *Slave*

1. Устанавливает сигнал SI - бит данных для устройства *Master.*
2. По переднему фронту SCK принимает бит данных с линии SO.
3. По заднему фронту SCK переходит к следующему циклу обмена.

При этом MCU является устройством *Slave.* Функцию устройства *Master* выполняет центральный процессор ЭВМ под управлением соответствующей утилиты. При записи по выделенному адресу в окне wBIOS формируется передний фронт сигнала SCK, а шестой бит байта данных шины ISA определяет значение сигнала SO. Чтение по этому адресу формирует задний фронт SCK, а шестой бит считанного байта определяется значением сигнала SI. Таким образом утилита формирует командные последовательности и производит обмен данными с MCU в режиме программирования.

Структура MCU содержит два программируемых блока — память программ и энергонезависимую память данных (EEPROM). Конструкция микросхемы MCU обеспечивает защиту от модификации и чтения в любом режиме как собственно его программы, так содержимого EEPROM. Другими словами, программа MCU и данные в EEPROM, как записанные пользователем в специальном режиме, так и сформированные в процессе функционирования контроллера в рабочем режиме, могут быть только уничтожены, но не изменены или считаны. И только после стирания содержимого памяти программ и EEPROM, MCU можно снова запрограммировать в *специальном* режиме. Возможность перепрограммирования MCU блокируется при установке крепежного кронштейна.

Работа MCU

После выхода из режима программирования, т. е. снятия сигнала сброс, MCU начинает работу, выполняя загруженную в него программу. Эта программа поддерживает выполнение команд, поступающих от центрального процессора машины. Для получения команд и обмена данными используются регистры rgCS и rgDATA для записи/чтения как ЭВМ, так и MCU, а rgPAGE_WR — только для чтения ЭВМ, которые обеспечивают интерфейс

Схема ПО MCU

Рисунок 2.15

При этом запись в rgCS вызывает прерывание MCU и взводит флаг запроса ЭВМ. MCU может передавать данные ЭВМ записывая байт в rgDATA и в rgCS. При этом запись в rgCS взводит флаг запроса MCU. Чтение регистров со стороны машины позволяет прочитать данные, записанные MCU, а чтение регистров со стороны MCU позволяет прочитать данные, записанные ЭВМ

Флаг запроса MCU снимается при чтении rgCS со стороны ЭВМ, а флаг запроса ЭВМ снимается при любом обращении к rgCS или rgDATA со стороны MCU.

Работа с дополнительными устройствами

Дополнительное оборудование подключается к спецканалам, которые, в свою очередь, управляются MCU (рис. 2.16.а). Рассмотрим кратко некоторые часто применяемые дополнительные устройства.

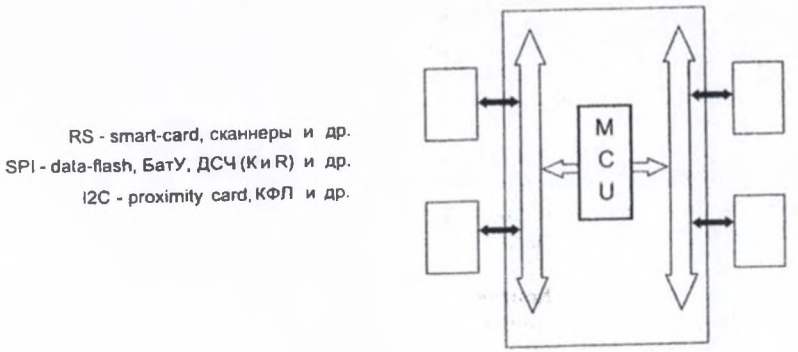


Рисунок 2.16.а

Рисунок 2.16.б

ДСЧ

Микроконтроллер по запросу CPU может сформировать байт случайного числа, получаемого с помощью шумовых диодов. Наличие у микроконтроллера аналогового компаратора позволяет использовать две схемы ДСЧ (спецификации R и K).

ТМ интерфейс/ RS-232

Микроконтроллер может поддерживать функцию однопроводного (One Wire) интерфейса, предназначенного для подключения устройств фирмы Dallas, или интерфейс RS-232 для подключения стандартных устройств.

Энергонезависимая память Data Flash

Микроконтроллер, через интерфейс SPI, подключен к энергонезависимой памяти (Serial EEPROM) объемом до 1М бит, что позволяет создавать пользовательские архивы.

Контроль физических линий

Микроконтроллер управляет двумя реле, которые могут быть использованы для физического подключения/отключения (коммутирования на землю или питание) управляющих цепей внешних устройств.

Батареечные устройства (БатУ)

Особым вариантом дополнительных устройств является семейство батареечных устройств (рис.2.16.б). Особенности этих устройств заключаются в том, что с их помощью можно организовать аудит тех событий, которые ранее были неконтролируемы. Так, например, БатУ позволяет в своей памяти (DF) фиксировать вскрытие корпуса и изъятие платы контроллера (в том числе для выключенного компьютера). Для этого служат блоки датчиков (I) и блоки управления (O). В памяти DF могут фиксироваться также события, связанные с аудитом администратора. Дело в том, что администратор по функциональным обязанностям обычно имеет доступ к журналам СЗИ, что может служить источником злоупотреблений. Если же факт операций с журналом и их тип фиксируется в DF БатУ, которая вообще закрыта от модификаций, то злоупотребления можно расследовать.

Возможности MCU БатУ позволяют также организовать аутентификацию устройств АС по схеме БатУ — «Аккорд» — ПЭВМ — АРМ АБИ и т.д.

Описание батарейного устройства (БатУ)

1. Состав.

БатУ представляет собой устройство на основе микроконтроллера (MCU) и имеет в своем составе резервную литиевую батарею, контроллер, осуществляющий переключение питания схемы с основного питания на резервное и наоборот, и вызывающий прерывание MCU при этих событиях. Кроме этого в состав БатУ входит интерфейсная схема, устраняющая дребезг контактов и вызывающая прерывание MCU при изменении состояния входов. Интерфейсная схема контролирует питание «Аккорда» в слоте PCI, режим «Аккорда» в слоте ISA, нахождение «Аккорда» в слоте PCI, режим работы «Аккорда» и состояние внешних цепей, работающих на замыкание-размыкание (КФЛ). Один из них может использоваться для контроля закрытого корпуса. Для хранения информации БатУ имеет энергонезависимую память. Обмен БатУ с «Аккордом» осуществляется по интерфейсу SPI.

2. Организация энергонезависимой памяти.

Память микросхемы объемом 1 Мбайт разбита на 2 части по 512 КБайт. Эта память организована по типу ADD-ONLY, т.е. в эту память можно только добавлять записи, а читать можно произвольный адрес записи.

Первая часть этой памяти представляет память событий объемом (64к-1) 8-ми байтных записей. Нулевая запись служебная, а по остальным адресам осуществляется последовательная запись текущих состояний входов интерфейсной схемы.

Структура записи: резервный байт, резервный байт, минута, час, день месяц, год, состояние входов.

Вторая часть памяти — это фискальная память по типу ADD-ONLY. Она также имеет (64к-1) 8-ми байтных записей произвольной структуры.

3. Основные принципы работы.

Основной цикл работы — это запись нового состояния в память событий. При снижении питания новое состояние записывается в ОЗУ микроконтроллера, а при восстановлении переписывается в память событий. Также БатУ выполняет команды, приходящие по SPI со стороны «Аккорда» (см. пункт 4). Предполагается, что при восстановлении питания осуществляется процедура HANDSHAKE (пункт 4.4.). После удачного выполнения процедуры HANDSHAKE нужная информация из БатУ переписывается в журнал «Аккорда».

При удачном выполнении этой процедуры БатУ выполняет все команды. При неудачном выполнении этой процедуры БатУ выполняет только процедуру HANDSHAKE (возможно ограничение повторов со стороны «Аккорда»).

4. Команды БатУ.

Команды посылаются по SPI, возврат OA5H или код ошибки. БатУ находится в состоянии SLAVE.

4.1. Произвольное чтение памяти состояний (RANDOM_READ)

Код = 81H

1-й байт младшая часть адреса записи (возврат 1).

2-й байт старшая часть адреса записи (возврат 2).

Количество записей (возврат 3).

После этого прием (кол-во записей × 8) байт.

4.2. Произвольное чтение DF (RD_DF)

Код = 86H.

1-й байт младшая часть адреса записи (возврат 1).

2-й байт старшая часть адреса записи (возврат 2).

Количество записей (возврат 3).

После этого прием (кол-во записей ×8) байт.

Примечание: при произвольном чтении все записи должны находиться на одной странице.

4.3. Запрос событий (REQ_SOST)

Прием:
1-й байт старшая часть адреса считанной записи
2-й байт младшая часть адреса считанной записи
3-й байт старшая часть адреса последней записи
4-й байт младшая часть адреса последней записи

4.4. HANDSHAKE (HAND_SHAKE)

Код = 83H.

Прием: 4 байта.

Возврат либо OA5H либо код ошибки.

Примечание: по нулевым адресам находится служебная информация.

Тип операций определяется выбранным методом аутентификации.

4.5. Установка времени (SETUP_TIME)

Код = 84H.

Передача: 7 байта: год, месяц, день, час, минута, секунда, 1/100 сек.

Год кодируется двоичным числом YEAR-1900, остальные параметры времени и даты кодируются двоично-десятичными числами.

4.6. Обнуление адресов (RST_DF)

Код = 85H.

Адрес последней записи в DF, адрес последней считанной записи в памяти состояний, и адрес последней записи в память состояний обнуляются.

Примечание: по нулевым адресам находится служебная информация.

4.7. Запись в DF (WR_DF)

Код = 87H.

Передача: 1-й байт: какое количество записей будет послано (N), затем посылается N × 8 байт.

4.8. Чтение SPD (TST_SPD) - служебный

Код = 88H.

Прием: 2 байта — значение SPD (указатель стека данных)

4.9. Нейтральная команда

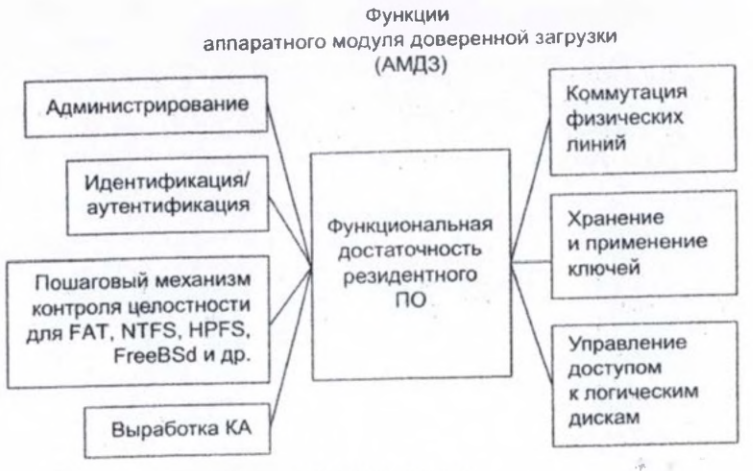
Код = 0

Служит для получения результата выполнения предыдущей команды.

На базе контроллера «Аккорд 4++» за счет функциональной полноты резидентного ПО можно разработать аппаратный модуль доверенной загрузки (АМДЗ) как средство, обеспечивающее доверенную загрузку ОС вне зависимости от ее типа для аутентифицированного защитным механизмом пользователя (рис. 2.17).

Внешний вид контроллера «Аккорд 4++» показан на цветной иллюстрации Ж.

Развитием ряда аппаратных средств СЗИ семейства «Аккорд» является сопроцессор безопасности «Аккорд СБ». Его структура показана на рис. 2.18.



АМДЗ - аппаратное средство, обеспечивающее доверенную загрузку ОС для аутентифицированного пользователя вне зависимости от типа ОС

Рисунок 2.17

Рисунок 2.18

8.2. Ресурсы «Аккорд СБ»

Для быстрого выполнения алгоритмов, требующих большого числа сложных вычислений, пользователю контроллера предоставляется ресурс процессора цифровой обработки сигналов (DSP), в качестве которого используется микросхема ADSP-2181 (см. рис. 2.19).

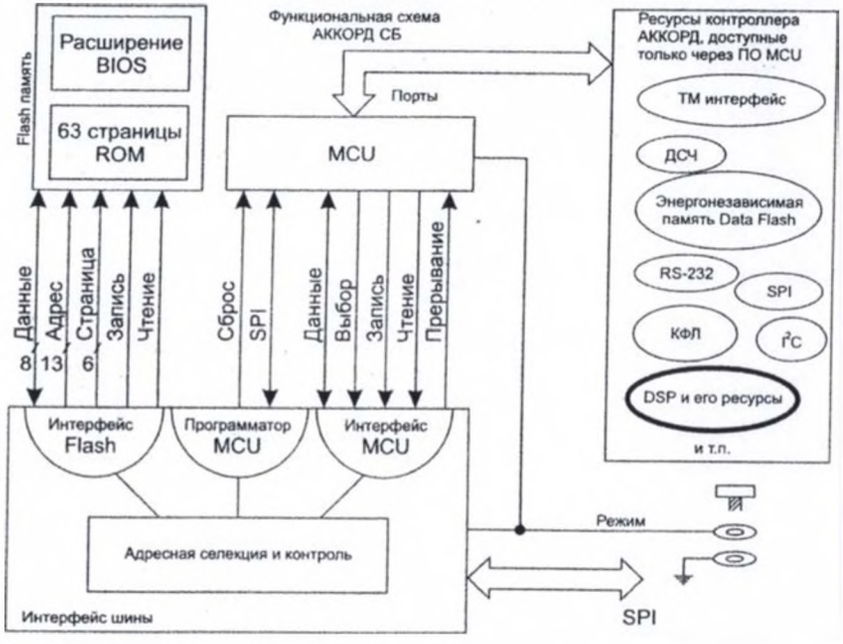


Рисунок 2.19

Технологический режим «Аккорда СБ»

Загрузка ПО производится в технологическом режиме (см. рис. 2.20).

КИ — контроллер интерфейса

АРМ Т — компьютер, специализированное устройство связи, ПО

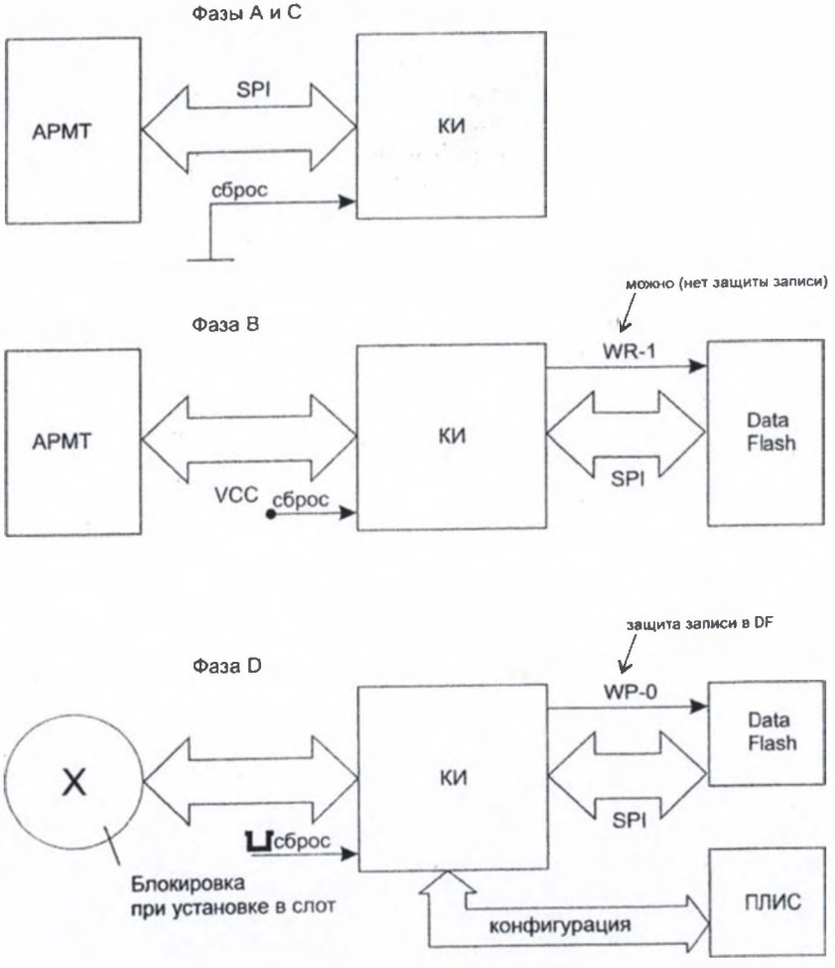


Рисунок 2.20

Интерфейс шины «Аккорд СБ» в технологическом режиме

Режимы специальный и обычный не отличаются от аналогичного режима контроллера «Аккорд 4++»

Фаза A — загрузка ПО для записи Data Flash.

Фаза B — запись в Data Flash файлов конфигурации интерфейса для 4-х режимов:
ISA специальный
ISA рабочий
PCI специальный
PCI рабочий

Фаза C — загрузка ПО для конфигурации интерфейса в соответствии с режимом работы.

Фаза D — конфигурация интерфейса кратковременно (200 mS). По включению питания. Производится определение типа шины ISA или PCI, определяется режим платы (специальный или рабочий), выбирается соответствующий файл из Data Flash и производится конфигурация ПЛИС. В случае ошибки процесс повторяется, а при правильном завершении контроллер интерфейса обеспечивает только блокировку записи в Data Flash и переконфигурацииПЛИС.

DSP инициализируется по команде машины для «Аккорда». Параметр этой команды указывает базовый адрес расположения трех портов в области ввода-вывода ЭВМ в соответствии с таблицей 2.4.

Таблица 2.4

Регистр	Смещение
rgADR_DSP	0
rgDATA_DSP	2
rgCS_DSP	4

Для доступа к внутренней памяти DSP по интерфейсу IDMA используются регистры rgADR_DSP и rgDATA_DSP. Причем rgADR_DSP, который служит для указания начального адреса обмена по IDMA, доступен только по

записи. Установив этот адрес, через rgDATA_DSP производится последовательная запись или чтение слов с автоматическим инкрементом адреса обмена. Для управления DSP используется регистр rgCS_DSP, который имеет только один младший значащий бит - флаг готовности DSP.
Запись 0 в rgCS_DSP сбрасывает этот флаг и вызывает прерывание DSP - IRQ2. Установить этот флаг можно только со стороны DSP — сформировав передний фронт на выходе шестого бита порта флагов (PF) — при его работе под управлением загруженного ПО. Запись 1 в rgCS_DSP, которая возможна только в *специальном* режиме, переводит DSP в режим сброса (рис 2.21).

Инициализация ресурсов DSP в специальном режиме

Блокируется интерфейс MCU-DSP. Разрешена работа интерфейса CPU-DSP доступ к областям DM и PM (память данных и память программ).

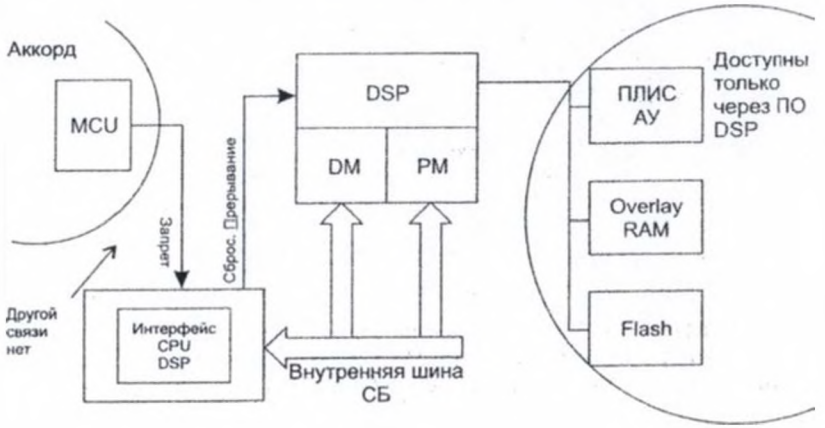


Рисунок 2.21

Инициализация DSP производится при помощи загрузки его ПО по каналу прямого доступа (IDMA). В основном режиме работы контроллера загрузка ПО DSP производится MCU, а в *специальном* эту операцию производит пользователь через интерфейс шины. Только в *специальном* режиме работы контроллера интерфейс шины разрешает пользователю обращаться ко всей

пользователь через интерфейс шины. Только в *специальном* режиме работы контроллера интерфейс шины разрешает пользователю обращаться ко всей внутренней памяти DSP, так же как и управлять сигналом его сброса. В *рабочем* режиме доступ к памяти DSP ограничен, а сигнал сброса недоступен, что обеспечивает целостность ПО DSP (рис 2.22).

Инициализация ресурсов DSP в рабочем режиме

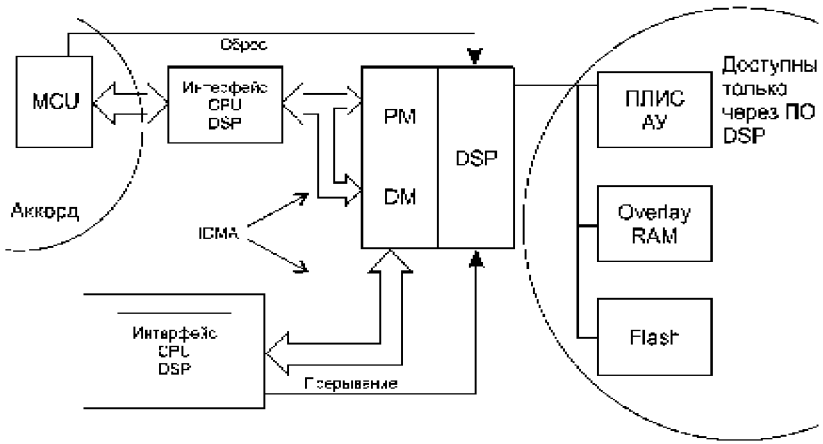


Рисунок 2.22

Интерфейс DSP разрешает доступ только к части области памяти DM (определенной ПО, загруженной в DSP из MCU), причем доступ может блокироваться ПО DSP и MCU. Схема ПО DSP аналогична приведенной на рис. 2.15.

Фаза A — интерфейс CPU-DSP заблокирован. Передача данных и ПО для конфигурации ПЛИС Аппаратного ускорителя из Data Flash.
Фаза B — запуск ПО и ожидание окончания операции.
Фаза C — передача данных и ПО для реализации системы команд DSP (функциональное).
Фаза D — запуск ПО, ожидание готовности, разрешение работы по интерфейсу DSP.

Внешний вид контроллера «Аккорд СБ» показан на цветной иллюстрации **З**.

8.3. Блок установки кодов аутентификации («БУКА»)

Как отмечалось, развитые версии контроллеров СЗИ имеют встроенные механизмы безопасной работы с ключами и выработки КА. Тем не менее, в ранних версиях СЗИ такие возможности не предусматривались. Это привело к необходимости разработки простого устройства, которое может дополнить СЗИ НСД функциями работы с ключами и установки кодов аутентификации. Было решено подключать такое устройство через имеющийся на ПЭВМ LTP-порт, причем в прозрачном режиме. БУКА содержит MCU с соответствующим ПО (ГОСТ 28147-89 — 30КБ/сек, ГОСТ 34.11-94 — 17 КБ/сек), опционно-ТМ интерфейс и аппаратный датчик случайных чисел.

Внешний вид «БУКА» показан на цветной иллюстрации **И**.

8.4. Специальные режимы контроллеров

8.4.1. Режимы «Аккорд СБ»

Контроллер «Аккорд СБ» имеет три режима работы: технологический, специальный и рабочий. В процессе функционирования системы по выработке/проверке КА используются два из них: специальный и рабочий.

Нормальный режим работы контроллера — рабочий. Для того чтобы перевести контроллер в спецрежим, необходимо отсоединить от него крепежную планку.

Последовательность действий.

1. Выключить компьютер.
2. Вынуть контроллер из компьютера.
3. Отвернув винты крепления, снять планку с контроллера.
4. Установить контроллер в свободный слот компьютера.
5. Включить компьютер.

После этого контроллер находится в специальном режиме. Для того чтобы перевести контроллер в рабочий режим, необходима обратная последовательность действий.

1. Выключить компьютер.
2. Вынуть контроллер из компьютера.
3. Установить планку на контроллер, завернув винты крепления
4. Установить контроллер в свободный слот компьютера.
5. Включить компьютер.

На рисунке 2.23 приведены основные компоненты контроллера «Аккорд СБ». Процессор MCU (поз.1) обеспечивает функции по работе с

Touch Memory. EEPROM в его составе используется для хранения ключевой информации. Особенностью EEPROM является то, что извне процессора она доступна только для записи. Процессор DSP (поз. 4) используется для работы с кодами аутентификации. Память DSP может быть закрыта от доступа извне при проведении работы с конфиденциальными данными. Data Flash (поз. 5), доступ к которой осуществляется только DSP, используется для хранения таблиц достоверности(ТД).

Взаимодействие с шиной компьютера ведется через ПЛМ Altera (поз. 3) под управлением соответствующего процессора. Причем программы обоих процессоров написаны так, что состав их функций исключает возможность получение секретной информации, хранящейся в контроллере.

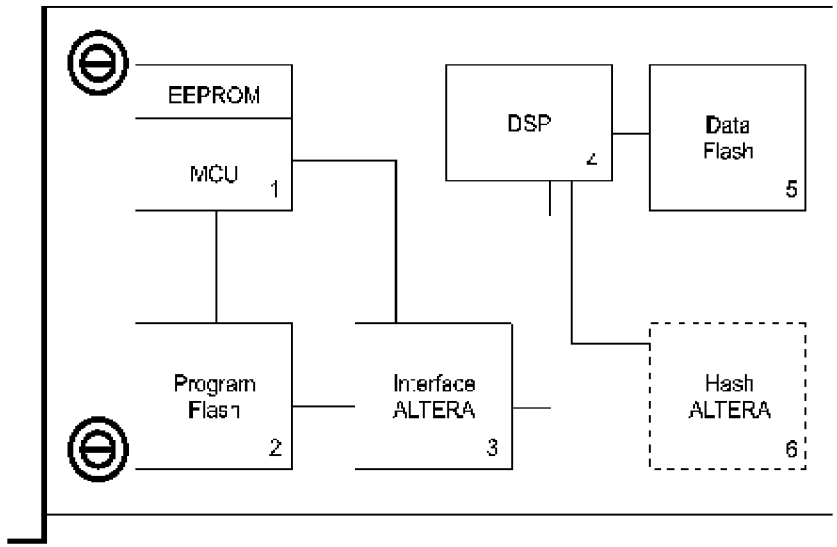


Рисунок 2.23

Специальный режим «Аккорд СБ»

В специальном режиме контроллера программа DSP не загружается. Программа MCU определяет состояние переключателя режима и позволяет запись в EEPROM. Режим используется для загрузки начальной информации в контроллер в процессе процедуры персонализации.

Рабочий режим «Аккорд СБ»

В рабочем режиме контроллера программа MCU реализует функции работы с Touch Memory и инициализации DSP. В DSP загружается программа для работы с КА и ТД. Функциональный набор программы MCU исключает возможность прочтения и модификации ключевой информации в EEPROM. Функциональный набор программы DSP позволяет загрузку ключевой информации для КА (таблицы достоверности), не позволяя ее прочтения. Это основной режим работы контроллера.

8.4.2. Режимы «Аккорд 4++»

«Аккорд 4++» оснащен MCU, отвечающим за выполнение процедур для работы с КА. Контроллер имеет три режима функционирования: технологический, специальный и рабочий. В процессе функционирования системы по выработке/проверке КА используются два из них: специальный и рабочий.

Переход из одного режима в другой выполняется аналогично «Аккорд СБ». Программа MCU определяет состояние переключателя и разрешает, либо запрещает запись ключевой информации в EEPROM.

Специальный режим «Аккорд 4++»

В специальном режиме программа MCU позволяет запись ключевой информации в EEPROM. Режим используется для проведения процедуры персонализации.

Рабочий режим «Аккорд 4++»

В рабочем режиме в программа MCU обеспечивает выполнение функций работы с КА и таблицами достоверности, но запрещает модификацию ключей доставки/восстановления. Работа с ТД сводится к загрузке и использованию ее при выработке КА, но не выдаче содержимого.

8.4.3. Режимы «БУКА»

«БУКА» имеет два режима функционирования: технологический и рабочий. Переход из одного режима в другой осуществляется при выключенном питании путем снятия/установки переключателя на плате.

Технологический режим «БУКА»

В технологическом режиме возможен полный доступ к EEPROM и ROM MCU. В этом режиме возможна запись/чтение программы и данных. Однако после завершения записи и установки бита защиты процессора доступ к его областях данных возможен только после их очистки. Режим используется для технологической инициализации «БУКА» программой MCU и проведения персонализации.

Рабочий режим «БУКА»

Основной режим функционирования. Программа MCU осуществляет

функции вычисления/проверки КА, загрузки таблиц, диагностирования. Доступа к ROM и EEPROM по чтению извне нет.

Режимы работы всех перечисленных контроллеров позволяют запись, но не чтение извне основной ключевой информации (ключей доставки/восстановления) в специальном режиме и запрещают доступ извне к этой информации в рабочем режиме. Переход из одного режима в другой осуществляется путем физического, а не программного воздействия. То есть для обеспечения целостности ключевой информации контроллеров необходимо поддержание их в рабочем режиме, что может быть достигнуто организационными мерами. Конфиденциальность ее обеспечивается конструктивно.

8.5. Выполнение контроллерами основных процедур

8.5.1. Процедура персонализации

Служит для инициализации контроллеров для работы в системе кодов аутентификации. Во время процедуры персонализации в контроллеры записывается ключевая информация и уникальный номер контроллера в системе. Под ключевой информацией понимается основной ключ доставки/восстановления Kb и резервный ключ доставки/восстановления Kr.

Производится на выделенном ПК - АРМ персонализации (АРМ П). АРМ П — это отдельный ПК с установленным на нем специальным ПО, служащим для персонализации. Контроллер переводится в специальный режим и устанавливается в компьютер. ПО персонализации загружает ключевую информацию, поступающую от АРМ АР на внешних носителях.

Персонализация «Аккорд СБ» и «Аккорд 4++»

Так как «Аккорд СБ» и «Аккорд 4++» оснащены встроенным контроллером Touch Memory, ключевая информация записывается в EEPROM с внешних ТМ, не выходя за пределы контроллера и не поступает в ПК. Работа ПО персонализации сводится к осуществлению интерфейса и вызову команды контроллера «персонализация».

Информация от АРМ АР поступает в открытом виде, поэтому конфиденциальность ее должна обеспечиваться на этапе установки организационными мерами.

Персонализация «БУКА»

«БУКА» не имеет встроенного ТМ-контроллера, поэтому для проведения персонализации на АРМ П должен устанавливаться какой-либо внешний. Вся ответственность по записи программы для MCU и ключевой информации при этом ложится на ПО персонализации, работающее на ПК. При этом сама программа поступает из дискового файла, а данные персонализации через ТМ контроллер.

8.5.2. Загрузка ТД

После проведения процедуры персонализации контроллеры, участвующие в системе имеют возможность обмениваться информацией с АРМ АР, закрывая данные с использованием ключей доставки/восстановления — K_b и K_r.

При закрытии ТД (T) используется сеансовый ключ K_s, получаемый на АРМ АР с использованием генератора случайных чисел. Таблица достоверности в файле загрузки (T_f) зашифровывается на этом ключе.

$$T_f = \gamma(T, K_s),$$

где γ — криптографическое преобразование гаммирования по ГОСТ 28147-89.

Сам сеансовый ключ, зашифровывается прямой заменой по ГОСТ 28147-89 на основном ключе доставки/восстановления K_b и помещается в файл загрузки.

$$K_{sf} = \varphi(K_s, K_b)$$

Загрузка ТД контроллером «Аккорд СБ»

Таблица достоверности при загрузке расшифровывается процессором контроллера DSP (рис. 2.23 поз. 4) в собственной памяти, недоступной со стороны ПК. Ключ для расшифрования (K_b) берется из EEPROM MCU (рис. 2.23 поз. 1) и передается в оперативную память DSP при инициализации.

DSP расшифровывает сеансовый ключ из файла загрузки прямой заменой по ГОСТ 28147-89:

$$K_s = j^{-1}(K_{sf}, K_b)$$

Затем расшифровывает саму таблицу гаммированием по ГОСТ 28147-89 на сеансовом ключе K_s

$$T = j^{-1}(T_f, K_s)$$

Расшифрованная ТД записывается в DataFlash (рис. 2.23 поз. 5), доступную только DSP. DataFlash имеет размер до 2Мбайт, что позволяет хранить большой объем ключевой информации. Перед записью ТД планируется перешифровка ее на собственном ключе контроллера (K), генерируемым в процессе персонализации контроллером MCU. Этот ключ вырабатывается и хранится внутри котроллера и никогда не используется и

не появляется вне его.

$$T_k = \gamma(T, K)$$

Загрузка ТД контроллерами «Аккорд 4++» и «БУКА»

Загрузка ТД и расшифрование ее производится MCU во внутренней памяти, недоступной со стороны ПК. Из EEPROM извлекается K_b, с помощью которого производится открытие сеансового ключа. Затем расшифровывается ТД и в открытом виде записывается в EEPROM.

Таким образом, ни ключи доставки/восстановления, ни КОПы (содержимое ТД) в открытом виде не появляются ни на шине ПК, ни на контактах самого процессора. Размер EEPROM ограничен 512 байтами, что, соответственно, ограничивает размер загружаемой информации. Часть памяти занята также ключами доставки/восстановления и служебной информацией. При длине КОП 16 байт и их количестве 16 размер ТД составит 256 байт, что позволяет обеспечить работу контроллеров по выработке персональных КА.

8.5.3. Выработка кода аутентификации

Выработка кода аутентификации «Аккорд СБ»

Производится процессором DSP в собственной памяти, недоступной со стороны ПК. От загруженных данных (D) для получения КА вычисляется хэш-функция по ГОСТ Р 34.11-94:

$$H = h(D)$$

Затем из DataFlash извлекается КОП (C_o) и при необходимости расшифровывается на собственном ключе K контроллера.

$$C_o = g^{-1}(C_o, K)$$

Ключ K и номер контроллера в системе (N_k) поступают от MCU при инициализации. Организация ТД позволяет производить расшифрование отдельных КОП без расшифрования всей таблицы.

Затем вычисляется хэш-функция от Н, номера контроллера и КОП:

$$H_{co} = h((H, N_k, C_o))$$

Используя часть полученного хэш, составляется КА:

$$C_a = (H_{co}, N_T, N_o, N_k)$$

где C_a — код аутентификации
N_T — номер таблицы достоверности,
N_o — номер операции, заданный в вызове
N_k — номер контроллера в системе

Взаимодействие при выработке КА иллюстрируется на рис. 2.24.

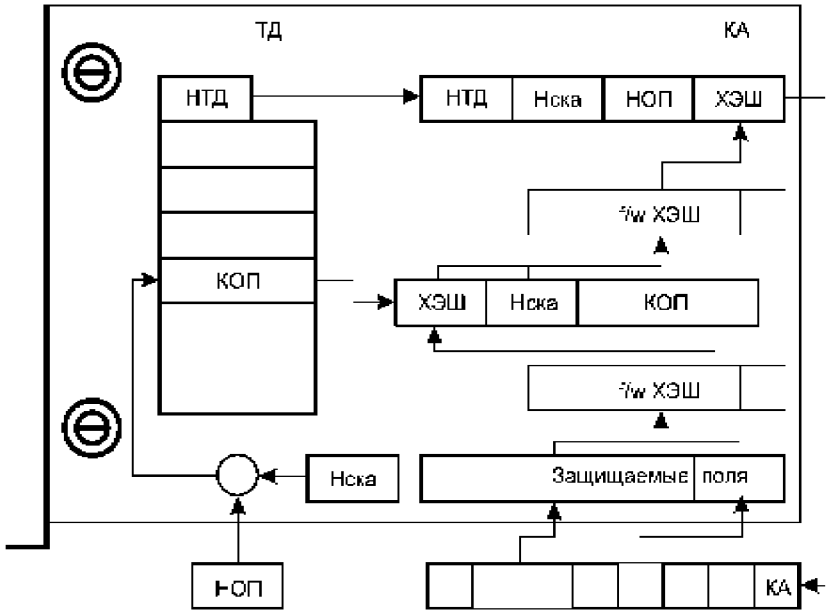


Рисунок 2.24

При выработке КА MCU может использовать HashAltera (рис.2.23 поз. 6), которая обеспечивает аппаратное вычисление хэш-функции. Это повышает производительность выработки/проверки КА.

Для дополнительного увеличения производительности этих операций предусмотрена возможность вычисления H вне контроллера. Такой

режим не снижает безопасности, т.к. работа с конфиденциальной информацией (КОП и К) по прежнему происходит внутри контроллера.

Схематично это показано на рис. 2.25.

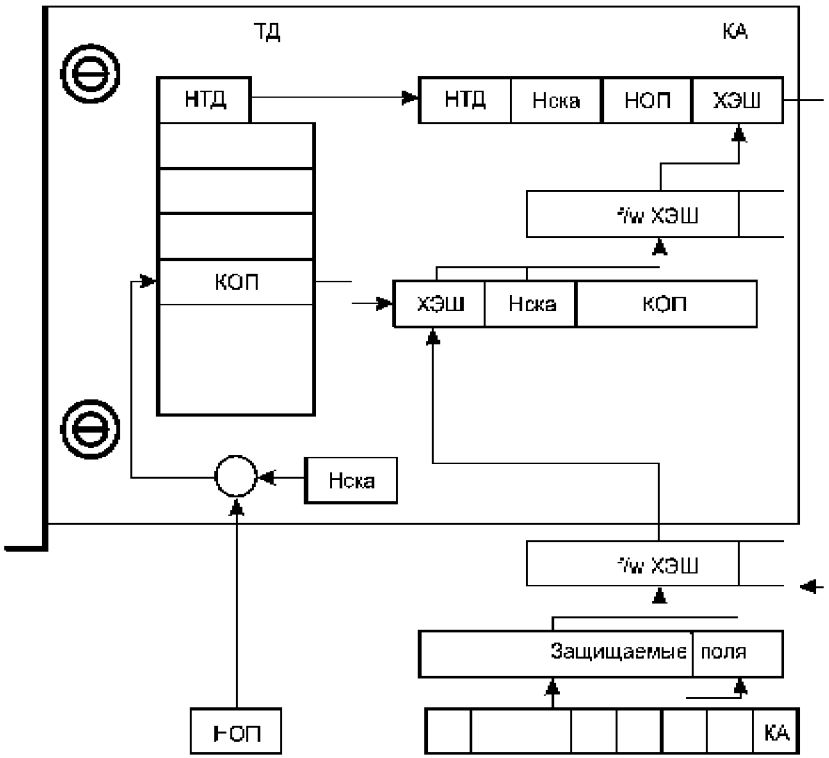


Рисунок 2.25

Выработка КА «Аккорд 4++» и «БУКА»

Производится MCU в собственной памяти, недоступной со стороны ПК. Последовательность операций сходна с «Аккорд СБ». В начале вычисляется хэш-функция данных Н. КОП извлекается процессором из EEPROM в открытом виде. Затем вычисляется хэш-функция и составляется КА.

Хотя «Аккорд 4++» и «БУКА» допускают вычисление хэш данных вне контроллера, этот режим не предполагается использовать, т.к. поток данных на станциях, оснащаемых перечисленными контроллерами, полагается низким.

8.5.4. Проверка кода аутентификации

Производится только «Аккорд СБ», т.к. остальные контроллеры используют для хранения ТД EEPROM MCU, объем которого недостаточен для записи информации обо всех КОП системы.

Проверка производится путем перевычисления КА, для данных. На вход контроллера поступают данные и соответствующий им КА. Информация, содержащаяся в КА (номер ТД, номер операции и номер контроллера в системе), позволяет DSP выбрать из DataFlash КОП, с использованием которого был ранее вычислен КА. Затем по для этого КОП вычисляется Нсо, который сравнивается с записанным во входном КА.

Схематично это изображено на рисунке 2.26.

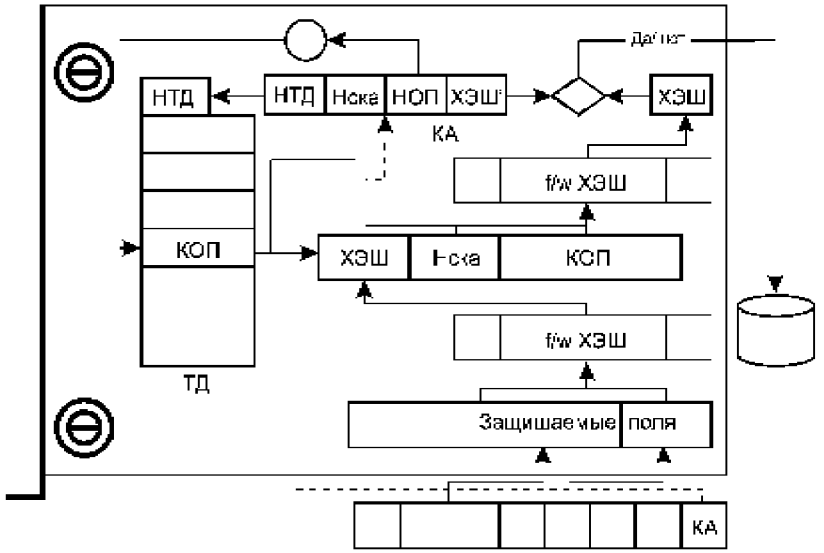


Рисунок 2.26

Все вычисления, связанные с использованием секретной информации, производятся в недоступной ПК памяти DSP. Как и в случае выработки КА, несекретная часть вычислений может быть вынесена за преде-

лы контроллера, для увеличения его производительности, что показано на рис. 2.27.

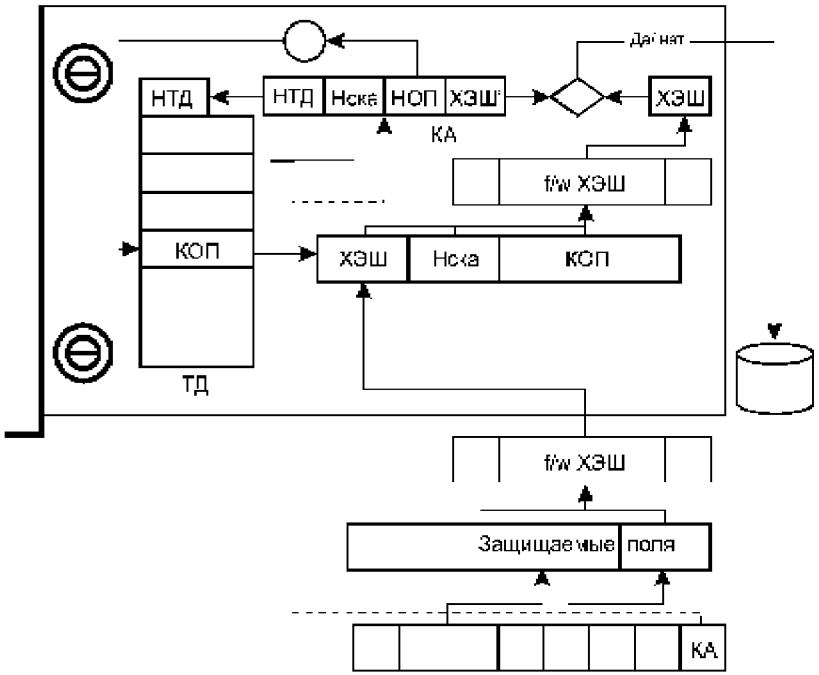


Рисунок 2.27

8.5.5. Функции тестирования

Контроллеры «Аккорд СБ», «Аккорд 4++» и «БУКА» имеют функции самотестирования. Функции выполняют ряд внутренних тестов с выдачей результатов об успехе/сбое.

Таким образом, функциональный набор контроллеров «Аккорд СБ», «Аккорд 4++» и «БУКА» для работы с КА позволяет загружать, модифицировать и использовать ключевую информацию, но не выдавать ее наружу. Ни в специальном, ни в рабочем режимах работы контроллеров программы MCU и DSP не предоставляют интерфейс, позволяющий прочтение на шине ПК секретных данных. Конструктивное решение контроллеров также не позволяет получить доступ к содержимому EEPROM и DataFlash.

9. ЗАКЛЮЧЕНИЕ

Из сказанного выше следует, что СЗИ может быть создана на базе не любого, а только такого контроллера, который обладает некоторыми минимальными ресурсами, а именно:

— содержит интерфейс к идентификатору с памятью на чтение/запись;

— содержит энергонезависимую память;

— содержит датчик случайных чисел;

— содержит ПЗУ пользовательского расширения BIOS, в котором зафиксированны следующие процедуры (или их аутентифицирующие коды):

— блокировка загрузки ОС с отчуждаемых носителей,

— процедуры идентификации (аутентификации),

— процедуры разбора файловой системы,

— процедуры расчета хэш-функции,

— процедуры работы с энергонезависимой памятью и ДСЧ.

Отметим еще раз, что эти ресурсы — минимально необходимые. Именно в этой конфигурации и был впервые описан комплекс «Аккорд» [6]. Началась эра аппаратной защиты.

Выбор системы защиты информации от несанкционированного доступа должен основываться на анализе требований, предъявляемых как к составу функциональных параметров, так и к параметрам производства и

сопровождения изделий, а также к эксплуатационным характеристикам. На наш взгляд, СЗИ НСД должна отвечать следующим требованиям.

1. Иметь сертификат в системе сертификации средств защиты информации для класса не ниже 1В, и выпускаться на основании лицензии органов, имеющих федеральные полномочия в указанной сфере. Производство технических и программных средств СЗИ должно быть аттестовано и подвергаться периодическому контролю.

2. Функциональные возможности СЗИ должны обеспечивать выполнение основных контрольных процедур до загрузки операционной системы, т.е. на аппаратном уровне. При этом контроль целостности системных областей и файлов, данных и процедур должен осуществляться устойчивым к воздействиям механизмом, основанным на применении хэш-функций. На базе этих средств, а также контроля запуска задач должна обеспечиваться корректная поддержка изолированности программной среды.

3. Спектр выпускаемых на аттестованном производстве СЗИ должен перекрывать проблемы защиты в гетерогенных сетях, основанных на интеграции наиболее популярных ОС, в том числе MS DOS, Novell Net Ware, Windows 3.11, Windows 95, Windows NT. Корректная работа в данных средах может основываться только на принципах операционной независимости программных средств СЗИ и транспортного механизма, не зависящего от типа сетевой ОС.

4. Состав атрибутов, на основе которых описываются правила разграничения доступа (ПРД) к объектам информационной системы должен быть таким, чтобы обеспечить возможность описания любой разумной непротиворечивой политики безопасности. При этом СЗИ не должна создавать трудностей для пользователей системы — не ограничивать функциональных возможностей, предоставляемых операционной системой, быть «прозрачной» в пределах политики безопасности для легального пользователя, аутентифицированного защитным механизмом.

5. Для применения в сетевых решениях обязательным является наличие средств централизованного управления безопасностью и средств аудита, в том числе в режиме on line. Средства аутентификации при этом должны обеспечивать не только аутентификацию операторов, но и технических средств комплекса.

6. Обеспечивая целостность и доступность информации, защиту её от несанкционированных модификаций, СЗИ должна предоставлять возможность работы с сертифицированными криптографическими средствами. Важным при этом является отсутствие скрытых (не документированных) возможностей, а также «опасных» реакций на действия операторов.

7. Одним из важных критериев выбора СЗИ является практика применения данной СЗИ в крупных системах.

Анализ показал, что полной мере всем упомянутым требованиям удовлетворяет только СЗИ «Аккорд». В настоящее время эксплуатируется уже более 45 000 систем защиты «Аккорд», в том числе в автоматизированных системах ЦБ РФ, СБ РФ, сотнях коммерческих банках и финансовых структур, Пенсионном Фонде России, Государственном Таможенном Комитете России, Федеральной пограничной службе и др. С использованием СЗИ «Аккорд» выпускаются СКЗИ «Верба», «АПДС», что отражено в сертификатах ФАПСИ на эти изделия. СЗИ «Аккорд» обеспечивает необходимый уровень защиты в системах вычислительной техники по классу 1В, что подтверждается соответствующими сертификатами. Выпуск СЗИ «Аккорд» осуществляется на основании лицензии на аттестованном про-

изводстве.

Литература

1. «Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации». Руководящий документ. Москва, Гостехкомиссия России, 1992.

2. «Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности СВТ». Руководящий документ. Москва, Гостехкомиссия России, 1992.

3. Хоффман Л. Дж. «Современные методы защиты информации», Москва, 1980.

4. В.А. Трахтенброт «Алгоритмы и вычислительные автоматы», Москва, «Советское радио», 1974.

5. «Средства вычислительной техники. Межсетевые экраны. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации». Руководящий документ. Москва, Гостехкомиссия, 1997.

6. Патент № 2067313 на изобретение «Устройство защиты от несанкционированного доступа к информации, хранимой в персональной ЭВМ». Москва, Роспатент, 1996.

Глава 3.
СЗИ НСД «АККОРД» И УПРАВЛЕНИЕ ЗАЩИТОЙ ИНФОРМАЦИИ НА ЕГО ОСНОВЕ

1. ВВЕДЕНИЕ

Сегодня некоторые принципы аппаратной защиты уже стали фактическим стандартом, и применяются практически всеми разработчиками СЗИ НСД. Другие, более сложные идеи, еще только пробивают себе дорогу. Наиболее последовательно и полно разработанные методы и механизмы аппаратной защиты реализованы в комплексах семейства «Аккорд».

СЗИ НСД «Аккорд» разработаны и выпускаются ОКБ САПР с 1995 года, и сегодня уже эксплуатируются десятки тысяч изделий.

Первым из семейства был разработан «Аккорд 1», который простотой, надежностью и эффективностью завоевал широкое признание, хотя, по нынешним меркам, представляется очень несложным, и даже наивным. Затем были контроллеры «Аккорд 1.5» со внутренней энергонезависимосимой памятью (ЭНП), «Аккорд 2» с ЭНП и датчиком случайных чисел (ДСЧ), «Аккорд 4», «Аккорд 4+» и «Аккорд 4++» на новой элементной базе, включающие возможности управления физическими линиями, подключения батареечных устройств как средства контроля администратора, и ряд других необходимых сегодня опций.

Параллельно развивался ряд программных продуктов:

v 1.31 — для DOS;
v 1.35 — для DOS, WINDOWS 3.x;
v 1.95 — для DOS, WINDOWS 9.x;
v 2.x — для WINDOWS NT.

Наиболее интересным сегодня является вариант СЗИ НСД, состоящий из контроллера «Аккорд 4++» с резидентным программным обеспечением v 2.0x, и специальным программным обеспечением, перечисленным выше.

Аппаратная часть контроллера «Аккорд 4++» выполнена по технологии ISP, т.е. не только встроенное ПО, но и логика работы микропроцессора могут перезаписываться в спецрежиме без замены платы контроллера.

Расширена область памяти для встроенного ПО, списка пользователей и журнала событий. Теперь в контроллере хранится описание ПРД пользователей ко всем ресурсам компьютера, *в том числе к файлам и каталогам на жестком диске.*

На плате установлен интерфейс для считывателя смарт-карт.

Предусмотрена установка до *128 Mb быстрой флэш-памяти.* Администратор может инсталлировать Windows 95/98 на флэш-диск и получить ОС, целостность которой гарантирована и постоянно контролируется в процессе работы!

На плату контроллера может устанавливаться дополнительное устройство с встроенным источником питания.

Данное устройство фиксирует:

— вскрытие корпуса компьютера;

— извлечение контроллера;

— очистку журнала событий администратором;

— установку контроллера;

— закрытие корпуса компьютера.

Данное устройство ведет собственный энергонезависимый журнал высокого уровня.

Контроллер содержит Power Manager, позволяющий корректно фиксировать события, связанные с отключением питания.

В версии «Аккорд 4++/КА» контроллер позволяет аппаратно вырабатывать коды аутентификации электронных документов.

Комплекс «Аккорд 4++» с резидентным ПО v 1.40 обеспечивает доверенную загрузку ОС любого типа с файловой структурой FAT12, FAT16, FAT32, NTFS, HPFS, Free BSD. В этой связи данный комплекс получил название «Аккорд АМДЗ» — «аппаратный модуль доверенной загрузки». «Аккорд АМДЗ» вместе со специальным программным обеспечением быстро завоевал популярность. Однако, как всякое сложное наукоемкое изделие, СЗИ НСД эффективна только при правильном применении, т. е. на первый план выходят вопросы администрирования и управления механизмами безопасности.

Данный раздел посвящен практическим вопросам управления защитой информации на базе СЗИ «Аккорд». Рассматриваются вопросы управления как для автономной ПЭВМ, так и для ЛВС.

Подробное знание материала этого раздела абсолютно необходимо администраторам безопасности информации, строящим защиту своих АС на базе СЗИ НСД «Аккорд». Особый интерес представляют разделы 5.3 и 6, подготовленные на основе анализа типичных вопросов и практики управления защитой информации.

В этой главе не рассматриваются изделия для шины «PCI», средства кодов аутентификации и др. Основное внимание уделено СЗИ НСД на базе семейства «Аккорд 4++».

2. ОБЩИЕ СВЕДЕНИЯ

Программно-аппаратный комплекс средств защиты информации от несанкционированного доступа (ПАКСЗИ НСД) «Аккорд», далее комплекс «Аккорд», предназначен для применения на ПЭВМ типа IBM PC AT в целях защиты ПЭВМ и информационных ресурсов от НСД и обеспечения конфиденциальности информации, обрабатываемой и хранимой в ПЭВМ при многопользовательском режиме ее эксплуатации.

Комплекс разработан ОКБ САПР при участии фирмы «Инфокрипт» на основании лицензии Государственной технической комиссии при Президенте РФ (Гостехкомиссии России).

В настоящее время технические средства комплекса защиты от НСД «Аккорд» выпускаются в трех основных модификациях — «Аккорд 4», «Аккорд 4+», «Аккорд 4++».

Эти модификации комплекса:

— могут использоваться на ПЭВМ с процессором 80386 и выше, объемом RAM 1 Мбайт и более;

— требуют для установки свободный слот ISA;

— используют для идентификации персональные идентификаторы DS 199X с объемом памяти до 64 Кбит;

— используют для аутентификации пароль до 12 символов;

— блокируют загрузку с FDD;

— предусматривают регистрацию от 16 до 32 пользователей;

— имеют энергонезависимую память;

— имеют аппаратный датчик случайных чисел (ДСЧ);

— обеспечивают контроль целостности программ и данных;

— обеспечивают создание изолированной программной среды;

— могут использоваться как для локальных ПЭВМ, так и для рабочих станций в составе ЛВС.

Особенности модификаций комплекса «Аккорд» приведены в таблице 3.1.

Состав комплекса определяется при поставке в соответствии с требованиями Заказчика и указывается в формуляре. Комплекс «Аккорд» прошел сертификационные испытания в «Системе сертификации средств защиты информации по требованиям безопасности информации» РОСС RU.0001.01БИ00 (система сертификации Гостехкомиссии России) и имеет сертификаты соответствия (Приложение 1). Комплекс позволяет реализовать единые принципы защиты информации в соответствии с Законами РФ и требованиями нормативных документов по безопасности информации, а также обеспечивает изолированную программную среду и возможность создания функционально замкнутых информационных систем на базе ПЭВМ. При этом каждый пользователь обладает индивидуальным идентификатором DS 199x («Touch memory»-»Память касания»), далее по тексту ТМ-идентификатор, и личным паролем (до 12 символов), необходимыми ему для входа в ПЭВМ и доступа к назначенным ресурсам.

Таблица 3.1

Особенности применения	"Аккорд 4"	"Аккорд 4+"	"Аккорд 4++"
Бездисковые рабочие станции ЛВС	-	+	+
Серверы ЛВС	-	+	+
АРМ администратора безопасности информации	-	+	+
Наличие внутреннего идентификатора	-	+	+
Контроль физических линий	-	+	+
Возможность перепрограммирования всех элементов	-	-	+
Наличие контроля отключения питания	-	-	+
Подключение устройства энергонезависимого аудита	-	-	+
Интерфейс для считывателя smart-карт	-	-	+

Кроме серии «Аккорд 4» выпускается еще ряд изделий, в том числе:

«Аккорд 5» — для работы с шиной PCI;

«Аккорд СБ» — сопроцессор безопасности (ISA/PCI);

БУКА — блок установки кодов аутентификации (подключается на LPT-порт);

«Аккорд-микро» — для мобильных ПЭВМ (подключается на COM-порт);

и другие.

Ниже, тем не менее, описываются основные особенности управления защитой информации на базе серии «Аккорд 4».

2.1. Технические и организационные требования

Для установки комплекса «Аккорд» требуется следующий минимальный состав технических и программных средств:

— IBM PC AT совместимую ПЭВМ (с процессорами 80386 и старше);

— наличие на ПЭВМ HDD (для контроллеров «Аккорд 4+», «Аккорд 4++» не обязательно);

— наличие свободного слота на материнской плате ПЭВМ (ISA или PCI);

— операционная система MS DOS, Windows 3.X, Windows 95/98, Windows NT, OS/2, Linux.

Объем дискового пространства, необходимого для установки программных средств комплекса составляет от 1 Мб до 2,2 Мб в зависимости от модификации программных средств комплекса.

Количество идентификаторов, используемых в комплексе «Аккорд», определяется заказчиком при поставке.

Комплекс «Аккорд» проверен на совместимость практически со всем доступным разработчику программно-аппаратным обеспечением ПЭВМ как зарубежного, так и отечественного производства. Совместимость обеспечивается правильной установкой и настройкой комплекса.

Для эффективного применения комплекса и поддержания необходимого уровня защищенности ПЭВМ и информационных ресурсов *необходимы*:

— физическая охрана ПЭВМ и ее средств, в том числе проведение мероприятий по недопущению изъятия контроллера комплекса;

— наличие администратора безопасности информации (АБИ) — привилегированного пользователя, имеющего особый статус и абсолютные полномочия (супервизора). Администратор БИ планирует защиту информации на предприятии (учреждении, фирме и т.д.), определяет права доступа пользователей в соответствии с утвержденным Планом защиты, организует установку комплекса в ПЭВМ, эксплуатацию и контроль за правильным использованием ПЭВМ с установленным комплексом, в том числе учет выданных ТМ-идентификаторов, осуществляет периодическое тестирование средств защиты комплекса;

— использование в ПЭВМ технических и программных средств, сертифицированных как в Системе ГОСТ Р, так и в ГСЗИ.

Внимание!

Применение комплекса «Аккорд» совместно с сертифицированными программными средствами криптографической защиты информации (СКЗИ) и/или программными средствами защиты информации от НСД (СЗИ НСД) позволяет значительно снизить нагрузку на организационные меры защиты информации, определенные условиями применения этих средств. При этом класс защищенности не снижается.

2.2. Особенности защитных функций комплекса

Комплекс «Аккорд» — это простой, но чрезвычайно эффективный комплекс технических средств, используя который можно надежно защитить информацию на компьютере без переделки ранее приобретенных программных средств.

Защитные функции комплекса реализуются применением:

1) дисциплины защиты от НСД к ПЭВМ, включая идентификацию пользователя по уникальному ТМ-идентификатору и аутентификацию с учетом необходимой длины пароля и времени его жизни, ограничением времени доступа субъекта к компьютеру;

2) процедур блокирования экрана и клавиатуры в случаях, в которых могут реализовываться угрозы информационной безопасности;

3) дисциплины разграничения доступа к ресурсам АС (ПЭВМ), определяемой атрибутами доступа, которые устанавливаются администратором БИ в соответствие каждой паре «субъект доступа — объект доступа» при регистрации пользователей;

4) дисциплины применения специальных процедур печати, управления стандартными процедурами печати, процедурами ввода/вывода на отчуждаемые носители информации;

5) контроля целостности критичных с точки зрения информационной безопасности программ и данных (дисциплины защиты от несанкционированных модификаций), в том числе файлов операционной системы и служебных областей жесткого диска;

6) средств функционального замыкания информационных систем за счет использования средств защиты комплекса;

7) других механизмов защиты в соответствии с требованиями нормативных документов по безопасности информации.

Комплекс «Аккорд» может применяться в произвольной и функционально замкнутой программной среде, обеспечивая при этом класс защищенности АС (ПЭВМ) 1В по классификации [8], надежно обеспечивая при этом:

— защиту от несанкционированного доступа к АС (ПЭВМ) и ее ресурсам;

— разграничение доступа в ресурсам, в т.ч. к внешним устройствам, в соответствии с уровнем полномочий пользователей;

— защиту от несанкционированных модификаций программ и данных, внедрения разрушающих программных воздействий (РПВ);

— контроль целостности программ и данных;

— функциональное замыкание информационных систем с исключением возможности несанкционированного выхода в ОС, загрузки с FDD и несанкционированного прерывания контрольных процедур с клавиатуры.

Отметим, что в комплексе «Аккорд» используются и некоторые дополнительные механизмы защиты от НСД к ПЭВМ (АС). Так, в частности, для пользователя администратор БИ может установить:

— время жизни пароля и его минимальную длину, практически исключив тем самым возможность быстрого его подбора;

— временные ограничения использования ПЭВМ установкой интервала времени по дням недели (с дискретностью 30 мин), в котором разрешена работа для данного пользователя;

— параметры управления экраном — гашение экрана через заранее определенный интервал времени (в случае, если в течение указанного интервала действия оператором не выполнялись). Возможность продолжения работы предоставляется только после проведения повторной идентификации по персональному ТМ-идентификатору пользователя;

— подачу соответствующих звуковых и визуальных сигналов при попытках несанкционированного доступа к ПЭВМ (АС) и ее ресурсам.

Предусмотрено подключение подсистемы криптографической защиты информации, которая позволяет пользователю зашифровать/расшифровать свои данные с использованием индивидуальных ключей, хранящихся в персональном ТМ-идентификаторе и внутренней энергонезависимой памяти (ЭНП) компьютера. Поставка криптографических систем защиты информации (в соответствии с действующим законодательством) и библиотеки программ для программирования работы с контроллером комплекса «Аккорд» оговаривается при заказе комплекса.

2.3. Построение системы защиты информации на основе комплекса

Построение системы защиты информации с использованием комплекса «Аккорд» и ее взаимодействие с программно-аппаратным обеспечением ПЭВМ показаны на рис. 3.1.

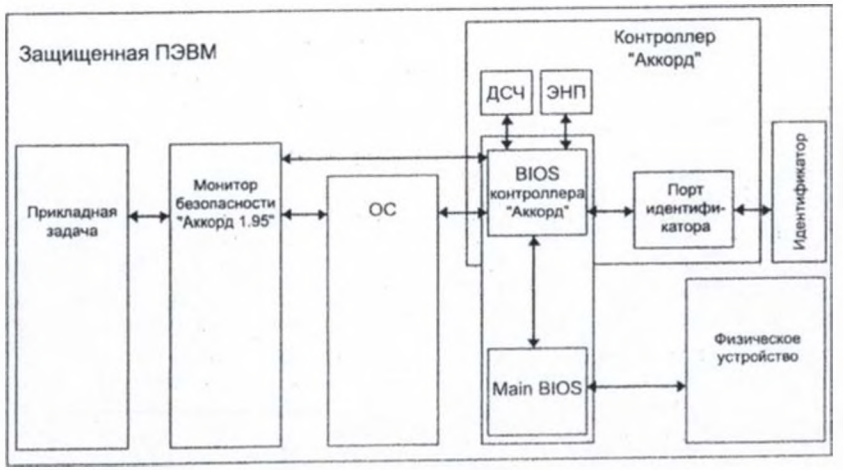


Рисунок 3.1

Защита информации с использованием средств комплекса основана на обработке событий, возникающих при обращении прикладных программ или системного ПО к ресурсам ПЭВМ. При этом средства комплекса перехватывают соответствующие программные и/или аппаратные прерывания, в случае возникновения контролируемого события (запрос прерывания) анализируют запрос и, в зависимости от соответствия полномочий субъекта доступа (его прикладной задачи), установленным ПРД, либо разрешают, либо запрещают обработку этих прерываний.

Комплекс «Аккорд» состоит из собственно средств защиты ПЭВМ от НСД и средств разграничения доступа к ее ресурсам, которые условно можно представить в виде четырех взаимодействующих между собой подсистем (Рис. 3.2) защиты информации.

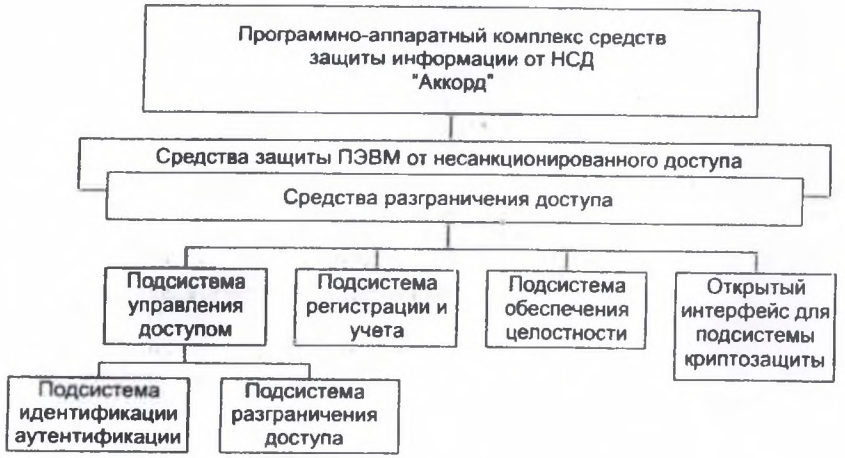


Рисунок 3.2

2.3.1. Подсистема управления доступом

Предназначена для защиты ПЭВМ от посторонних пользователей, управления доступом к объектам доступа и организации совместного их использования зарегистрированными пользователями в соответствии с установленными правилами разграничения доступа. Под посторонними пользователями понимаются все лица, не зарегистрированные в системе (не имеющие зарегистрированного в конкретной ПЭВМ ТМ-идентификатора). Защита от посторонних пользователей обеспечивается процедурами идентификации (сравнение предъявленного ТМ-идентификатора с перечнем зарегистрированных на ПЭВМ) и аутентификации (подтверждение подлинности) с защитой от раскрытия пароля. Для идентификации (аутентификации) пользователей в комплексе «Аккорд» используются интеллектуальные персональные идентификаторы DS 199X («Touch memory» - «Память касания»), отличающиеся высокой надежностью, уникальностью, наличием быстродействующей памяти, удобством пользования, приемлемыми массо-габаритными характеристиками и низкой ценой.

В комплексе «Аккорд» реализованы принципы дискреционного и мандатного управления доступом. Зарегистрированному пользователю устанавливаются права доступа по принципу регистрации «белого списка» разрешенных к запуску программ (задач), которые прописываются в ПРД. При запросе пользователя на доступ обеспечивается однозначная трактовка установленных ПРД, и, в зависимости от уровня полномочий пользователя, разрешается или запрещается запрошенный тип доступа.

2.3.2. Подсистема регистрации и учета

Предназначена для регистрации в системном журнале различных событий, происходящих в ПЭВМ. При регистрации событий в системном журнале регистрируются:

— дата и время события;

— пользователь, осуществляющий регистрируемое действие;

— действия пользователя (сведения о входе/выходе пользователя в/из системы, запусках программ, событиях НСД, изменении полномочий и др.).

Доступ к системному журналу возможен только администратору БИ (супервизору). В системный журнал заносятся сведения более чем о 200 типах событий, а также осуществляется архивация данных.

2.3.3. Подсистема обеспечения целостности

Предназначена для исключения несанкционированных модификаций (как случайных, так и злоумышленных) программной среды, в том числе программных средств комплекса, обрабатываемой информации, обеспечивая при этом защиту ПЭВМ от внедрения программных закладок и вирусов. В комплексе «Аккорд» это реализуется:

— проверкой целостности аппаратной части ПЭВМ;

— проверкой целостности назначенных для контроля служебных областей диска, системных файлов, в том числе КСЗИ НСД, пользовательских программ и данных;

— контролем обращения к операционной системе напрямую, в обход прерываний DOS;

— исключением возможности использования ПЭВМ без контроллера комплекса;

— механизмом создания изолированной программной среды, запрещающей запуск привнесенных программ, исключающей несанкционированный выход в ОС.

При проверке на целостность вычисляется контрольная сумма файлов и сравнивается с эталонным (контрольным) значением, хранящимся в энергонезависимой памяти контроллера «Аккорд». Эти данные заносятся при установке контроллера и могут изменяться в процессе эксплуатации ПЭВМ. В комплексе «Аккорд» используется сложный алгоритм расчета контрольных сумм — вычисление значения их хэш-функций — исключающий факт необнаружения модификации файла или контролируемого объекта (Приложение 2). Защита от модификации программы расчета хэш-функций обеспечивается тем, что она хранится в памяти контроллера комплекса.

2.4. Состав комплекса

Комплекс «Аккорд» включает программные и аппаратные средства.

Аппаратные средства

— одноплатный контроллер (ТУ 4024-001(002) 11443195-98), устанавливаемый в свободный слот материнской платы ПЭВМ;

— контактное устройство-съемник информации. Устанавливается обычно на передней панели ПЭВМ в любом подходящем месте (в зависимости от модификации съемника). При этом подключение осуществляется к задней планке контроллера посредством разъема RJ-11;

— интеллектуальный персональный идентификатор DS 199X («Touch memory» — «Память касания») — ТМ-идентификатор. Представляет собой полупассивное микропроцессорное устройство, снабженное элементом питания, в виде «таблетки» диаметром 16 мм и толщиной 3-5 мм в удобной пластмассовой (металлической) оправке. Каждый ТМ-идентификатор обладает уникальным номером (48 бит), который формируется технологически и подделать который практически невозможно. Объем памяти, доступной для записи и чтения составляет до 64 Кбит в зависимости от типа идентификатора. Срок хранения записанной информации, обеспечиваемый элементом питания, — не менее 10 лет.

Количество и тип ТМ-идентификаторов, модификации контроллера и контактного устройства оговаривается при поставке комплекса.

Программные средства

Специальное ПО разграничения доступа, контроля обращения к ресурсам и регистрации событий поставляется в различных версиях, в зависимости от используемой операционной системы.

2.4.1. Принцип работы комплекса

Плата контроллера комплекса «Аккорд» устанавливается в свободный слот материнской платы ЭВМ, производится установка программного обеспечения на жесткий диск, настройка комплекса, в том числе установление прав разграничения доступа, и регистрация пользователей. При регистрации пользователя администратором БИ определяются его права доступа: списки исполняемых программ и модулей, разрешенных к запуску данным пользователем и список стартовых (исполняемых непосредственно после загрузки ОС) программ и др. — см. раздел «Администрирование комплекса». После регистрации пользователю выдается на руки персональный ТМ-идентификатор, о чем делается запись в журнал учета носителей информации.

Особенностью и, несомненно, преимуществом комплекса «Аккорд» является проведение процедур идентификации, аутентификации и контроля целостности защищаемых объектов до загрузки операционной системы. Это обеспечивается при помощи специального ПО, записанного в энергонезависимую память контроллера. Данная область памяти доступна только на чтение, что исключает несанкционированную модификацию процедур контроля. BIOS контроллера получает управление во время так называемой процедуры ROM-SCAN. Суть данной процедуры в следующем.

В процессе начального старта после проверки основного оборудования BIOS компьютера начинает поиск внешних ПЗУ в диапазоне от C 800:0000 до E000:0000 с шагом в 2K. Признаком наличия ПЗУ является наличие слова AA55H в первом слове проверяемого интервала. Если данный признак обнаружен, то в следующем байте содержится длина ПЗУ в страницах по 512 байт. Затем вычисляется контрольная сумма всего ПЗУ, и если она корректна — будет произведен вызов процедуры, расположенной в ПЗУ со смещением. Такая процедура обычно используется для инициализации дополнительных устройств, установленных в ПЭВМ. В комплексе «Аккорд» в этой процедуре проводится идентификация и аутентификация пользователя, контроль аппаратуры ПЭВМ, служебных областей диска и файлов операционной системы. Если любая из перечисленных процедур завершается некорректно, то загрузка выполняться не будет.

При установленном контроллере и инсталлированном ПО комплекса «Аккорд» загрузка компьютера осуществляется в следующем порядке.

1. BIOS компьютера выполняет стандартную процедуру POST (проверку основного оборудования компьютера) и, по ее завершении переходит к процедуре ROM-SCAN, во время которой управление перехватывает контроллер комплекса «Аккорд» и выводится сообщение:

«Access system BIOS v.1.xx copyright OKB SAPR 1993-1999_ s/n .....»

2. На монитор выводится окно с приглашением пользователю предъявить свой ТМ-идентификатор.

Это окно остается на мониторе до момента контакта ТМ-идентификатора пользователя и съемника информации, либо до завершения установленного промежутка времени, в течение которого необходимо предъявить ТМ-идентификатор.

3. Если идентификатор не зарегистрирован, то выводится сообщение:

«Незарегистрированный ТМ-идентификатор»

и происходит возврат к п.2, а по истечении установленного для идентификации времени на экран монитора выводится сообщение:

«Таймаут»

Для возобновления работы необходимо выключить и повторно включить питание компьютера.

4. При легальном ТМ-идентификаторе выводится окно с приглашением пользователю ввести пароль для аутентификации и начинается отсчет времени:

«Введите пароль»

5. При неправильно введенном пароле выводится сообщение:

«Недопустимый пароль»

и происходит возврат к п.2.

6. При правильно введенном пароле выводится сообщение:

«Доступ разрешен»

7. Выполняется процедура контроля целостности. Если процедура завершается корректно, то продолжается процедура загрузки ОС.

Вся процедура идентификации/аутентификации занимает 7-10 секунд. Устойчивость ее зависит от длины пароля. Для проведения процедуры аутентификации предусмотрен режим ввода пароля в скрытом виде — в виде символов <*>. Этим предотвращается возможность раскрытия индивидуального пароля и использования утраченного (похищенного) ТМ-идентификатора. Если предъявленный пользователем идентификатор не зарегистрирован в списке (сообщение «Неизвестный идентификатор») или нарушена целостность защищаемых файлов (сообщение «Нарушение целостности»), загрузка ОС не производится. Для продолжения работы потребуется вмешательство администратора БИ.

Таким образом, контрольные процедуры (идентификация, аутентификация, проверка целостности системных файлов и т.д.) осуществляются до загрузки ОС, при этом обеспечивается защита от разрушающих программных воздействий (РПВ). В любом другом случае, т.е. при неподтверждении прав пользователя на работу с данной ПЭВМ, загрузка ОС не выполняется.

3. УСТАНОВКА ПРОГРАММНО-АППАРАТНОГО КОМПЛЕКСА СЗИ НСД «АККОРД»

Установка программно-аппаратного комплекса СЗИ НСД «Аккорд» осуществляется в следующие порядке.

1. Установка подсистемы идентификации/аутентификации, включающая:

— установку платы контроллера в свободный слот ПЭВМ и регистрацию администратора БИ (супервизора), настройку комплекса в соответствии с конфигурацией технических средств ПЭВМ;

— регистрацию пользователей, назначение пользователям личных ТМ-идентификаторов, паролей и прав доступа;

— назначение списка файлов, контролируемых на целостность.

2. Установка подсистемы разграничения доступа, включающая:

— установку специального ПО на жесткий диск ПЭВМ;

— установку пользователям правил разграничения доступа (ПРД) к ресурсам ПЭВМ.

Регистрация пользователей, назначение им ПРД, контроль целостности аппаратной части ПЭВМ и файлов описаны в разделе 4 данной главы. Порядок первоначальных установок описан ниже.

3.1. Установка аппаратных средств

1. Установка в ПЭВМ контроллера комплекса.

Внимание!

Установка контроллера должна производится только при выключенном питании компьютера!

Перед установкой аппаратной части комплекса необходимо:

— Отключить питание.

— Вскрыть корпус системного блока ПЭВМ, удалить заглушку на задней панели блока и выбрать свободный слот на материнской плате для установки контроллера комплекса.

Контроллеры «Аккорд 4+» и « Аккорд 4++», входящие в состав комплекса, имеют два режима доступа к аппаратным ресурсам платы контроллера.

Режим 0 (стандартный): доступ к области кода расширения BIOS только по чтению.

Режим 1 (специальный): доступ к области кода расширения BIOS по чтению/записи, причем при старте компьютера код не исполняется, а области,

скрытые при работе операционной системы в режиме 0, становятся доступны по чтению/записи. Переход из стандартного режима в специальный требует снятия установочной металлической планки, которая крепится к плате контроллера двумя винтами. В специальном режиме возможна перезапись внутреннего ПО контроллера без изменения аппаратной части. При записи кода в BIOS контроллера следует отключить любые менеджеры памяти, установленные на компьютере.

Штатные операции изменения режима работы производятся под контролем администратора БИ(сотрудника отдела(службы) защиты информации). При этом возможна установка пломбы на крепежный винт, которая является индикатором целостности как встроенного ПО, так и конфиденциальной информации.

2. Назначение джамперов и разъемов.

Расположение джамперов на платах контроллеров изделия «Аккорд» показано на рис. 3.3 и 3.4.

Джамперы J0-J2 предназначены для установки базового адреса ПЗУ «Аккорд» в памяти ПЭВМ. Состояние "1" джампера соответствует состоянию "1" определенного разряда шины адреса, "0" – состоянию "0" соответственно. Состояние "0" определяется размыканием контактов. Состояние "1" определяется замыканием контактов. К разъему внутреннего съемника подключается кабель внутреннего контактного устройства (съемника информации) для ТМ-идентификаторов.

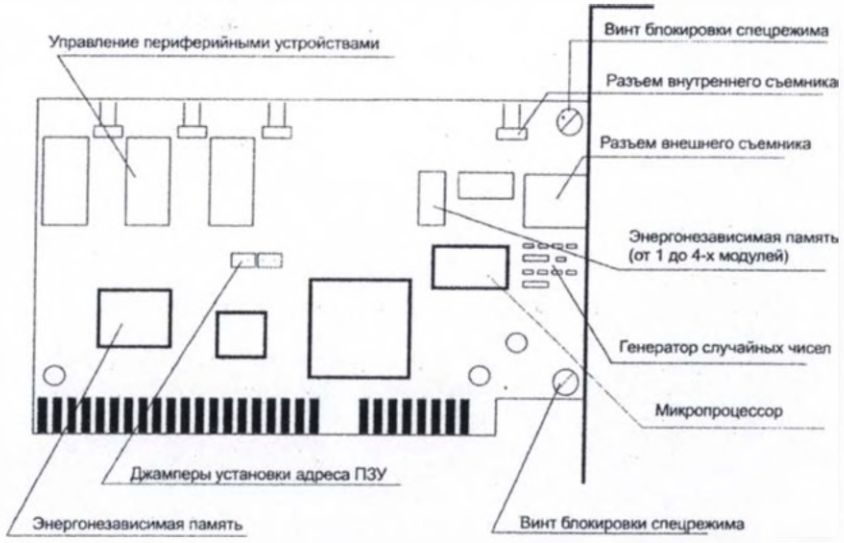


Рисунок 3.3. Плата контроллера «Аккорд 4+»

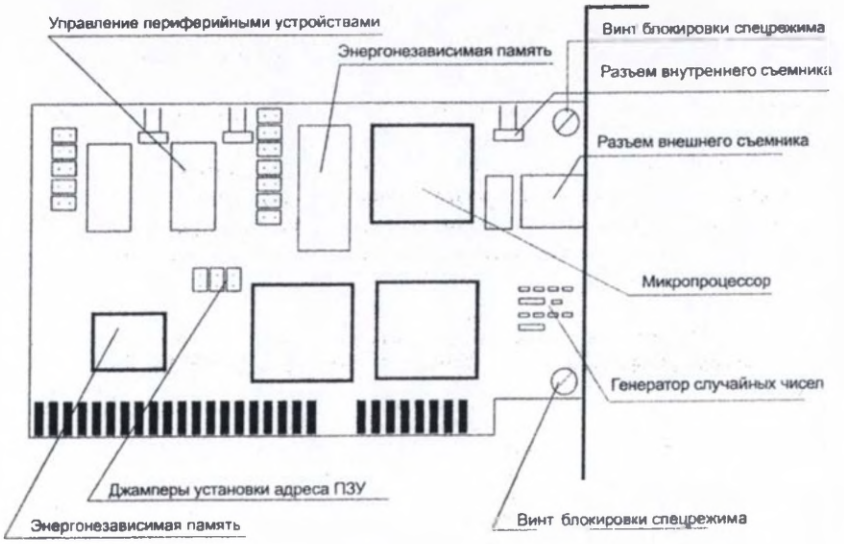


Рисунок 3.4. Плата контроллера «Аккорд 4++»

3. Выбор и установка начального адреса ПЗУ контроллера.

ПЗУ контроллера занимает в памяти 8К и может быть установлено по адресам (здесь и далее адреса памяти указываются в шестнадцатеричном виде в формате «сегмент:смещение») от C800:0000 до DC00:0000. Следует заметить, что стандартная область для установки пользовательских ПЗУ в ПЭВМ типа IBM PC C800:0000 - DF80:07FF, а по адресу C000:0000, как правило, установлено ПЗУ видеоадаптера. Желательно пользоваться стандартной областью (особенно в компьютерах типа COMPAQ, DELL, Hewlett Pakkard, DEC и других фирм Brand name).

Перед установкой начального адреса ПЗУ контроллера комплекса следует выяснить, какие дополнительные ПЗУ присутствуют в ПЭВМ, а также все занятые ими области памяти (есть смысл обращать внимание только на диапазон возможной установки ПЗУ). Это можно выполнить при помощи поставляемой программы MEMSCAN.EXE, а также при помощи популярных программ SYSINFO из набора Norton Utilities или Microsoft Diagnostics.

Для выбора начального адреса ПЗУ комплекса «Аккорд» в памяти ПЭВМ необходимо запустить программу MEMSCAN.EXE с поставляемой дискеты. После запуска программы на экран выводится сообщение, показанное на рис. 3.5.

Внимание!

В SETUP компьютера должен быть отключен режим «Shadow RAM» для области памяти, в которой расположен BIOS контроллера «Аккорд».

В правой части сообщения показываются все занятые и незанятые адреса ПЗУ, присутствующих в ПЭВМ, в левой — положение переключателей джамперов, определяющих начальный адрес ПЗУ «Аккорд» на плате контроллера. Выбор начальных адресов ПЗУ осуществляется с помощью клавиш <стрелка вниз> и <стрелка вверх>. При наличии свободных участков следует выбрать наименьший (но не менее 8К) и задать начальный адрес ПЗУ равным начальному адресу этого участка установкой замыкателей соответствующих джамперов на плате контроллера в соответствии с их положением, указанным в таблице 3.2 и 3.3.

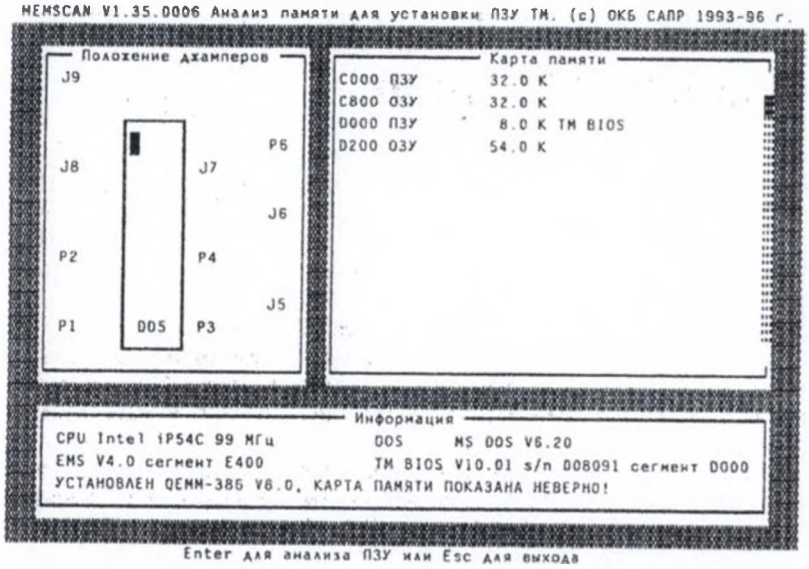


Рисунок 3.5. Вид сообщения программы memscan.exe для контроллера «Аккорд»

Необходимо особо отметить, что если активны какие-либо менеджеры памяти типа QRAM, QEMM-386, 386MAX, EMM386, LastByte и т.п., то программа memscan.exe НЕ ПОКАЖЕТ все свободные участки, т.к. менеджеры памяти могут создать в этих свободных участках области UMB. Поэтому следует временно исключить из файлов AUTOEXEC.BAT и CONFIG.SYS строки, отвечающие за активизацию таких менеджеров. Затем следует перезагрузиться, после чего можно выбрать при помощи программы MEMSCAN.EXE нужный адрес ПЗУ. После того, как адрес выбран — можно восстановить старое содержимое файлов AUTOEXEC.BAT и CONFIG.SYS.

Установка джамперов, предлагаемая изготовителем, определяет начальный адрес ПЗУ D000:0000 и обеспечивает стабильную работу комплекса на большинстве ПЭВМ, однако возможны конфликты с платами модемов, контроллеров сканеров и другими дополнительными платами, установленными в компьютер.

Для установки начального адреса ПЗУ наряду с информацией, полученной с использованием программы MEMSCAN.EXE, следует воспользоваться данными, приведенными в таблицах 3.2 и 3.3.

Нумерация джамперов выполнена слева направо (вид со стороны элементов, разъем ЭВМ снизу)

0 — джампер разомкнут, 1 — джампер замкнут.

Таблица 3.2
Примеры установок джамперов для адресов ПЗУ на плате контроллера «Аккорд 4++»

J0	J1	J2	Базовый адрес ПЗУ
0	0	0	D000
0	0	1	C800
1	0	1	CC00

В настоящее время также выпускается контроллер «Аккорд 4+»с двумя джамперами (без J0).

Таблица 3.3
Примеры установок джамперов для адресов ПЗУ на плате контроллера «Аккорд 4+»

J1	J2	Базовый адрес ПЗУ
0	0	D000
1	0	D400
0	1	D800

4. Подсоединение контактного устройства (съемника информации)

При использовании внутреннего контактного устройства (съемника информации) типа DS 1909, DS 1909 T, R 1909, Эллиас их установка производит-

ся, как правило, на заглушке зарезервированного места для дисководов. Для этого необходимо снять заглушку, просверлить в ней отверстие диаметром 9,4 мм, в случае, если используется съемник с пружинной шайбой или 6 мм — при использовании съемника с клеевой основой, вставить в отверстие контактное устройство и закрепить его на заглушке пружинной (резиновой) шайбой или при помощи клеевой основы, вывести провод контактного устройства внутрь ПЭВМ и установить заглушку на место.

Внимание!

Провод центрального контакта съемника (обозначен на соединительном разъеме знаком <треугольник>) должен соответствовать правому контакту разъема. Неправильное подсоединение съемника информации к плате контроллера к фатальным последствиям не приведет, однако контроллер работать не будет.

Подсоединение выносного контактного устройства осуществляется с помощью разъема RJ-11 (подобного телефонному разъему) на задней планке контроллера.

5. Установка контроллера в свободный слот материнской платы ПЭВМ

Контроллер устанавливается в любой свободный слот материнской платы, как это показано на рис. 3.6, и фиксируется стопорным винтом к задней панели корпуса.

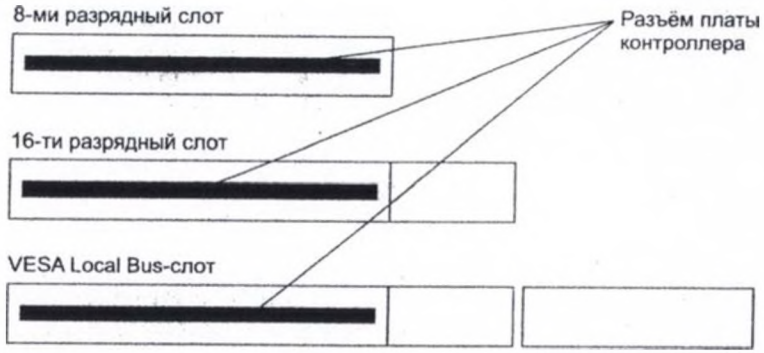


Рисунок 3.6. Установка контроллера комплекса в слотах материнской платы

При помощи программы администратора, записанной в энергонезависимой памяти контроллера, следует обязательно зарегистрировать администратора БИ и назначить ему ТМ-идентификатор. Также обязательно произвести контроль аппаратуры и дисков, для того чтобы в памяти контроллера запомнилась конфигурация данной ПЭВМ. Если все действия произведены правильно, то после выхода из программы администрирования по клавише <Esc>

становятся доступными для выбора остальные пункты стартового меню администратора. Если этого не происходит, то необходимо вернуться в режим администрирования и провести регистрацию администратора (супервизора) более внимательно в соответствии с «Руководством администратора».

Далее необходимо перезагрузить компьютер и убедиться в том, что в процессе загрузки появляется сообщение на синем фоне: «Прислоните ТМ-идентификатор...» и после прикосновения ТМ-идентификатором Супервизора к съемнику информации происходит загрузка ПЭВМ и выводится стартовое меню администратора.

Зарегистрировать пользователей и назначить им права доступа (ПРД) к ресурсам ПЭВМ с помощью программы администратора (в стартовом меню выбрать пункт «Администрирование»). Аппаратная часть комплекса «Аккорд» установлена!

3.2. Установка ПО разграничения доступа на жесткий диск

Установка ПО комплекса на жесткий диск ПЭВМ осуществляется в следующей последовательности.

1. Вставьте в дисковод для гибких дисков дистрибутивную дискету 1, входящую в комплект поставки.

2. Запустите находящуюся на дискете программу INSTALL.EXE. Выберите из меню вариант инсталляции для контроллера "Аккорд 4+", или "Аккорд 4++". Программа создаст на диске C: каталог C:\ACCORD со всеми необходимыми подкаталогами и скопирует туда выбранное программное обеспечение. На данном этапе не производится никаких изменений жесткого диска, кроме создания каталогов или файлов. В частности, будут созданы файлы AUTOEXEC.ACC и CONFIG.ACC, которые представляют собой копии файлов AUTOEXEC.BAT и CONFIG.SYS с внесенными изменениями, необходимыми для работы комплекса.

Рекомендуется распечатать эти файлы и ознакомиться с изменениями, которые необходимо внести в файлы AUTOEXEC.BAT и CONFIG.SYS для правильного функционирования системы «Аккорд». Эти изменения должны быть рассмотрены на предмет корректности их установки с учетом загружаемого ими окружения (особенно при использовании меню в файле AUTOEXEC.BAT). Это необходимо сделать, так как бывают случаи, когда вызовы программ комплекса включаются дважды или некорректно, хотя программа INSTALL в большинстве случаев правильно создает файлы AUTOEXEC.ACC и CONFIG.ACC.

3. После завершения программы INSTALL.EXE необходимо запустить программу TMAC4P.EXE из каталога C:\ACCORD:

Формат командной строки для загрузки драйвера TMAC4P.EXE:

TMAC4P.EXE AUTO
или: TMAC4P.EXE /U для выгрузки драйвера

Запуск TMAC4P.EXE ? или TMAC4P.EXE /? выводит подсказку на экран. Командную строку для TMAC4P.EXE можно набирать любыми буквами.

Если установлен контроллер «Аккорд 4++», то следует запустить драйвер TMAC4PP.EXE с такими же ключами.

При помощи программы C:\ACCORD\ACED.EXE (см. п.4.2) зарегистрировать администратора БИ и пользователей *с теми же ТМ-идентификаторами и именами, которые зарегистрированы в контроллере.*

Активизация подсистемы разграничения доступа к ресурсам ПЭВМ(АС) заключается в изменении установок в файлах AUTOEXEC.BAT и CONFIG.SYS в соответствии с дополнениями, указанными в созданных при инсталляции файлах AUTOEXEC.ACC и CONFIG.ACC.

Эти изменения сводятся к следующему:

В файл AUTOEXEC.BAT должен быть включен вызов драйвера TMAC4P.EXE с необходимыми параметрами.

Вместо какой-либо программной оболочки в файле AUTOEXEC.BAT должен быть установлен вызов C:\ACCORD\ACRUN.EXE. Следует заметить, что программа ACRUN.EXE может быть запущена с ключом /R, т.е. C:\ACCORD\ACRUN.EXE /R. В этом случае при завершении работы программы ACRUN.EXE будет произведена перезагрузка ПЭВМ. Именно этот режим работы следует считать основным, однако на время тестирования ключ /R можно не включать. Программа ACRUN.EXE является монитором прав доступа и запускает стартовую задачу, назначенную тому пользователю, который вошел в систему.

В файл CONFIG.SYS должна быть включена загрузка драйвера AMDZ.SYS:

DEVICE= C:\ACCORD\AMDZ.SYS

Данный драйвер запрещает применять клавиши <Ctrl>, <Alt> и любые их комбинации в процессе исполнения файлов CONFIG.SYS и AUTOEXEC.BAT. Нажатие любой комбинации данных клавиш вызывает перезагрузку компьютера. Тем самым исключается возможность прерывания исполнения файлов CONFIG.SYS и AUTOEXEC.BAT пользователем. Данная строка должна быть помещена в файле CONFIG.SYS первой. В случае применения QEMM-386 данная строка должна быть первой после вызова DOSDATA.SYS, QEMM386.SYS, DOS-UP.SYS. Загрузка QEMM386.SYS должна выполнятся с ключами BE:N BF:N. Также для любого менеджера памяти следует запретить использование области памяти, в которой находится BIOS контроллера.

Вы можете использовать программу ACEDPM.EXE вместо программы ACED. ACEDPM.EXE функционально является аналогом ACED.EXE, но при работе использует всю доступную расширенную память (свыше 1Mb). Обратите внимание, что *при установке правил разграничения доступа к сетевым ресурсам следует явно указать имя сервера, имя тома* и только потом каталог или файл на сервере (см. 5.3, пример 18).

3.3. Снятие средств защиты комплекса «Аккорд»

Внимание!

Снятие защиты разрешено только администратору БИ (супервизору).

Для снятия защиты необходимо выполнить следующие действия:

1. Загрузить систему ТМ-идентификатором администратора БИ (супервизора).
2. Удалить из файлов CONFIG.SYS и AUTOEXEC.BAT вызовы, относящиеся к комплексу «Аккорд».
3. Отключить питание.
4. Вскрыть корпус системного блока ПЭВМ.
5. Снять аппаратную часть комплекса.

4. АДМИНИСТРИРОВАНИЕ СРЕДСТВ ЗАЩИТЫ ИНФОРМАЦИИ КОМПЛЕКСА «АККОРД»

Администрирование средств защиты информации комплекса «Аккорд» осуществляется с помощью двух подсистем:

— подсистемы администрирования внутреннего ПО, расположенного в энергонезависимой памяти контроллера. С помощью этой подсистемы администратор СЗИ может добавлять и удалять пользователей, назначать пользователям ТМ-идентификаторы и пароли, контролировать аппаратную часть ПЭВМ, прикладные и системные файлы, получает доступ к системному журналу контроллера, назначает метки конфиденциальности томам на жестком диске и уровни доступа пользователям.

— подсистемы администрирования, входящей в состав специального ПО, установленного на жесткий диск ПЭВМ (редактор правил разграничения доступа пользователей, монитор прав доступа, программа работы с журналом событий и др.). С помощью данных программ администратор БИ описывает правила разграничения доступа (ПРД) пользователей, добивается их исполнения и контролирует действия пользователей и прикладного ПО, установленного на ПЭВМ.

Термины

Пользователь — субъект доступа к объектам (ресурсам) ПЭВМ.

Администратор — администратор службы безопасности информации.

ТМ-идентификатор (или ТМ) — идентификатор Touch-memory DS-199X.

Использовать ТМ-идентификатор — приложить ТМ-идентификатор к съемнику.

Меню — окно с изображением кнопок с названиями команд. Перемещения по меню осуществляется с помощью мыши или <Tab>. Выбор команды — мышью (левая клавиша) или <Enter>, выход из меню — <Esc> или командой в меню.

Окно ввода/вывода — служит для ввода и отображения буквенно-цифровой информации, а так же может выполнять функции меню. Содержит окно для ввода буквенно-цифровой информации, окна списков, кнопки команд, окна флагов. Ввод буквенно-цифровой информации должен заканчиваться <Enter> или перемещением в другое окно, движение списка в окне — с помощью стрелок или мышью. Перемещение по окнам и кнопкам команд, выбор команд и выход из окна — аналогично работе с меню.

Сообщения — информация, выводимая на дисплей, которая сообщает о действиях, требуемых от пользователя, о состоянии программы и о нормально завершенных действиях.

Ошибки — информация, выводимая на дисплей, указывающая на неправильность действий, сбои, аварии комплекса.

Пояснения — в описании некоторых команд даются пояснения и рекомендации администратору для использования этих команд. Пояснения выделены мелким шрифтом.

4.1. Подсистема администрирования внутреннего ПО контроллера

При загрузке компьютера с установленным контроллером серии «Аккорд 4+»(«Аккорд 4++»), если список зарегистрированных пользователей пуст, на экран выводится стартовое меню администратора (Рис. 3.7). В этом меню доступен для выбора только один пункт «Администрирование».

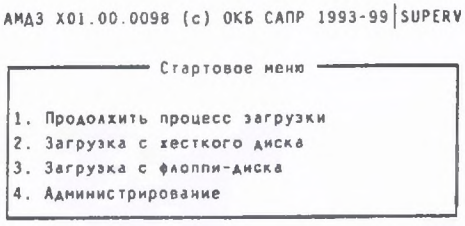


Рисунок 3.7. Стартовое меню администратора

Клавишей <Enter> запустите программу администрирования. На экран выводится главное меню (Рис. 3.8).

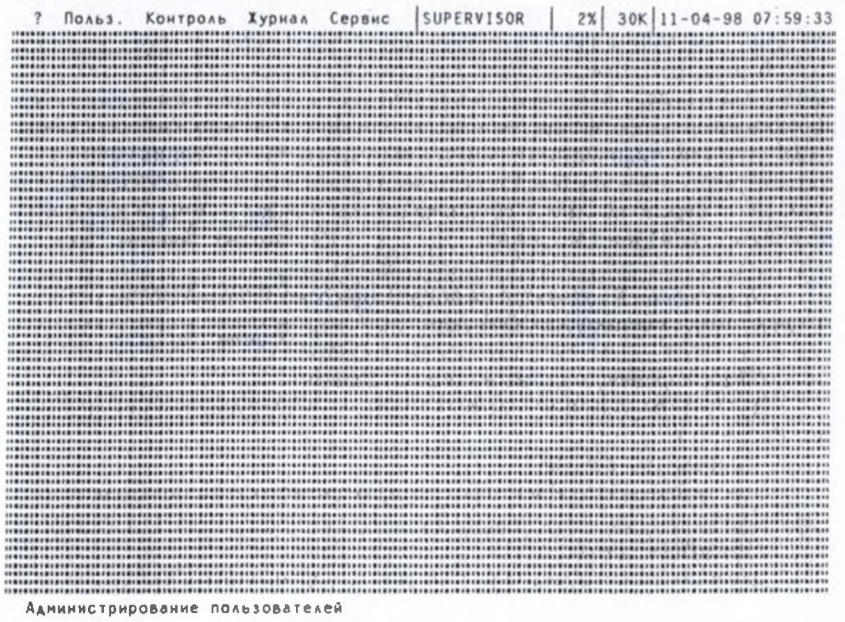


Рисунок 3.8. Главное меню администратора

Главное меню состоит из следующих полей:
— строка команд (левая половина верхней строки),
— информационная строка (правая половина верхней строки),
— статус (HELP) - строка (нижняя срока),
— рабочее поле (все остальное пространство).

В строке команд доступны следующие опции:
— <Польз.>;
— <Контр>;
— <Журнал>;
— <Сервис>;
— <Помощь>.

4.1.1. Список пользователей

В меню выберите команду <Польз.>. На экран выводится подменю списка пользователей (Рис. 3.9).

В подменю списка пользователей зарезервированы две группы пользователей — «Администраторы» и «Обычные». Для каждой из групп можно задать общие параметры, которые будут устанавливаться и действовать по умолчанию для каждого пользователя в группе при создании этого пользователя. Для каждого из зарегистрированных пользователей можно изменить данные параметры при индивидуальной настройке. Для редактирования общих параметров группы пользователей необходимо клавишами <стрелка> или мышью установить курсор на строке заголовка группы и нажать <Enter> или дважды щелкнуть левой кнопкой мыши.

Примечание. Контроллер на аппаратном уровне поддерживает работу с мышью, подключенной на COM-порт.

Общие параметры группы «Администраторы»

Для группы «Администраторы» установлены следующие общие параметры (Рис. 3.10.):

— параметры пароля;

— результаты ИА (Идентификации/Аутентификации пользователя).

Параметры пароля

Для пользователя, у которого введен пароль, можно регулировать следующие параметры пароля (Рис. 3.11.):

— «Кто может менять пароль» — установка этого параметра позволяет пользователю самому менять пароль после истечения времени действия, или смену пароля может осуществлять только администратор.

— «Минимальная длина» — параметр определяет минимальную длину пароля.

— «Время действия (дни)» — определяет период смены пароля.

— «Попыток для смены» — параметр устанавливает число попыток для смены пароля (для выполнения смены пароля необходимо ввести старый пароль, а затем дважды — новый).

Результаты ИА

В разделе «Результаты ИА» устанавливается, какия информация о пользователе, полученная в результате процесса Идентификации/Аутентификации, будет передаваться из контроллера в программную подсистему разграничения доступа (если таковая установлена на компьютере). Не рекомендуется изменять установки по умолчанию (Рис. 3.12). Если на диске компьютера установлено другое программное обеспечение, кроме СЗИ «Аккорд», которое работает с платой контроллера, то рекомендации по установке этих параметров дает фирма-разработчик данного ПО.

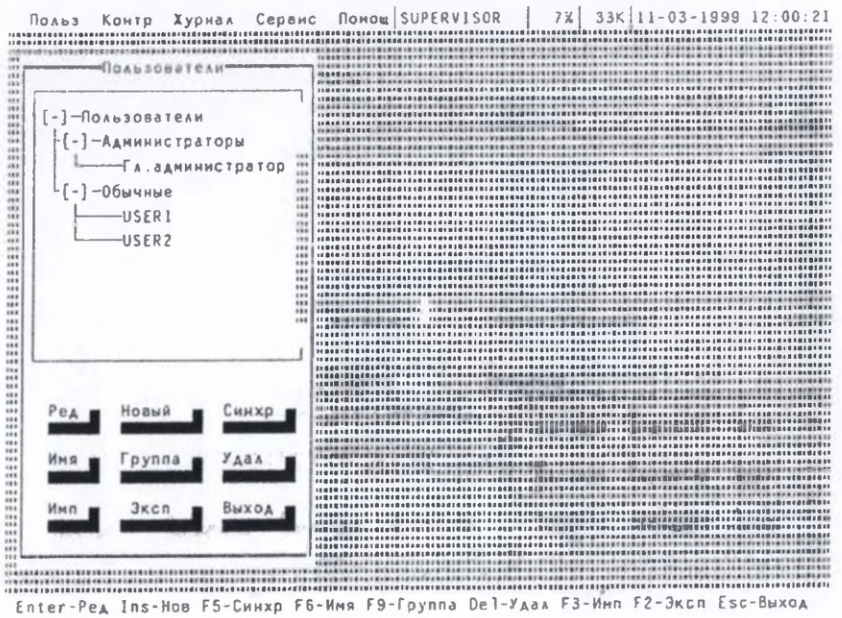


Рисунок 3.9. Подменю списка пользователей

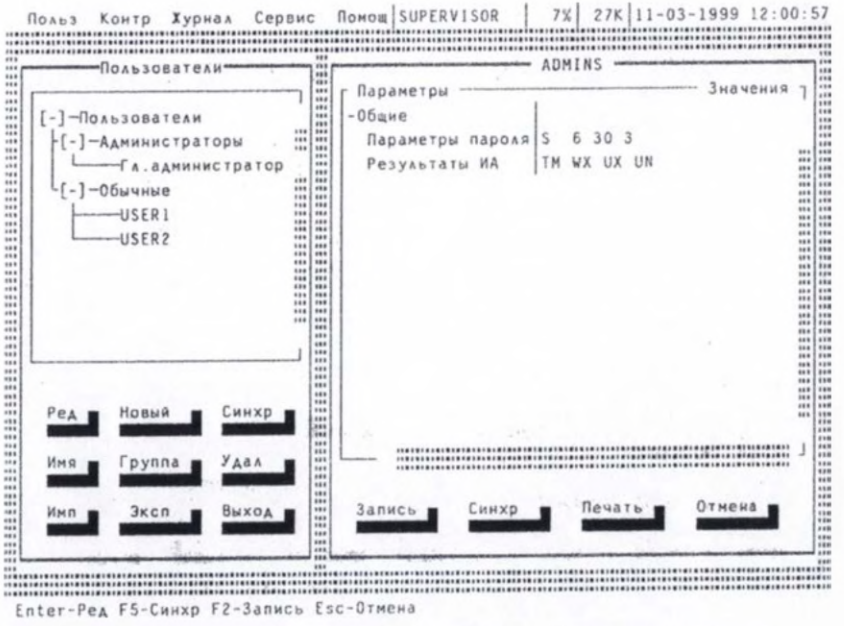


Рисунок 3.10. Общие параметры для группы «Администраторы»

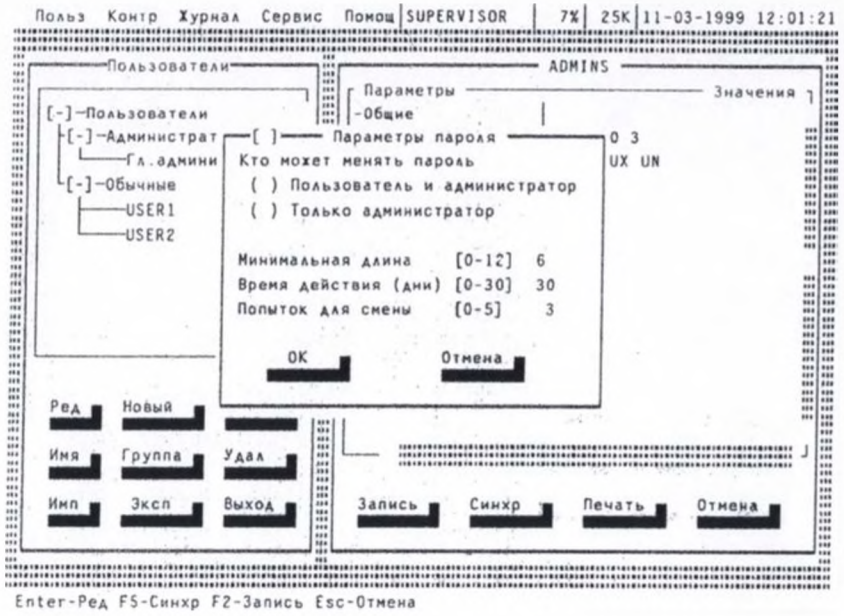


Рисунок 3.11. Параметры пароля

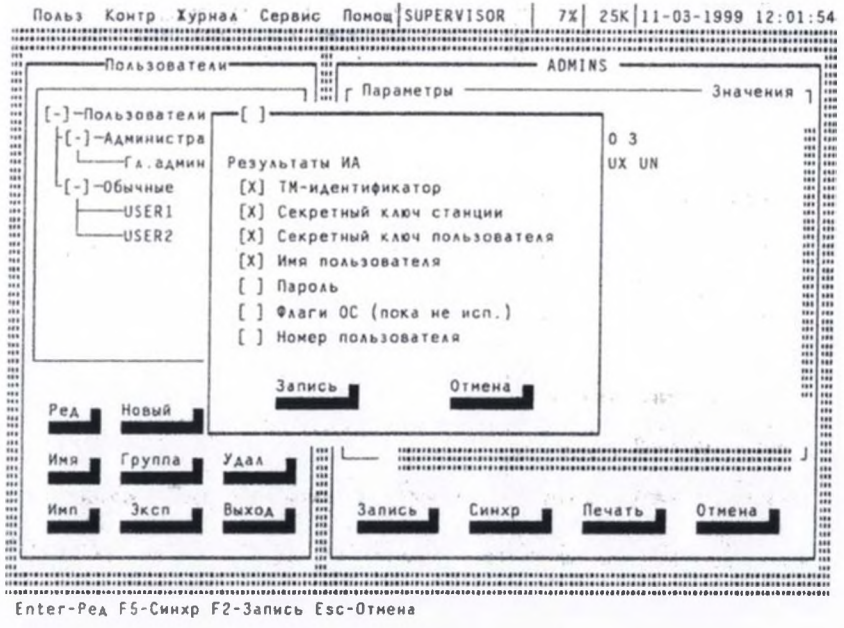


Рисунок 3.12. Результаты ИА

Общие параметры группы «Обычные» (пользователи)

Для группы «Обычные» (пользователи) установлены следующие общие параметры (Рис. 3.13):

— параметры пароля;
— временные ограничения;
— загрузка ОС;
— форма допуска;
— доступ к логическим дискам;
— доступ к устройствам;
— результаты ИА (Идентификации/Аутентификации пользователя).

Разделы «Параметры пароля» и «Результаты ИА» такие же, как общие параметры группы «Администраторы». Другие пункты рассмотрим подробнее.

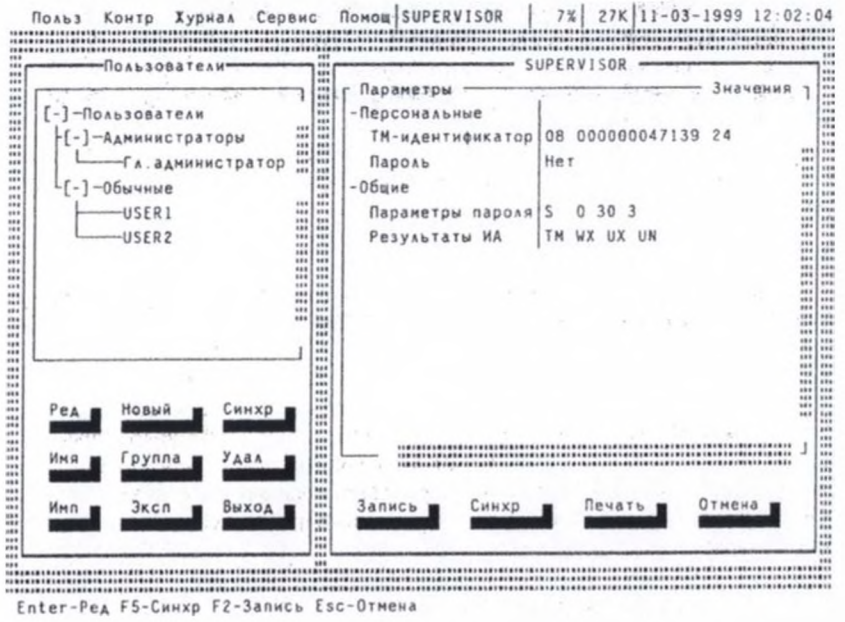


Рисунок 3.13. Общие параметры группы «Обычные» пользователи

Временные ограничения

Администратор может устанавливать для пользователя ограничения на вход в систему с точностью до 30 минут в любой день недели. Выберите пункт «Временные ограничения» и нажмите <Enter>. На экран выводится окно «Временные ограничения» (Рис. 3.14).

Клавишами <стрелка> можно перемещаться по матрице времени входа в систему. Клавиша <Пробел> меняет знак "+" на "-" и обратно, т.е. разрешает или запрещает загрузку компьютера данному пользователю в данный временной интервал.

Загрузка ОС

В контроллере «Аккорд» предусмотрена возможность управления режимом загрузки Windows 95/98 и загрузкой различных конфигураций ПО с использованием меню в файле CONFIG.SYS.

Выберите пункт «Загрузка ОС» и нажмите <Enter>. На экран выводится окно со списком возможных вариантов загрузки Windows 95, меню выполнения CONFIG.SYS для Windows 95 и MSDOS (если она установлена) (Рис. 3.15). С помощью клавиши <Пробел> в квадратных скобках можно установить или сбросить флаг разрешения выбора того или иного сценария загрузки. Клавиша <F6> служит для перемещения курсора от одного окна к другому. Отмеченные флагом пункты меню становятся доступными пользователю для выбора в процессе загрузки ОС путем нажатия на клавишу с номером пункта. Клавиши со стрелками блокируются на момент загрузки как на основной, так и на дополнительной (цифровой) клавиатуре.

Внимание!

Для успешной работы данной опции под Window 95 в файле MSDOS.SYS в разделе [Options] должна быть прописана строка BootMenu=1.

Форма допуска

Этот параметр позволяет устанавливать для пользователя уровень доступа к логическим разделам жесткого диска при использовании дисциплины мандатного доступа (см. подменю «Установки» в пункте «Сервис» меню администратора). Выберите пункт «Форма допуска». При нажатии клавиши <Enter> циклически меняется значение уровня доступа пользователя. Значение 3 соответствует самому низкому уровня доступа (несекретно), значение 1 соответствует самому высокому — секретно. Необходимо помнить, что установленная для пользователя форма допуска работает только при включенном режиме мандатного доступа (подменю «Установки» в пункте «Сервис» меню администратора).

Доступ к логическим дискам

Данный раздел позволяет администратору управлять доступом к логическим дискам на жестком диске ПЭВМ (Рис. 3.16).

Доступ изменяется дискретно (либо он есть, либо его нет) и конфигурация доступа может быть индивидуальной для каждого пользователя. Разграничение доступа выполняется контроллером до загрузки ОС и поэтому не зависит от типа установленной на компьютере операционной системы. Если включен режим мандатного доступа, то дискреционный доступ к логическим дискам является дополнением режима мандатного доступа.

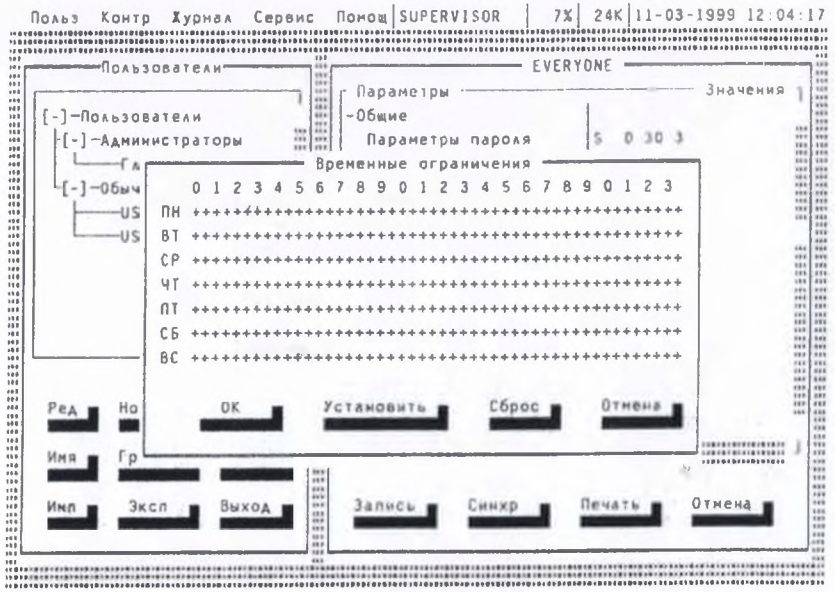


Рисунок 3.14. Временные ограничения на загрузку компьютера

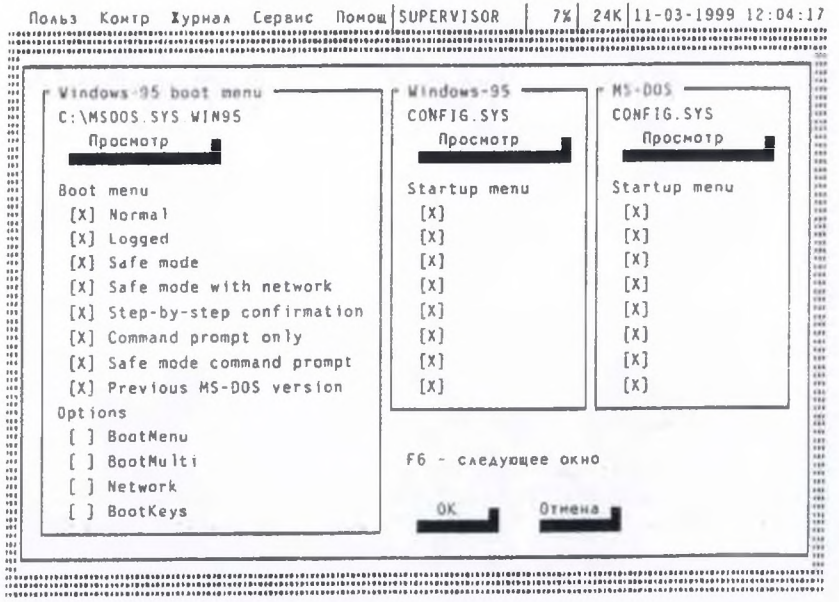


Рисунок 3.15. Управление загрузкой ОС

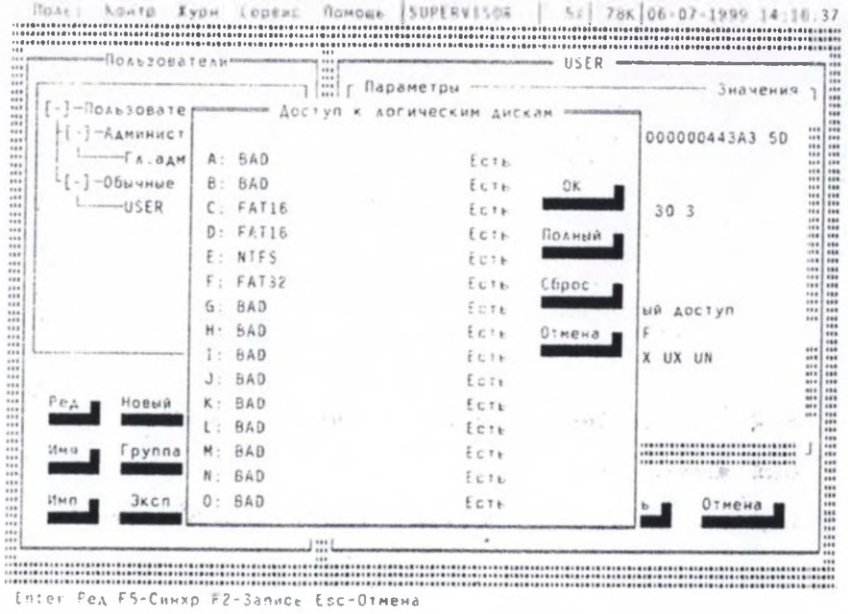


Рисунок 3.16. Доступ к логическим дискам

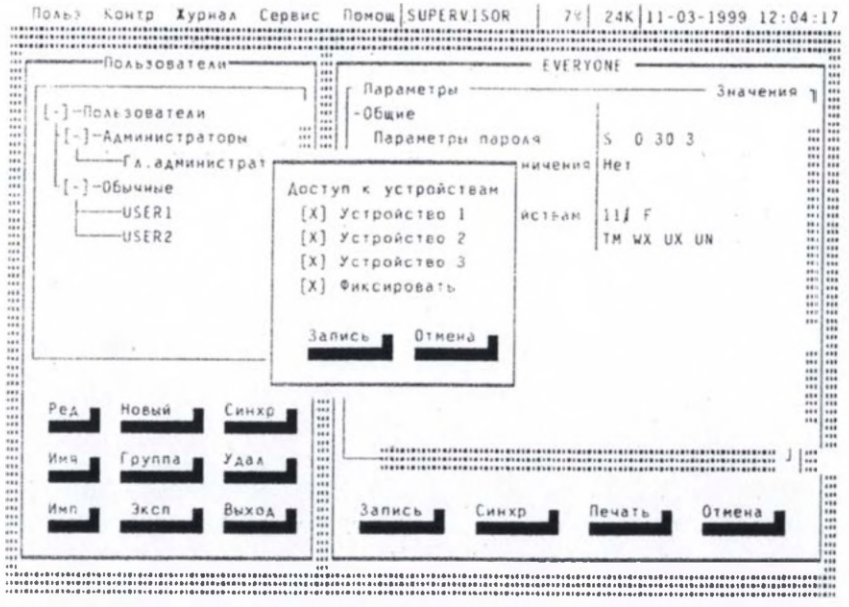


Рисунок 3.17. Управление доступом к устройствам

Управление устройствами
(для контроллера с установленными реле управления)

Встроенное ПО дает возможность управлять 3-мя независимыми гальванически развязанными контактными парами, с помощью которых можно блокировать доступ отдельных пользователей к внешним (по отношению к контроллеру) устройствам, например, к накопителю FDD, CD-ROM или принтеру. Причем, запрет действует либо на момент загрузки операционной системы, либо на весь сеанс работы пользователя.

Для установки режимов управления устройствами нужно выбрать пункт «Управление устройствами» и нажать <Enter>. На экран выводится окно со списком устройств (Рис. 3.17).

С помощью клавиши <Пробел> в квадратных скобках можно установить или сбросить флаг разрешения работы устройства. Переход к пунктам <Запись> <Отмена> осуществляется клавишей <Tab> или мышью. Флаг, установленный в строке «Фиксировать», запрещает работу устройств на весь сеанс работы пользователя.

Внимание!

На управляемую контактную пару может быть заведен сигнал напряжением не более 5В и силой тока не более 300 Ма.

Регистрация супервизора (администратора безопасности информации)

При инициализации контроллера в списке уже находится пользователь «Гл. администратор», но для него не введены никакие атрибуты. Для регистрации администратора системы нужно выбрать строку <Гл. администратор>, нажмите <Enter>. На экран выводится окно ввода-вывода «Параметры пользователя» (Рис. 3.18).

Назначение ТМ-идентификатора

Для назначения пользователю нового ТМ-идентификатора необходимо выбрать команду <ТМ идентификатор> (Рис. 3.18). На экран выводится информация о ТМ-идентификаторе (Рис. 3.19). Далее выбирается команда <Новый>. На запрос ТМ-идентификатора (Рис. 3.20) нужно приложить ТМ-идентификатором к съемнику. Для отмены текущей операции выбирается команда <Отмена>.

Генерация секретного ключа

После назначения ТМ-идентификатора на экране появляется меню генерации секретного ключа (Рис. 3.21), который уникален для каждого пользователя и записывается во внутреннюю память регистрируемого ТМ-идентификатора. Этот секретный ключ пользователя используется в мониторе правил разграничения доступа ACRUN, который позволяет каждому пользователю создать изолированную программную среду (ИПС) и персональный набор файлов, контролируемых на целостность. В отличие от аппаратуры «Аккорд», ACRUN ориентирован на конкретные операционные системы, и поставляется по отдельному заказу.

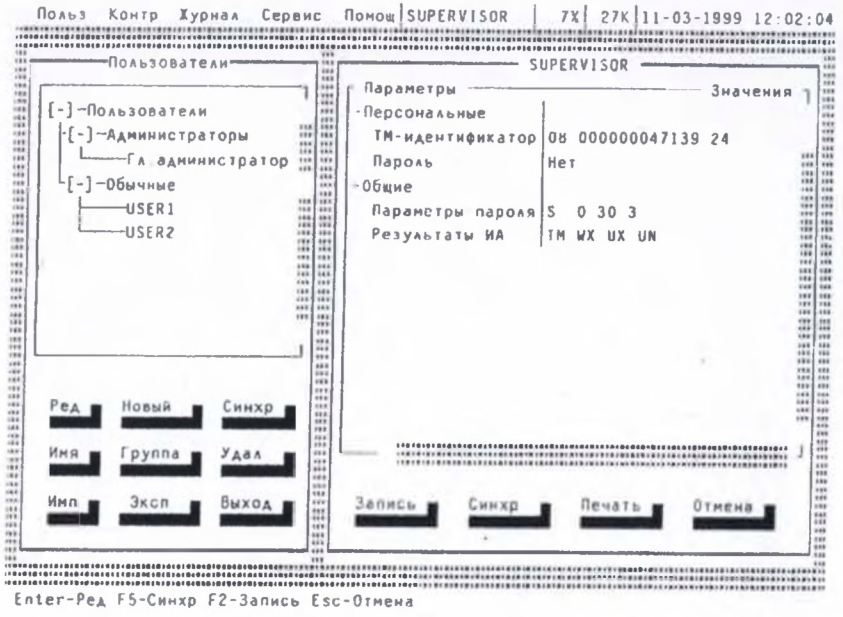


Рисунок 3.18. Редактирование параметров пользователя «Гл. администратор»

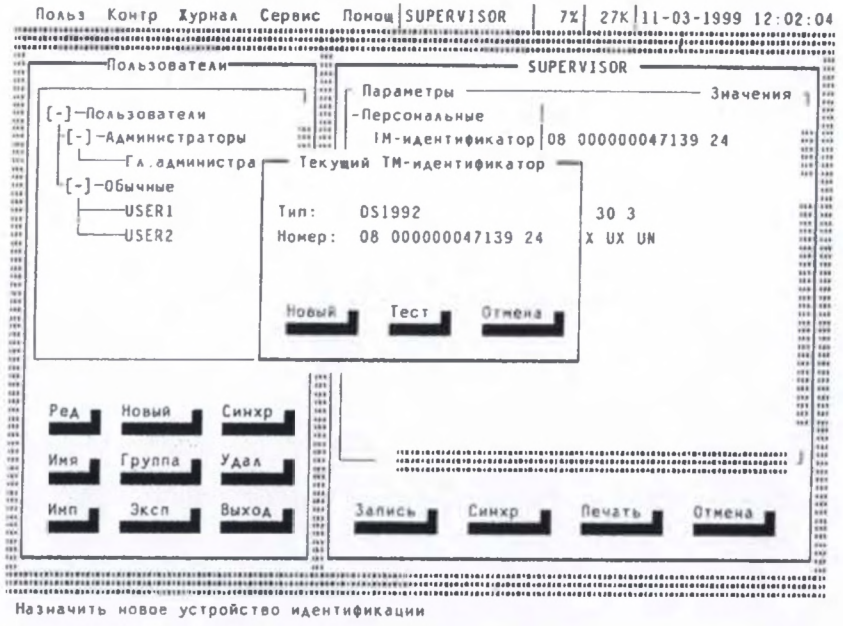


Рисунок 3.19. Информация о ТМ-идентификаторе

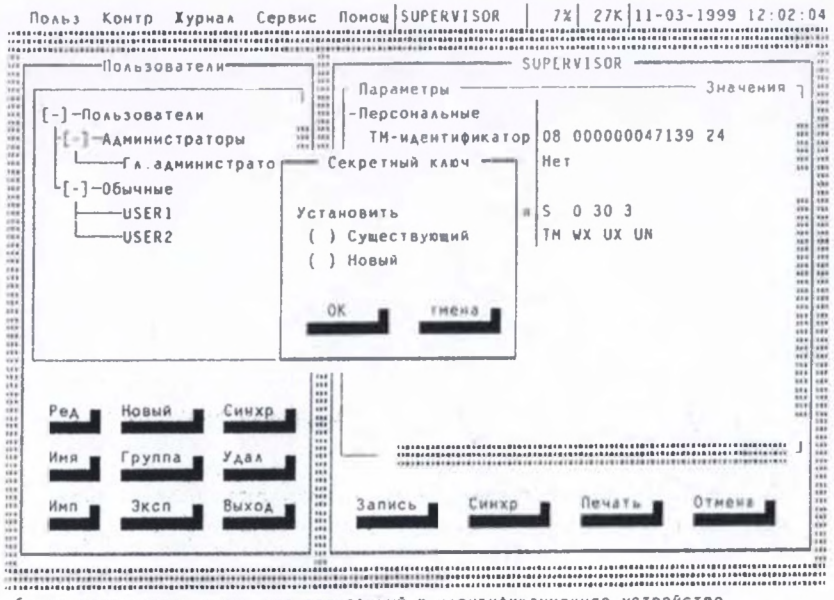


Рисунок 3.20. Запрос ТМ-идентификатора

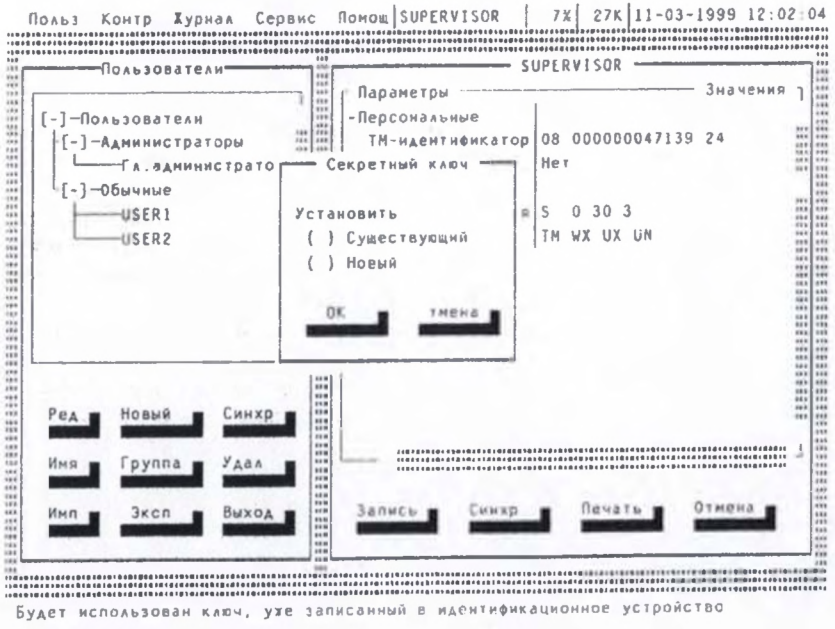


Рисунок 3.21. Генерация секретного ключа пользователя

Внимание!

ТМ-идентификатор, в котором не записан секретный ключ, воспринимается как недопустимый в процессе идентификации/аутентификации пользователя, даже если его номер высвечивается в строке «ТМ-идентификатор».

Генерация и запись ключа осуществляется автоматически после выбора опций <Новый> и <OK>. На запрос ТМ-идентификатора необходимо прикоснуться ТМ к съемнику.

Примечание. Секретный ключ может быть уже записан в ТМ в следующих случаях:

— при перерегистрации пользователя;

— при регистрации одного пользователя на нескольких компьютерах с установленной системой «Аккорд».

Генерировать секретный ключ следует только при первой регистрации, т.к. каждая новая генерация затирает предыдущее значение ключа, и ТМ-идентификатор не будет читаться на других компьютерах.

В этом случае для отмены генерации нового ключа и регистраци записанного в ТМ ранее следует выбрать опцию <Существующий> и <OK>, и на запрос ТМ-идентификатора прикоснуться ТМ к съемнику.

Назначение пароля

Для назначения пароля в окне «Параметры пользователя» (Рис. 3.18) следует выбрать строку «Пароль» с последующим нажатием <Enter>. На экран выводится окно ввода пароля (Рис. 3.22).

Ввод пароля осуществляется дважды — в строке «Новый пароль» и в строке «Еще раз». Введите новый пароль. Повторите ввод пароля во второй строке. Пароль может состоять из букв, цифр и символов клавиатуры. Вводимые символы на экране отображаются точками. При несовпадении введенных последовательностей выводится сообщение об ошибке. В этом случае операцию придется повторить. Символы могут вводиться как в верхнем, так и в нижнем регистре. Будьте внимательны! Длина пароля должна быть не меньше параметра, установленного в строке «Минимальная длина» в разделе «Параметры пароля». Если длина введенного пароля меньше, выводится сообщение об ошибке. Не допускается ввод в качестве пароля последовательностей типа: "123456" или "qwerty". При вводе подобных последовательностей символов выдается сообщение об ошибке.

Внимание!

Если пользователю не назначается пароль, то в строке «Минимальная длина» в разделе «Параметры пароля» следует установить длину пароля 0, иначе при записи данных о пользователе (по клавише F2) выводится сообщение об ошибке.

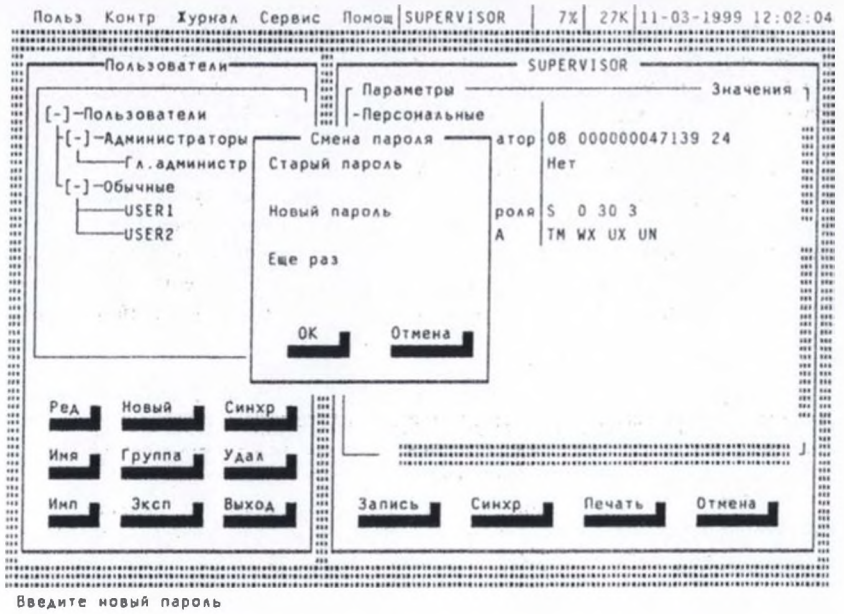


Рисунок 3.22. Окно ввода пароля

Для сохранения параметров "SUPERVISOR"а и выхода в окне «Параметры пользователя» выберите команду <Запись> (клавиша <F2>).

Регистрация нового пользователя

В списке пользователей (Рис. 3.18) установите курсор на заголовке группы «Обычные». По команде <Новый> или клавише <Insert>на экран выводится окно ввода имени пользователя. Администратор должен присвоить каждому пользователю уникальное в данной вычислительной среде (отдельный компьютер или локальная сеть) имя. Рекомендуется использовать в качестве «имени» фамилию пользователя. После ввода имени пользователя на экран выводится окно ввода-вывода «Параметры пользователя» (Рис. 3.23), в котором необходимо зарегистрировать ТМ-идентификатор и пароль пользователя. При вводе нового пользователя общие параметры, установленные для группы, присваиваются ему по умолчанию, но в окне «Параметры пользователя» их можно изменить. Если администратор безопасности изменяет общие параметры группы, то установить их для всех пользователей группы можно по команде <Синхр.> (Синхронизировать).

Удаление пользователя из списка зарегистрированных

Если возникла необходимость удалить пользователя, то в подменю списка пользователей (Рис. 3.9) нужно выбрать и пометить имена пользователей, предназначенных для удаления из списка. После нажатия клавиши <Del> и подтверждения команды удаление будет осуществлено.

Редактирование параметров пользователей

В этом режиме администратор производит изменение параметров доступа пользователя к защищенным объектам. В подменю списка пользователей (Рис. 3.9) выберите имя пользователя, параметры которого необходимо отредактировать, нажмите клавишу <Enter>. На экран выводится окно (Рис. 3.23). Произведите изменения в окне ввода/вывода «Параметры пользователя».

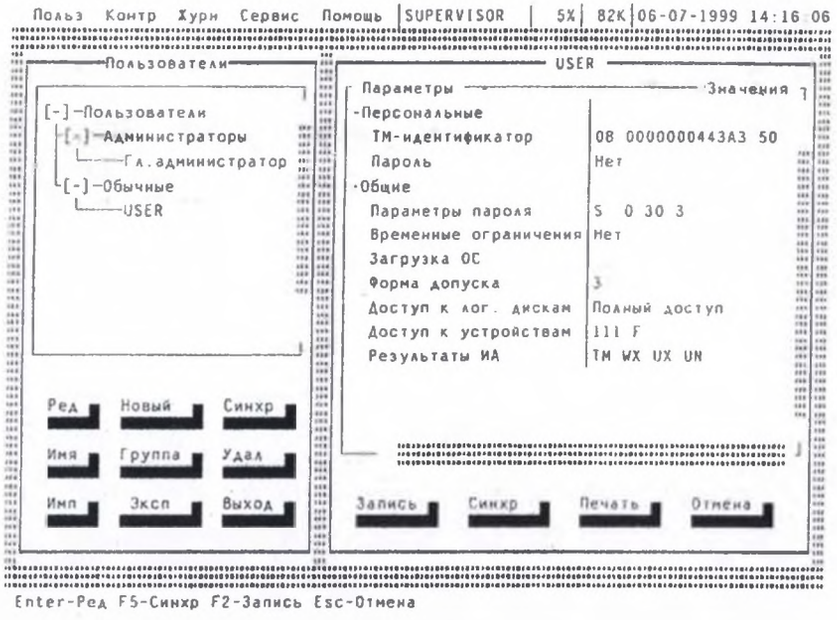


Рисунок 3.23. Редактирование параметров пользователя

4.1.2. Контроль

В этом режиме администратор осуществляет управление контролем состав и параметров аппаратной части ПЭВМ, выбором файлов для контроля их целостности.

В главном меню выбирается команда <Контроль>. На экран выводится подменю контроля, состоящее из пунктов:

— <Аппаратура>
— <Диски>
— <Файлы>
— <Реестр Win 95>
— <Реестр Win NT>.

Контроль аппаратуры

Для контроля аппаратуры в подменю выберается команда <Аппаратура> и подтверждается нажатием <Enter>. На экран выводится окно контроля аппаратуры (Рис. 3.24).

В левой колонке выводится список контролируемых устройств, в средней — состояние устройств и контрольные суммы, записанные в энергонезависимой памяти контроллера, а в правой — текущее состояние аппаратуры и контрольных сумм. Скролирование окна производится клавишами <Page Up> и <Page Down> или мышью в правой полосе прокрутки. Если данные совпадают, то они высвечиваются в обеих колонках одинаковым цветом. При несовпадении данные в колонках высвечиваются разными цветами. Проанализировав причину возникших отличийБ администратор может запустить операцию обновления комбинацией клавиш <Alt>+<U>. Для включения устройства в список контролируемых необходимо установить на него курсор и нажать клавишу <Insert>. Для снятия отметки используется клавиша <Del>. Клавиша <Пробел> раскрывает/сворачивает дерево параметров в контролируемой группе. Выход из режима контроля аппаратуры осуществляется клавишей <Esc>.

После регистрации в СЗИ «Аккорд» хотя бы одного пользователя контроль аппаратуры производится при каждой загрузке компьютера после идентификации/аутентификации пользователя. Если обнаруживается несовпадение параметров конфигурации, записанных в памяти контроллера и текущих параметров системы, то выдается сообщение на красном фоне «Разберитесь с ошибками» и загрузка компьютера блокируется для обычного пользователя или выводится стартовое меню, если идентифицирован администратор.

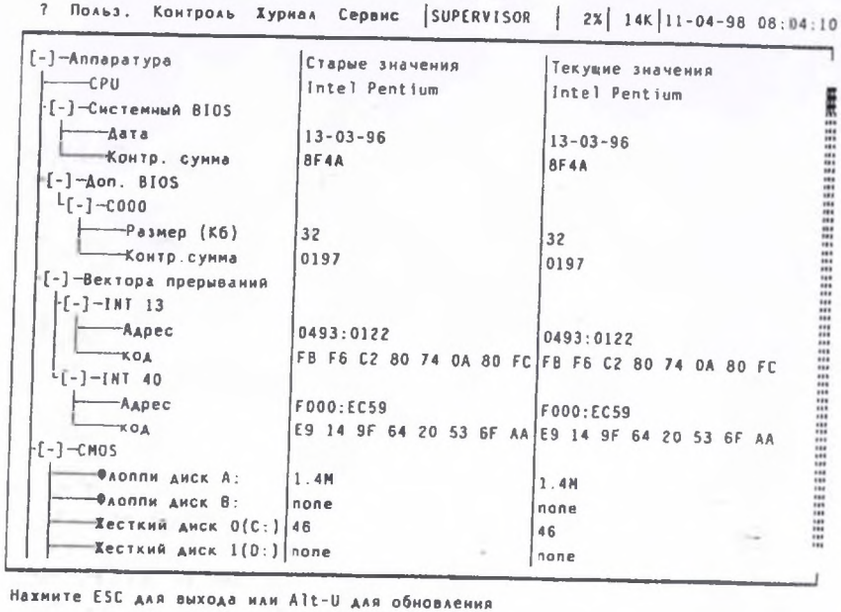


Рисунок 3.24. Окно контроля аппаратной части компьютера

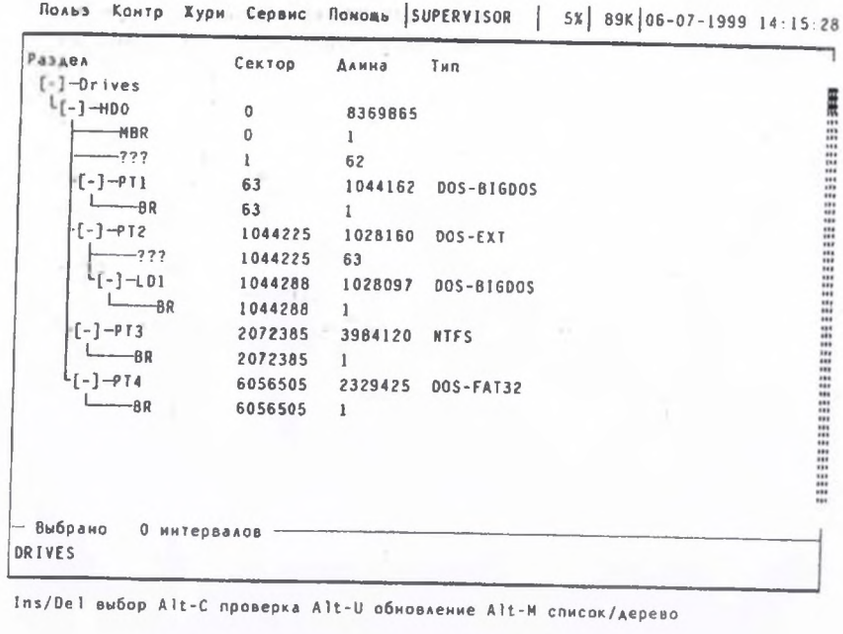


Рисунок 3.25. Окно контроля служебных областей диска

Контроль целостности служебных областей жестких дисков

В подменю <Контроль> выбирается команда <Диски> и подтверждается нажатием <Enter>. На экран выводится окно контроля служебных областей дисков (Рис. 3.25.). Поддерживаются файловые системы следующих типов: FAT12, FAT16, FAT32, NTFS, HPFS, FreeBSD.

В окне контроля выводится дерево всех дисков, установленных на данном компьютере, с указанием файловой системы каждого диска. Перемещение по дереву выполняется стрелками или клавишами <Page Up> и <Page Down>. Для включения области диска в список контролируемых необходимо установить на него курсор и нажать клавишу <Insert>. Для снятия отметки используется клавиша <Del>. В список контролируемых можно вносить служебные области с любых дисков, установленных в компьютере, независимо от файловой системы. Для пересчета и записи в память контроллера хэш-функций контролируемых областей используется комбинация клавиш <Alt>+<U> (обновление).

Контроль целостности файлов

В подменю <Контроль> выбирается команду <Файлы> и подтверждается нажатием <Enter>. На экран выводится окно контроля файлов (Рис. 3.26). СЗИ обеспечивает контроль целостности программ и данных до загрузки ОС, защиту от внедрения разрушающих программных воздействий (РПВ). Поддерживаются файловые системы следующих типов: FAT12, FAT16, FAT32, NTFS, HPFS, FreeBSD.

В окне контроля файлов выводится список всех дисков установленных в системе с указанием файловой системы каждого диска. Перемещаться по списку можно клавишами <Стрелка вниз>, <Стрелка вверх>. Клавиша <Пробел> раскрывает/сворачивает дерево каталогов на диске или подкаталогов в каталоге. Перемещение по дереву выполняется стрелками или клавишами <Page Up> и <Page Down>. Для включения файла в список контролируемых необходимо установить на него курсор и нажать клавишу <Insert>. Для снятия отметки используется клавиша <Del>. В список контролируемых можно вносить файлы с любых дисков, установленных в компьютере, независимо от файловой системы. Для пересчета и записи в память контроллера хэш-функций файлов используется комбинация клавиш <Alt>+<U> (обновление), для расчета хэш-функций и сравнения с записанными в контроллере (для выявления измененных файлов) используется комбинация клавиш <Alt>+<C> (проверка). Комбинация клавиш <Alt>+<M> изменяет представление на экране файлов в виде списка, либо в виде дерева. Хэш-функция контролируемых файлов пересчитывается при каждой загрузке компьютера с установленным контроллером «Аккорд АМДЗ» и сравнивается с эталонным значением, записанным в памяти контроллера. Если обнаруживается несовпадение, то выдается сообщение на красном фоне «Разберитесь с ошибками» с указанием в нижней строке состояния на каком этапе выявлена ошибка («Контроль аппаратуры» или «Контроль файлов») и загрузка компьютера блокируется для обычного пользователя или выводится стартовое меню, если идентифицирован администра-

тор. Администратор, запустив программу администрирования, может выполнить операцию проверки в разделе <Контроль>/<Файлы> и выявить измененные файлы.

Примечание. Если в каталоге находятся файлы, внесенные в список контролируемых, то этот каталог нельзя свернуть клавишей <Пробел> при отображении каталогов в виде дерева.

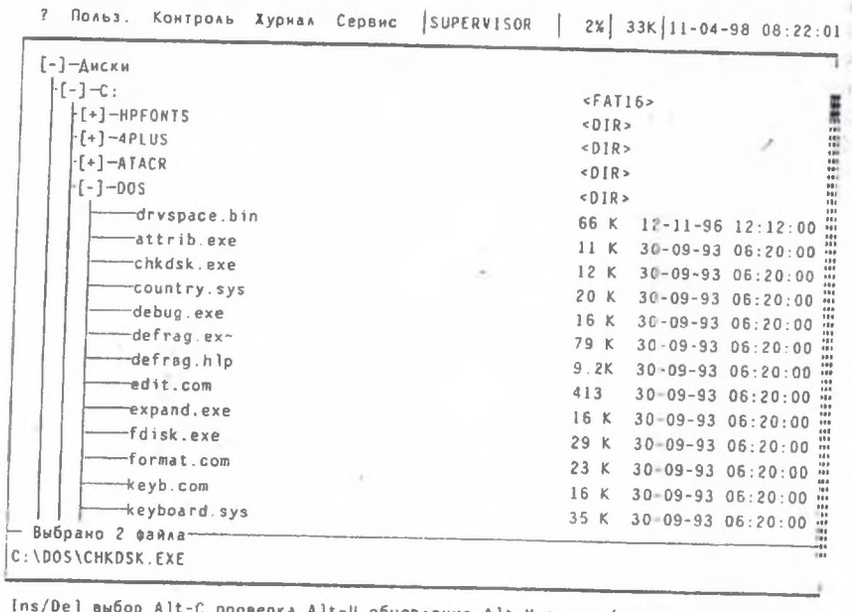


Рисунок 3.26. Окно контроля целостности файлов

Количество файлов, которые можно установить на контроль, зависит от операционной системы и от длины пути к каталогу, где находятся файлы. Для MS DOS эта цифра примерно равна 400, для Win 95 — 300, а для Win NT — 250 файлов.

Системный журнал

В СЗИ «Аккорд» ведется системный журнал, размещенный в энергонезависимой памяти контроллера. В журнал заносится информация о сеансах работы пользователей с указанием номера ТМ-идентификатора и все попытки несанкционированного доступа к компьютеру.

Для просмотра журнала в главном меню выбирается команда <Журнал> и подтверждается нажатием <Enter>. На экран выводится окно системного журнала (Рис. 3.27). В левой колонке выводится дата и время начала сеанса работы, а для остальных событий этого сеанса выводится только время в виде смещения от начала работы. Во второй колонке выводится наименование выполненной операции. В третьей — серийный номер ТМ-идентификатора. В четвертой — результат операции. Расшифровка наименований и результатов операций дана в Приложении 3. В самой верхней строке экрана после имени пользователя выводится процент заполнения области памяти контроллера, отведенной под системный журнал.

Выход из режима просмотра журнала по клавише <Esc>. Стереть содержимое журнала можно с помощью клавиши <Delete>.

4.1.3. Сервис

В подменю «Сервис» пункт «Ключ станции» зарезервирован для дальнейших разработок, и изменять параметры в этом пункте не рекомендуется.

С помощью пунктов «Прерывания» и «Память» можно просмотреть прерывания и области памяти компьютера.

Пункт «Установки» позволяет изменять некоторые параметры (Рис. 3.28.):

«Страница ТМ» — определяет, с какой страницы памяти ТМ-идентификатора располагается служебная информация. Данный параметр изменять не рекомендуется.

«Звуковое сопровождение» — включение данного флага означает, что процедура начальной идентификации/аутентификации будет сопровождаться звуковыми сигналами.

«Мандатный механизм» — включение этого флага активизирует подсистему мандатного доступа, которая при входе в систему пользователя монтирует логические диски в соответствии с уровнем допуска пользователя и метками конфиденциальности дисков (томов). (Рис. 3.29).

«Таймаут для ТМ» и «Таймаут для пароля» определяют интервал времени, отведенный для процедур начальной идентификации и аутентификации, соответственно.

4.1.4. Выход из программы

Выход из программы администрирования выполняется по клавише <Esc> из главного меню. После этого на экране снова появляется стартовое меню администратора (Рис. 3.7). Администратор может выбрать вариант загрузки или перезагрузить компьютер. При корректном входе в систему зарегистрированным ТМ-идентификатором пользователя меню не выводится, а выполняется обычная загрузка установленной операционной системы.

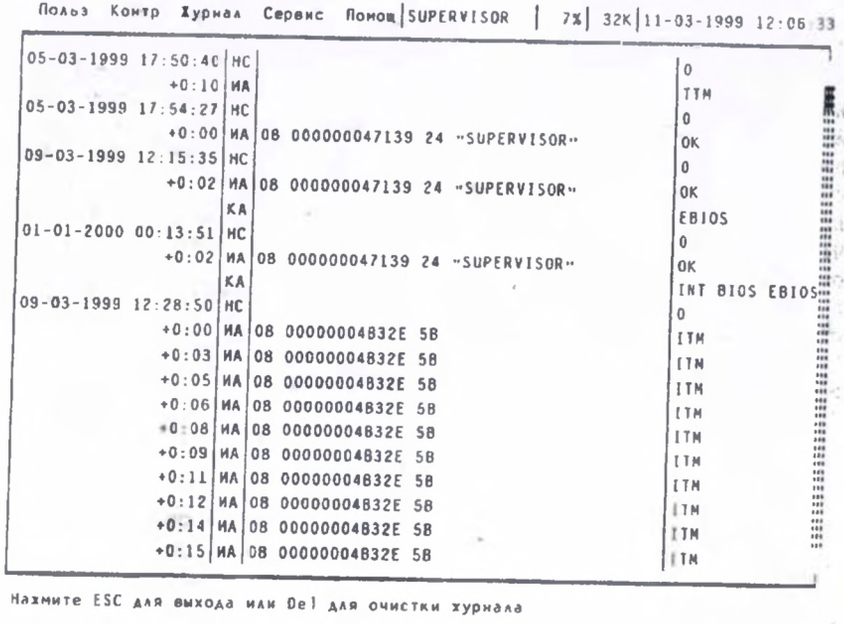


Рисунок 3.27. Системный журнал контроллера

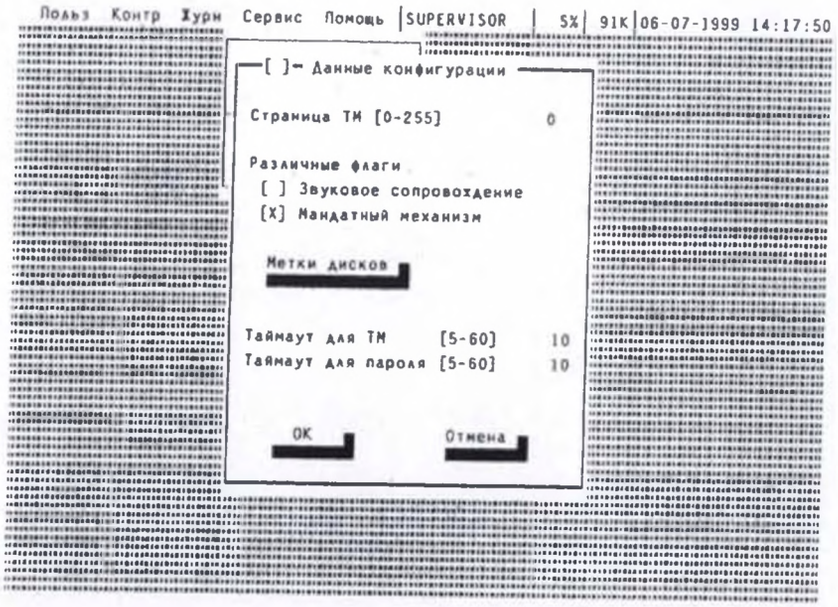


Рисунок 3.28. Окно установок параметров конфигурации

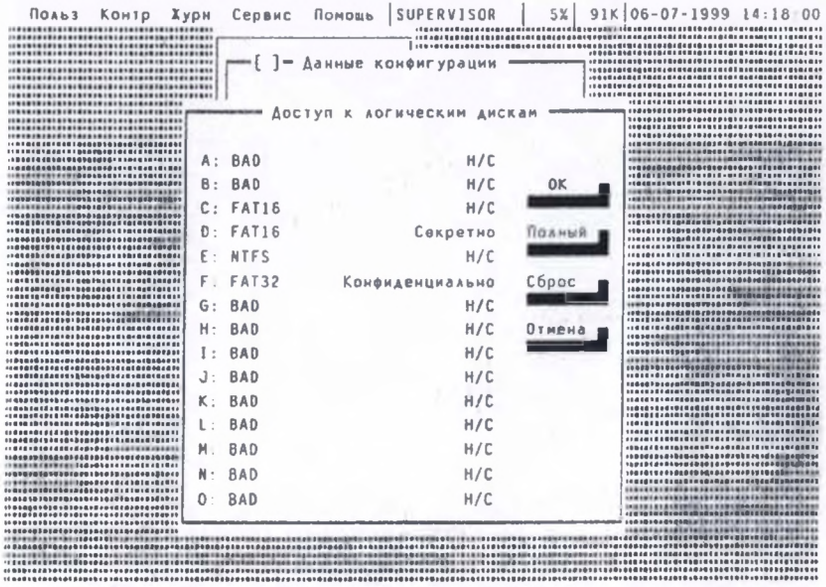


Рисунок 3.29. Установка меток конфиденциальности логических дисков

4.2. Подсистема администрирования специального ПО разграничения доступа пользователей к ресурсам ПЭВМ

Назначение

Подсистема администрирования, входящая в состав специального ПО, установленного на жесткий диск ПЭВМ предназначена для описания правил разграничения доступа (ПРД) пользователей, выполнения этих ПРД в течение всего сеанса работы данного пользователя и контроля за действиями пользователей. Данная подсистема прозрачна для пользователя и не требует от него выполнения каких-либо дополнительных процедур. Сообщения данной подсистемы выдаются только при попытке несанкционированного доступа (НСД) к защищаемым ресурсам.

Программа ACED.EXE — редактор параметров (атрибутов) дискреционного механизма разграничения доступа субъектов (пользователей) к объектам ПЭВМ или АС. Программа используется администратором БИ или субъектами, наделенными правами администратора (супервизора).

Запуск программы

На приглашение ОС набирается команда: C:\ACCORD\ACED.EXE.

Если список зарегистрированных пользователей пуст, на экран выводится главное меню (Рис. 3.30) и информация о программе.

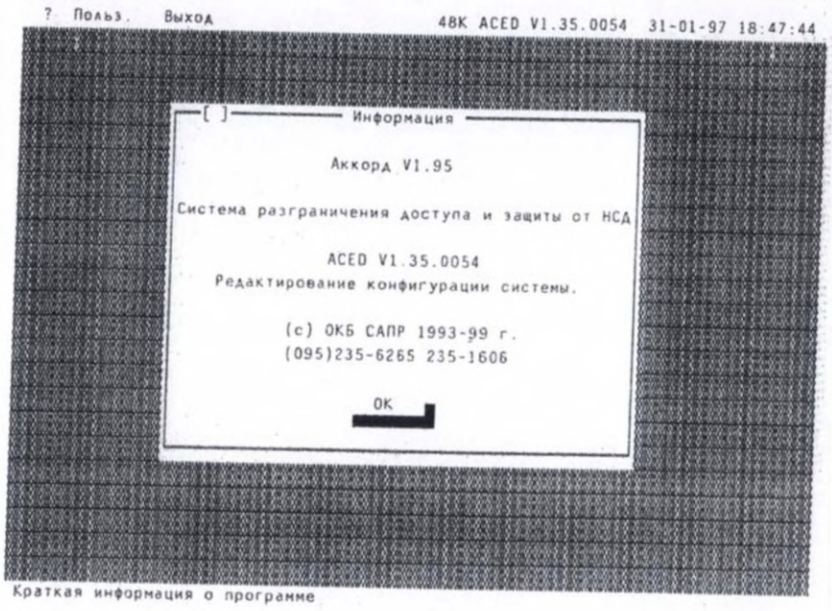


Рисунок 3.30. Главное меню

Ошибки

1. «Не загружен драйвер TMAC4P!»

Причина: не загружен драйвер TMAC4P.EXE .

Действия:

— нажать «OK», происходит выход из программы;

— загрузить драйвер TMAC4P;

— повторить запуск программы.

2. «Недопустимая версия драйвера TMAC4P!»

Причина: версии ACED.EXE и TMDRV не совпадают.

Действия:

— нажать «OK», происходит выход из программы;

— запустить программу ACED, соответствующую версии TMAC4P, либо перезагрузить ПЭВМ, загрузить драйвер TMAC4P, соответствующий версии ACED, и повторить запуск программы ACED.

Предупреждение

«Файл не найден. «C:\ACCESS.USR». Будет создан новый файл.»

Причина: отсутствует файл «C:\ACCESS.USR».

Действия: нажать «OK». Происходит создание файла пользователей.

Это предупреждение является нормальным только при первом запуске ACED после установки комплекса «Аккорд». В других случаях оно свидетельствует об ошибках в построении системы защиты информации.

Если в списке присутствует хотя бы один зарегистрированный пользователь с правами супервизора, на экран выводится главное меню (Рис. 3.31) и запрашивается ТМ-идентификатор и пароль.

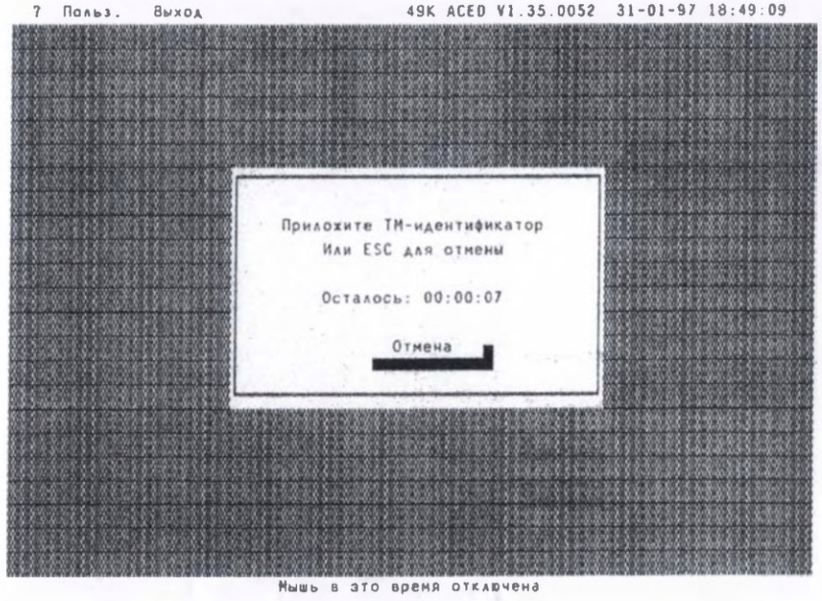


Рисунок 3.31. Запрос ТМ-идентификатора

Ошибки

1. «Незарегистрированный ТМ»

Причина: использован ТМ, незарегистрированный в системе защиты.

Действия: используйте зарегистрированный ТМ или зарегистрируйте используемый ТМ.

2. «Неверный пароль»

Причина: введен пароль несоответствующий данному ТМ-идентификатору.

Действия: введите правильный пароль.

3. Выход из ACED без предупреждения.

Причина: истекло время (Строка «Осталось XXс») для использования ТМ или ввода пароля.

Действия: перезапустите ACED. Приложите ТМ к съемнику и введите пароль в течение времени, отведенного для этих операций.

Главное меню состоит из следующих полей:

— строка команд (левая половина верхней строки),
— информационная строка (правая половина верхней строки),
— статус (HELP) — строка (нижняя срока),
— рабочее поле (все остальное пространство).

Регистрация нового пользователя

Для регистрации нового пользователя в подсистеме разграничения доступа необходимо выбрать команду <Польз.> главного меню. На экран выводится список пользователей (Рис. 3.32).

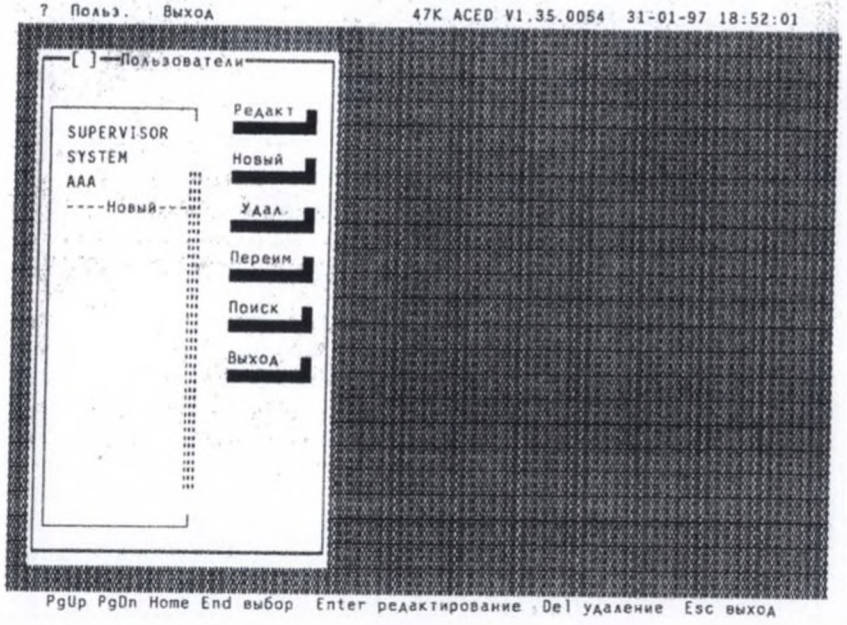


Рисунок 3.32. Список пользователей

В списке выбирается команда <Новый>.

Примечание. При первом запуске программы после установки в файле пользователей уже записаны два пользователя: "SUPERVISOR" и "SYSTEM", но для них не введены никакие атрибуты.

Первоначально необходимо зарегистрировать администратора системы. Для этого выбирается строка <SUPERVISOR> и подтверждается нажатием <Enter>. На экран выводится окно ввода-вывода «Параметры пользователя» (Рис. 3.33).

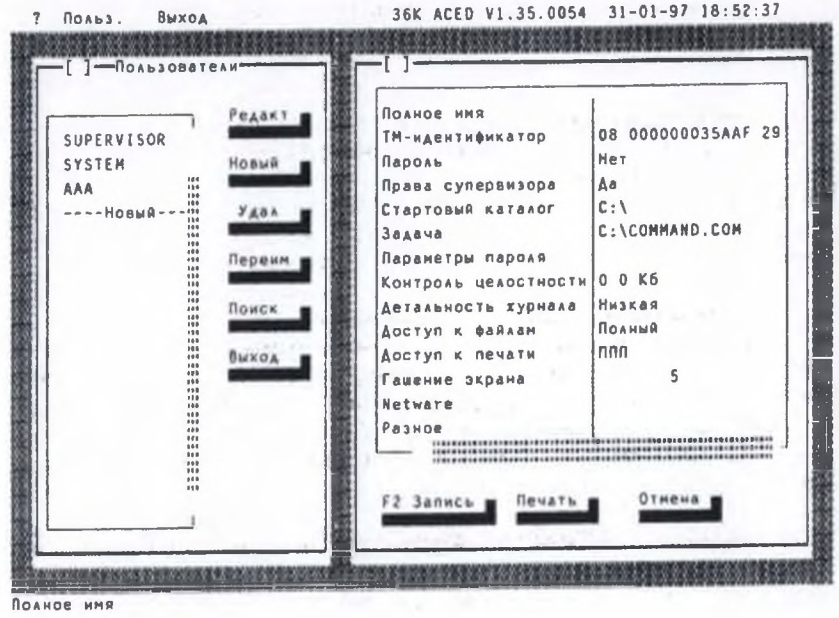


Рисунок 3.33. Параметры пользователя

В этом окне вводятся необходимые параметры пользователя. Для сохранения параметров пользователя и выхода используется команда <Запись> (<F2>). Для печати параметров выберите команду <Печать>. Для отказа от сохранения и выхода — <Отмена> (<ESC>).

Примечание. Некоторые параметры пользователя являются обязательными, без ввода которых невозможен ввод остальных, (например — «Имя»). Некоторые параметры недоступны при различных типах ТМ-идентификаторов (например «Контроль целостности» недоступен, если ТМ-идентификатор не имеет памяти). Пользователь «SYSTEM» является универсальным шаблоном для задания ограничений по доступу к ресурсам. Если запрещен доступ к некоторым дискам и каталогам для пользователя «SYSTEM», то все остальные пользователи также не получат доступа к этим дискам и каталогам.

Печать параметров пользователя

Эта команда предназначена для распечатки параметров пользователя на твердом носителе или в файле для ведения архива на твердых или магнитных носителях. При выборе команды <Печать> выводится окно (Рис. 3.34), в котором устанавливаются параметры, необходимые для печати. Для продолжения выберите <Ok>. Выводится окно (Рис. 3.35) выбора устройства для печати. Для отмены печати используется команда<Отмена>, для печати — <OK>.

Удаление зарегистрированных пользователей из списка

В ряде случаев (например, при увольнении ранее зарегистрированного пользователя) бывает необходимо удалить пользователя из списка. Для этого в подменю списка пользователей (Рис. 3.32) помечаются имена пользователей, предназначенных для удаления из списка. После нажатия клавишы <Del> и подтверждения команды будет выполнено удаление.

Редактирование параметров пользователей

В этом режиме администратор производит изменение параметров доступа к защищенным объектам. В подменю списка пользователей (Рис. 3.32) выбирается имя пользователя, параметры которого необходимо отредактировать. На экран выводится окно (Рис. 3.33) «Параметры пользователя», в котором проводятся изменения. Выйдите из окна ввода/вывода с сохранением изменений.

Примечание. Запуск ACED.EXE возможен пользователем, не имеющим статуса SUPERVISOR. В этом случае пользователь может редактировать только свои параметры <Пароль> и <Гашение экрана>.

Задание имени пользователя

Администратор должен присвоить каждому пользователю уникальное в данной вычислительной среде (отдельный компьютер или локальная сеть) имя. Рекомендуется использовать в качестве имени фамилию пользователя. Имя пользователя задается только при регистрации нового пользователя и не может редактироваться в дальнейшем.

Установка/снятие статуса супервизора у пользователя

Допускается назначение этого статуса любому выделенному пользователю. Статус супервизора позволяет запускать программу ACED.EXE и управлять параметрами прав доступа пользователей. Установка/снятие статуса супервизора осуществляется выбором строки «Права Супервизора» (Рис. 3.33).

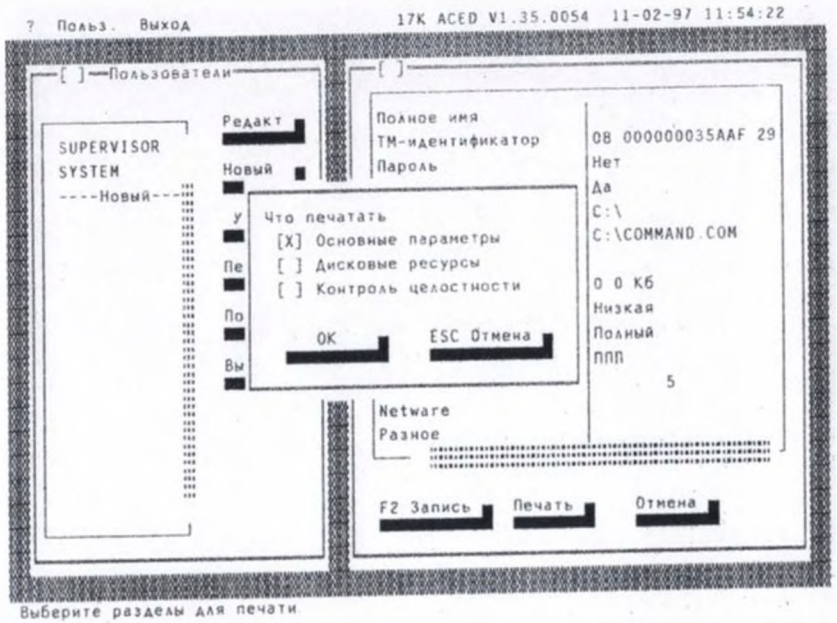


Рисунок 3.34. Выбор параметров пользователя для печати

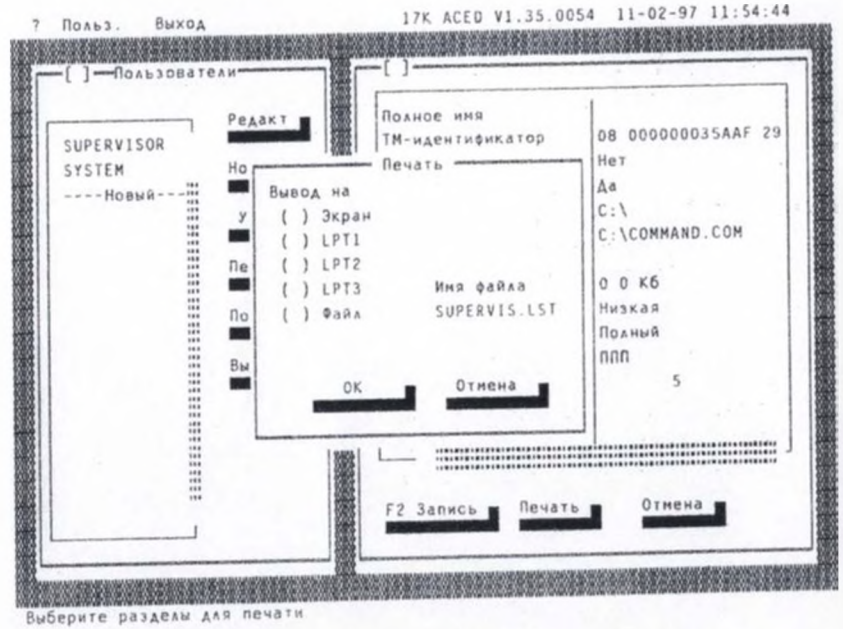


Рисунок 3.35. Выбор устройства для вывода параметров пользователя

Установка стартового каталога пользователя

Этот параметр определяет каталог, в который попадает пользователь после загрузки компьютера. Стартовый каталог должен назначаться из тех, к которым пользователь имеет доступ. В строке «Стартовый каталог» (Рис. 3.33), вводится полное имя каталога с клавиатуры или мышью путем выбора необходимого диска и каталога этого диска из списка.

Примечание. Если имя стартового каталога набирается с клавиатуры, то необходимо полностью набрать имя каталога. Например, если имеется в ввиду корневой каталог диска C:, то необходимо набрать «C:\», а не «C:».

Задание запускаемой программы пользователя

Этот параметр определяет задачу, которая запускается после загрузки компьютера. Стартовая задача может быть файлом с расширением *.BAT, *.EXE, *.COM. В окне «Запускаемая программа» (Рис. 3.33) мышью или с клавиатуры вводится путь и имя запускаемой программы.

Примечание. Если имя стартовой задачи набирается с клавиатуры, то необходимо полностью набрать имя запускаемого файла, включая имя каталога, в котором находится файл и расширение файла. Стартовая задача у пользователя может не задаваться.

Регистрация ТМ пользователя

При выборе команды <ТМ идентификатор> (Рис. 3.33) на экран выводится информация о ТМ-идентификаторе (Рис. 3.36). По команде <Новый> на запрос ТМ-идентификатора предъявите ТМ-идентификатор пользователя (Рис. 3.37).

Перерегистрация ТМ пользователя

Выберите команду <ТМ-идентификатор> (Рис. 3.33). На экран выводится окно с информацией о старом ТМ (Рис. 3.38). По команде <Новый> на запрос нового ТМ-идентификатора предъявите новый ТМ.

Примечание. Программа запрещает разным пользователям регистрировать один и тот же ТМ-идентификатор.

После предъявления ТМ-идентификатора как при регистрации нового, так и при перерегистрации существующего ТМ, на экране появляется меню генерации секретного ключа пользователя (Рис. 3.39)

Для генерации выбирается опция <Сгенерировать> и <OK>.

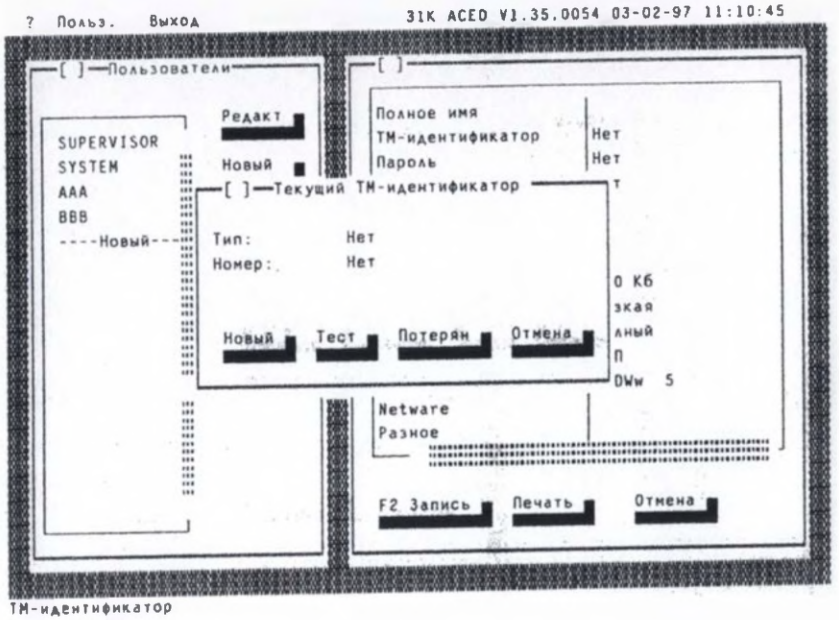


Рисунок 3.36. Регистрация нового ТМ-идентификатора

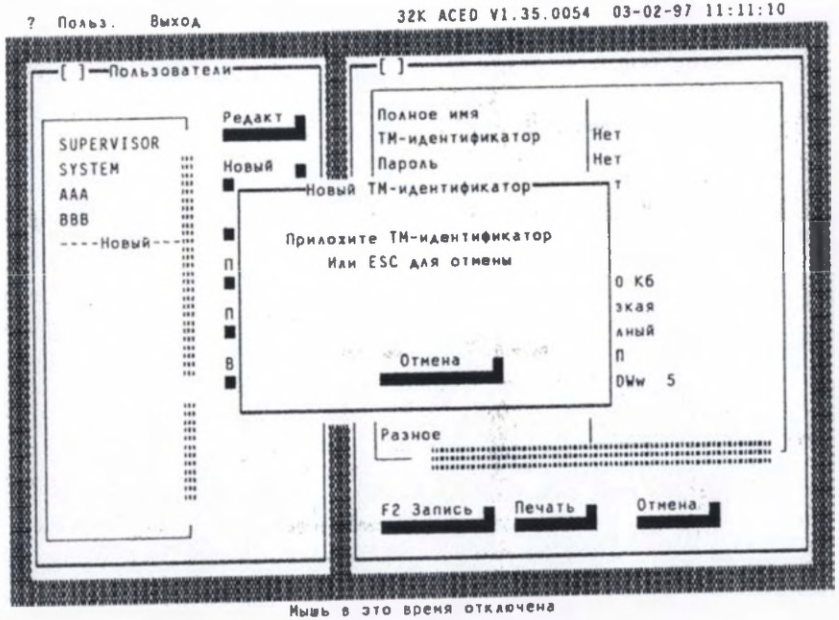


Рисунок 3.37. Запрос ТМ-идентификатора

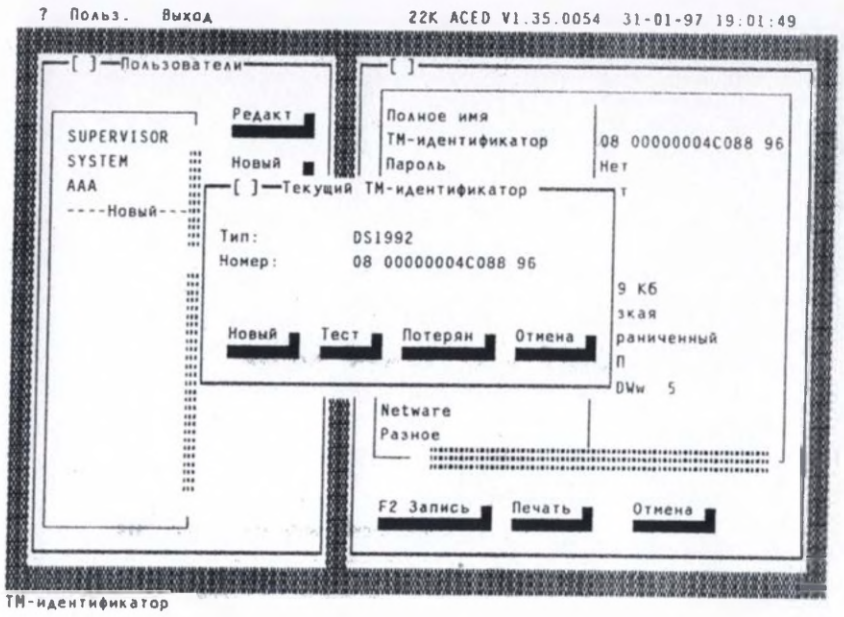


Рисунок 3.38. Перерегистрация TM-идентификатора

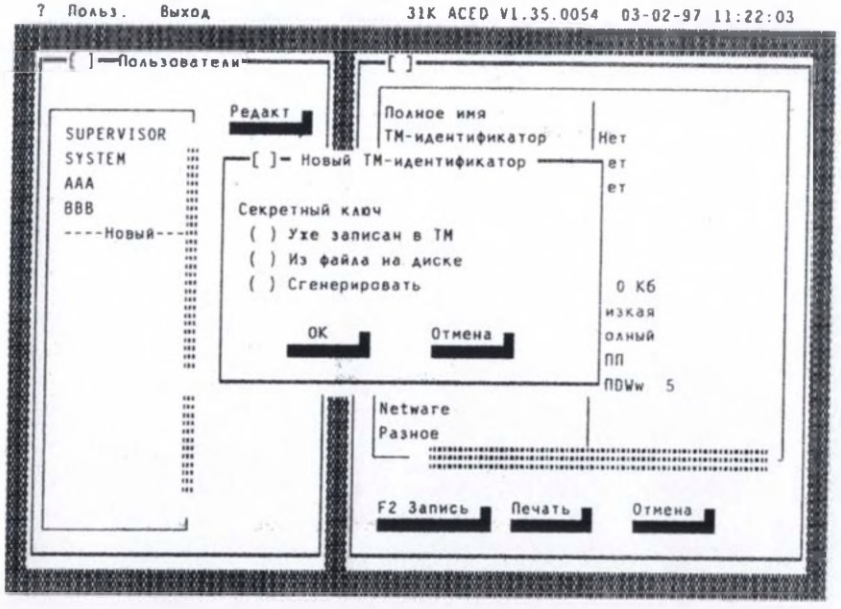


Рисунок 3.39. Генерация секретного ключа пользователя

Примечание. Секретный ключ может быть уже записан в TM, например, при перерегистрации пользователя, или при использовании системы «Аккорд» в составе криптографических комплексов, или ключ уже сгенерирован при регистрации TM в аппаратной части комплекса. В этом случае выбирается опция «Уже записан в TM».

Установка доступа к дисковым ресурсам

По команде <Доступ к файлам> в окне «Параметры пользователя» (Рис. 3.33) выводится окно прав доступа пользователя к ресурсам ПЭВМ (Рис. 3.40).

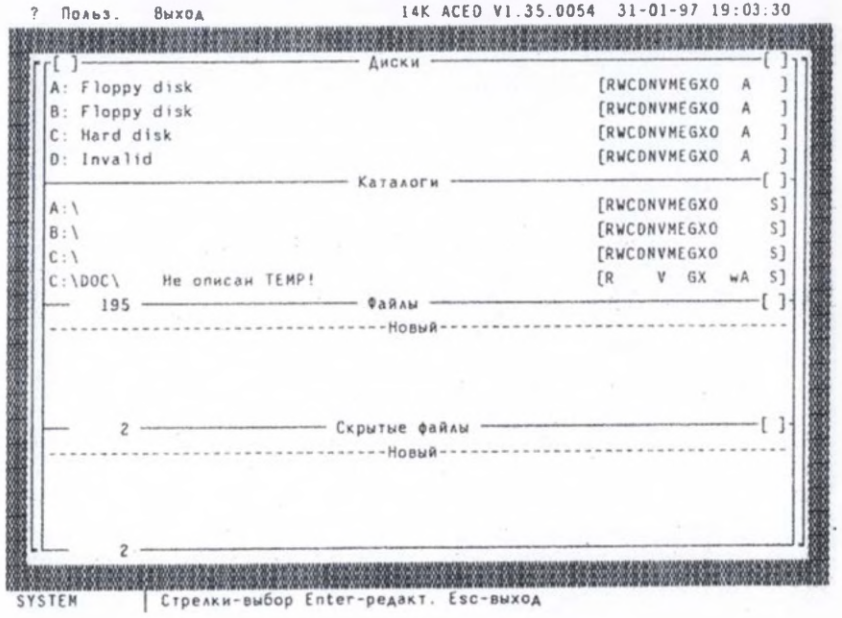


Рисунок 3.40. Доступ к дискам, каталогам, файлам

Установка доступа к дискам

В левой части окна «Диски» (Рис. 3.40) выведен перечень всех дисков, в правой — атрибуты доступа к дискам (в квадратных скобках). В окне «Диски» выберите строку с названием диска. Открывается окно корректировки атрибутов доступа к выбранному диску (Рис. 3.41)

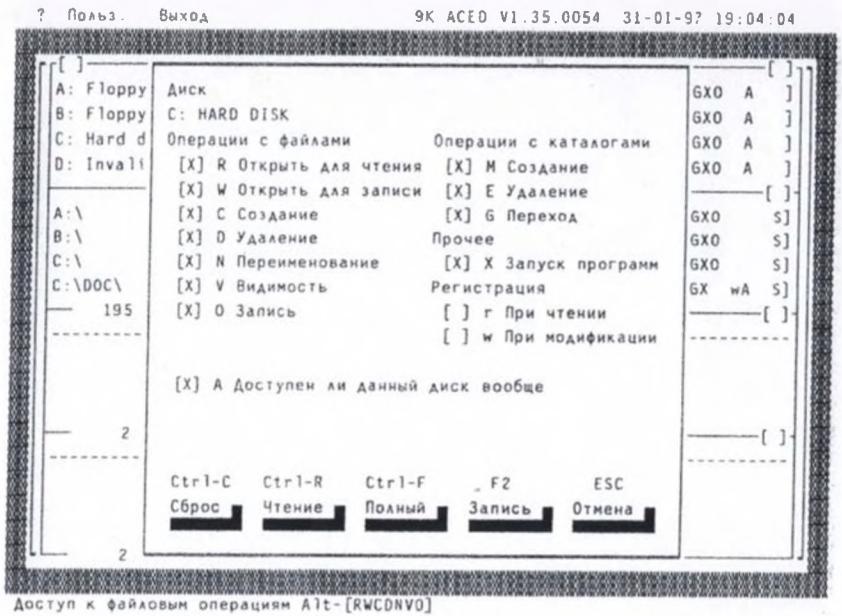


Рисунок 3.41. Атрибуты доступа к диску

Мышью или клавиатурой задаются необходимые атрибуты доступа к диску. Для выхода из режима с сохранением применяется команда <Запись>, без сохранения — <Отмена>. После выхода из режима в строке диска в квадратных скобках указан статус диска. Буквенное сокращенное выражение соответствует параметрам окна диска (Рис. 3.41). Атрибуты доступа к диску имеют различный приоритет по отношению один к другому, параметры с более низким приоритетом игнорируются, если не установлен атрибут с более высоким приоритетом. Атрибуты доступа к диску разделены на 5 групп.

1. Операции с файлами:

R — разрешение на открытие файлов только для чтения.

W — разрешение на открытие файлов для записи.

C — разрешение на создание файлов на диске.

D — разрешение на удаление файлов.

N — разрешение на переименование файлов.

V — видимость файлов. Позволяет делать существующие файлы невидимыми для программ. Этот атрибут имеет более высокий приоритет, чем R,W,D,N,O.

O — эмуляция разрешения на запись информации в файл. Этот атрибут имеет более низкий приоритет, чем W (открыть для записи).

2. Операции с каталогом:

M — создание каталогов на диске.

E — удаление каталогов на диске.

G — разрешение перехода в этот каталог.

3. Прочее:

X — разрешение на запуск программ.

4. Регистрация:

r — регистрируются все операции чтения файлов диска в журнале.

w — регистрируются все операции записи файлов диска в журнале.

5. Операции с диском:

A — доступ к диску. Этот атрибут имеет более высокий приоритет, чем атрибуты групп 1, 2, 3, 4.

Примечание. Для группового манипулирования атрибутами диска пользуйтесь командами <Сброс> (сбрасывает все параметры), <Чтение> (устанавливает атрибуты R, V, G, X, A), <Полный> (устанавливает все атрибуты, кроме атрибутов группы «Регистрация»).

Установка доступа к каталогам

В окне «Каталоги» (Рис. 3.40) выбирается строка с названием каталога, который необходимо описать. Если Вы хотите определить каталог, который ранее не был определен, то необходимо выбрать команду <Новый> и ввести его название. Выводится окно атрибутов каталога (Рис. 3.42).

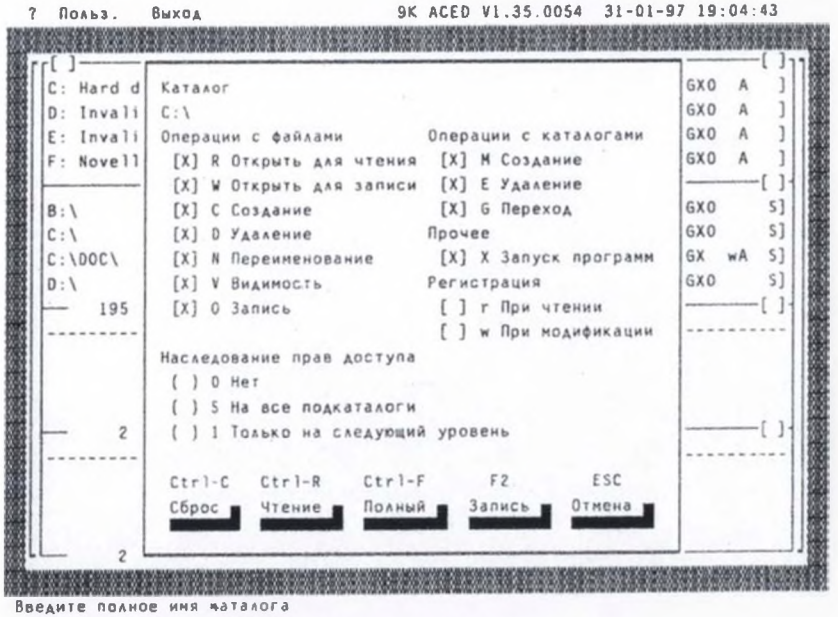


Рисунок 3.42. Атрибуты доступа к каталогу

Для удобства работы в окне «Каталог» можно пользоваться клавишей <F4>, при этом на экран выводится все дерево каталогов для всех дисков (Рис. 3.43).

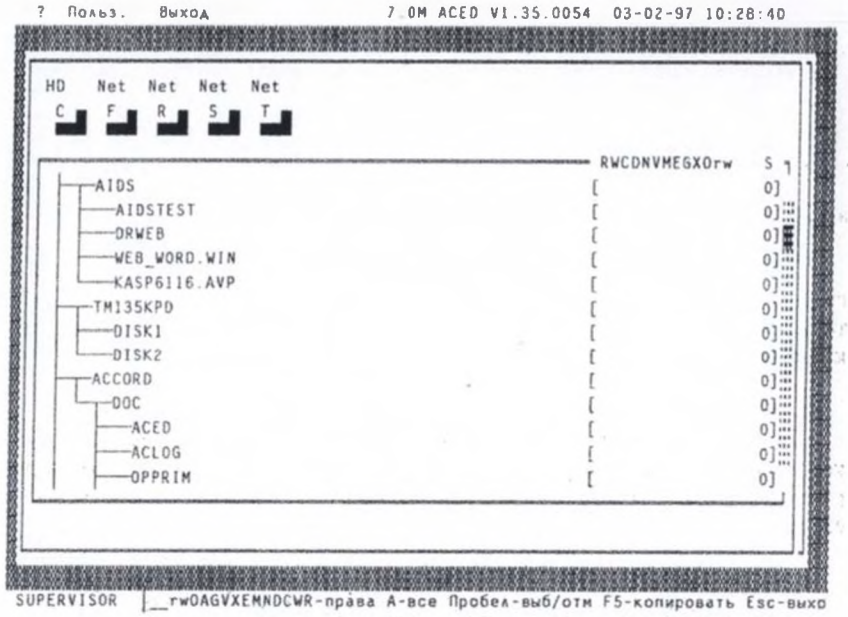


Рисунок 3.43. Выбор каталога из списка

Ошибки

1. Не хватает памяти

Причина: недостаточно оперативной памяти для разворачивания дерева каталогов.

Действия: освободите оперативную память. Если количество директорий и поддиректорий на диске слишком велико (более 1000), то вы не получите возможности пользоваться клавишей <F4>

Предупреждение

Не описан TEMP Причина: Не описан каталог TEMP.

Действия: установите полный доступ к каталогу, на который установлено устройство TEMP в файле AUTOEXEC.BAT. Например, если в файле AUTOEXEC.BAT есть строка: SET TEMP=C:\WINDOWS\TEMP, установите полный доступ к каталогу C:\WINDOWS\TEMP. Этот каталог может использоваться рядом программ (например, для создания временных файлов). По свойствам и приоритетам параметры каталога совпадают с параметрами диска за исключением группы V «Наследование прав доступа». Этот параметр

может принимать три значения: S — параметры доступа наследуются существующими и созданными в дальнейшем подкаталогами всех уровней текущего каталога, т.е. для них устанавливаются те же параметры доступа; 1 — параметры доступа текущего каталога наследуется только подкаталогами следующего уровня; 0 — параметры доступа текущего каталога не наследуется подкаталогами.

Примечание. Приоритет одноименных параметров диска выше, чем параметров каталога. Например, если параметр «открыть файл с записью» для диска отсутствует, то на этом диске нельзя будет открыть файл для записи даже в тех каталогах, в которых этот параметр установлен. Таким образом для того, чтобы параметр для подкаталога «работал» он должен быть установлен для содержащего его диска. Одноименные же параметры каталогов и их подкаталогов, не связанных наследованием, действуют независимо один от другого. Например, если параметр для каталога отсутствует, а для его подкаталога, не связанного наследованием, присутствует, то этот параметр «работает» в подкаталоге.

Установка доступа к файлам

В окне «Файлы» (Рис. 3.40) выбирается строка с нужным именем файла (если необходимый файл отсутствует в списке, выбирается команда <Новый> и вводится имя файла).

Примечание. При вводе имени файла можно пользоваться простым групповым обозначение имени файла. Например, можно *.bak, *.exe и т.п., нельзя *a.exe, a*.bat, &a.dat, a.* и т.п. Выводится окно для определения параметров доступа к файлу (Рис. 3.44), в котором устанавливаются атрибуты доступа к файлам.

Примечание. Приоритет одноименных параметров для файлов, установленных в окне «Файлы» выше, чем их приоритет, установленный для диска и каталога. Например, если параметр «открыть файл с записью» для диска и каталога отсутствует, а для файла присутствует, то данный файл можно открыть для записи. Поэтому список файлов в окне «Файлы» называется «белым».

Установка доступа к скрытым файлам

Данный режим используется для формирования «черного» списка файлов и устройств, т.е. таких файлов, которые недоступны пользователю, даже если они находятся в доступных каталогах. В окне «Скрытые файлы» (Рис. 3.40) следует выбрать строку с нужным именем файла (если необходимый файл отсутствует в списке — вводится имя файла с помощью команды <Новый>). Выводится окно для установки атрибутов доступа к скрытому файлу.

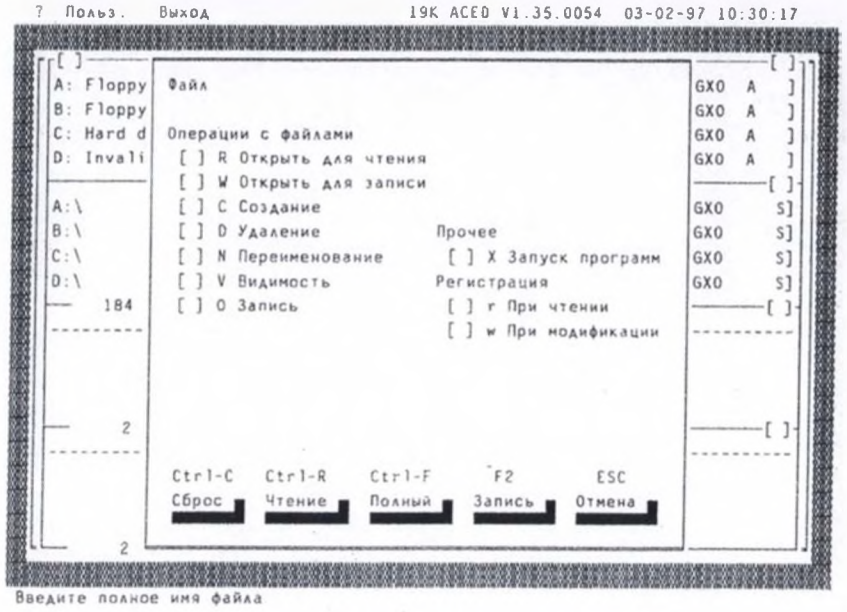


Рисунок 3.44. Атрибуты доступа к файлу

В этом окне можно установить атрибуты доступа к скрытым файлам.

Атрибуты доступа к скрытым файлам:

R — разрешение чтения секторов.

W — разрешение записи секторов.

Кроме того, в качестве скрытого файла можно задавать имена драйверов. Например, для принтеров /PRN: или /LPT1:, для драйвера 32-битного доступа к файлам — /IFSHLP. В этом случае обеспечивается ограничение доступа к тем устройствам, которые обслуживаются перечисленными в списке драйверами. Независимо от состояния параметров R и W, все файлы и драйверы «черного» списка недоступны для любых операций пользователя. Если эти параметры установлены — доступ к файлам контролируется DOS, если — нет, то — DOS и BIOS. Во втором случае работа с диском несколько замедляется.

Установка гашения экрана

Гашение экрана используется для временного отключения экрана и доступа к компьютеру по истечении времени паузы в работе пользователя на клавиатуре, либо по нажатием пользователем клавиши «Гашение» <Ctrl><F12> при кратковременном перерыве работы. После срабатывания гашения экрана

вернуться в рабочий режим можно только с помощью ТМ-идентификатора пользователя. Команда <Гашение экрана> выбирается в списке параметров пользователя (Рис. 3.33). Выводится окно «Параметры Screen-saver» (Рис. 3.45).

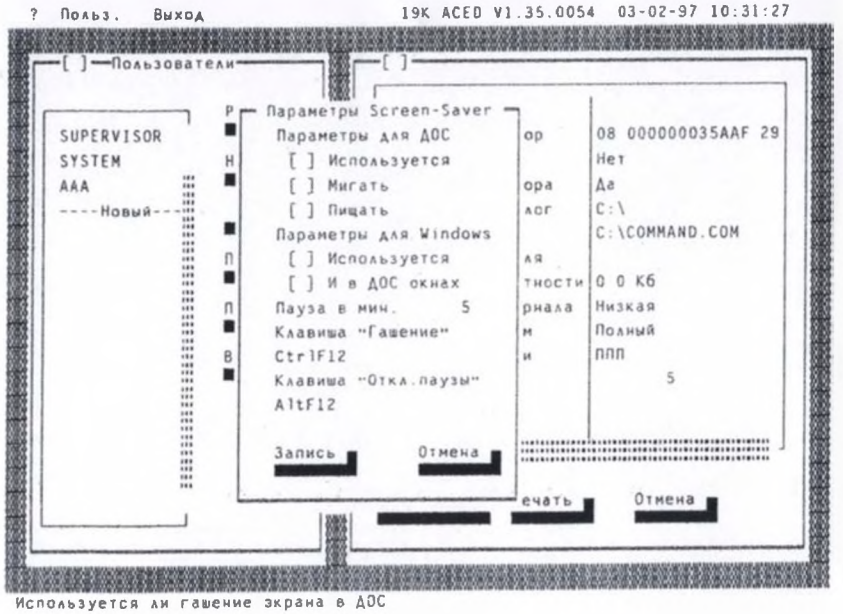


Рисунок 3.45. Параметры гашения экрана

В этом окне устанавливаются параметры гашения. Для выхода из режима с сохранением параметров используется команда <Запись>, без сохранения — <Отмена> или <Esc>. Если необходимо задать режим гашения экрана, устанавливается параметр «Используется» (этот параметр имеет наиболее высокий приоритет), затем, если нужно, параметры «Мигать» (мигание индикаторов <Num Lock»>, <Caps Lock»> и <Scroll Lock»> в режиме гашения), «Пищать» (звуковые сигналы в режиме гашения), время гашения (переход в режим гашения экрана, если клавиатура и мышь не используется в течение установленного времени). Также можно установить клавиши принудительного включения режима («Гашение») и отключения контроля интервала времени («Откл. паузы»).

Примечание. При работе в MS-DOS следует отключить использование любых программ гашения экрана (например, Screen-saver в Norton Commander).

Использование подобных программ совместно с комплексом «Аккорд» приводит к некорректной работе компьютера, т.к. одни и те же ресурсы используются разными резидентными программами. При работе в Windows, напротив следует включить использование стандартного Screen-saver, т.к. «Аккорд» использует его ресурсы. Следует только установить интервал времени срабатывания Screen-saver в Windows большим, чем в «Аккорде».

Установка детальности протокола работы пользователей

Во время работы каждого пользователя ведется журнал, в котором регистрируются действия, которые он совершает. Администратору рекомендуется в текущей работе использовать низкую детальность ведения журнала. Среднюю и высокую детальность следует использовать при изучении работы вновь используемых задач с целью определения особенностей задачи, а именно: создание новых постоянных и временных каталогов и файлов, используемых прерываний и т.д. Выбирается команда <Детальность журнала> (Рис. 3.33) Нажатием левой клавиши мыши или клавишей <Enter> устанавливается детальность ведения протокола работ данного пользователя.

Примечание. Уровни детальности.

Низкая — регистрация входа/выхода в/из системы, нарушения при входе в систему, запуск задач.

Средняя — то же, а также операции доступа к файлам и каталогам.

Высокая — то же, что и при средней детальности, а также прямой доступ к диску и выполнение функций просмотра каталогов.

Установка доступа пользователя к выводу информации на принтер

Установка осуществляется командой <Доступ к печати> (Рис. 3.33). Нажатием левой клавиши мыши или клавишей <Enter> устанавливается нужный уровень доступа к печати данного пользователя.

Уровни доступа

Полный — полный доступ пользователя к выводу на печать.

Нет — пользователь не имеет доступа к выводу на печать.

Контроль целостности

Программа позволяет контролировать целостность файлов, установленных администратором. Установка контроля целостности возможна только для пользователей, которые имеют ТМ-идентификатор с памятью. Для установки режима применяется команда <Контроль целостности> (Рис. 3.33). Выводится окно контроля целостности файлов (Рис. 3.46).

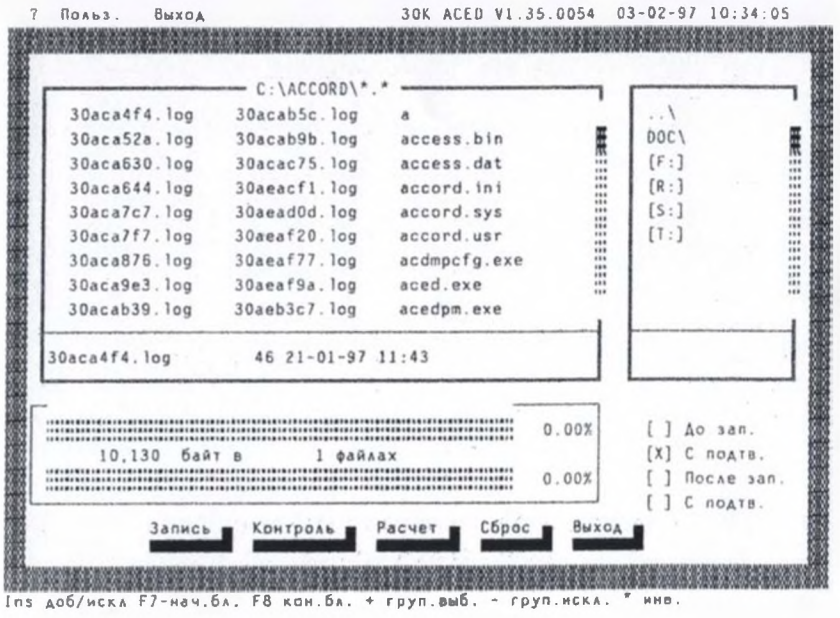


Рисунок 3.46. Контроль целостности файлов

1. В окне «Каталоги» выбирается нужный каталог, при этом на экран выводится содержимое каталога. Мышью (левая кнопка) или клавиатурой (стрелки и <пробел> или <Ins>) помечаются необходимые файлы, (Повторная отметка исключает файлы из списка).

Примечание. С помощью клавиатуры возможна групповая пометка файлов.

Для этого курсор помещается на начало помечаемого блока, нажимается клавиша <F7>, затем курсор помещается на конец блока и нажимается <F8>. Все файлы блока будут отмечены.

<+> — позволяет пометить группу файлов по имени.

<->— исключает файлы по имени.

<*>— инверсия помечает не помеченные файлы и исключает помеченные.

<Сброс> — очищает список помеченных файлов во всех каталогах.

2. Особенности процесса контроля целостности устанавливаются в окне «Процесс», а именно <До зап.> — контроль целостности до запуска стартовой задачи. <С подтв.> — запрос подтверждения контроля целостности до запуска стартовой задачи. <После зап.> — контроль целостности после завершения

стартовой задачи. <С подтв.> — запрос подтверждения контроля целостности после завершения стартовой задачи.

3. Выполняется расчет хэш-функции путем выбора команды <Расчет>.

4. Осуществляется запись хэш-функции в ТМ-идентификатор путем выбора команды <Запись>.

5. Выполняется контроль хэш-функции путем выбора команды <Контроль>.

Выход из режима — команда <Выход>.

Выход из программы

В главном меню выбирается команду <Выход> и подтверждается выход из программы.

Программа ACED.EXE является лишь редактором параметров доступа пользователя к объектам доступа. Разграничение доступа пользователей к ресурсам компьютера реализуется программой ACRUN.EXE, которая использует атрибуты доступа, подготовленные Администратором.

5. РЕКОМЕНДАЦИИ ПО УПРАВЛЕНИЮ МЕХАНИЗМАМИ ЗАЩИТЫ КОМПЛЕКСА «АККОРД»

Настоящий раздел является руководством по управлению механизмами защиты программно-аппараного комплекса защиты информации от НСД «Аккорд» и предназначен для конкретизации задач и функций должностных лиц организации (предприятия, фирмы), планирующих и организующих защиту информации в системах и средствах информатизации на базе ПЭВМ с применением комплекса. Приведены основные функции администратора безопасности информации, порядок установки прав доступа пользователей к информационным ресурсам, организации контроля работы ПЭВМ (АС) с внедренными средствами защиты и другие сведения необходимые для управления защитными механизмами комплекса.

Для лучшего понимания и использования защитных механизмов комплекса рекомендуется предварительно ознакомиться с комплектом эксплуатационной документации на комплекс, а также принять необходимые защитные организационные меры, рекомендуемые в документации. Применение защитных мер комплекса «Аккорд» должно дополняться общими мерами предосторожности и физической безопасности ПЭВМ (АС).

Программно-аппаратный комплекс защиты информации от несанкционированного доступа (ПАК СЗИ НСД) «Аккорд» — это простой, но чрезвычайно эффективный комплекс технических и программных средств, используя который можно надежно защитить информацию на ПЭВМ (в АС) без переделки ранее приобретенных программных средств. СЗИ «Аккорд» обеспечивает для пользователя «прозрачный» режим работы, при котором пользователь, как правило, не замечает внедренной системы защиты. Таким образом, дополнительная нагрузка, связанная с эксплуатацией СЗИ, не ложится на пользователя, а замыкается на администраторе безопасности информации (БИ). В этой связи для обеспечения эффективности работы АС администратор БИ обязан досконально изучить и правильно применять возможности системы защиты информации на базе СЗИ «Аккорд».

Не умаляя достоинств комплекса «Аккорд», прежде всего сильной аппаратной поддержки большинства защитных механизмов, надо сказать, что комплекс не может решить все проблемы по созданию комплексной защиты информационных систем. Надо четко понимать, что комплекс «Аккорд» — это лишь хороший инструмент, позволяющий службе безопасности информации (СБИ) значительно проще и надежнее решать одну из стоящих перед ней задач — защиту от НСД к ПЭВМ и информационным ресурсам, разграничение доступа к объектам доступа, обеспечение целостности программ и данных в соответствии с принятой в организации (предприятии, фирме и т.д.) политикой информационной безопасности.

Использование ПЭВМ с внедренными средствами защиты комплекса не требует изменения существующего программного обеспечения, необходимы лишь квалифицированное применение комплекса (правильная установка, настройка и эксплуатация в соответствии с принятыми на предприятии ПРД) и обеспечение некоторой организационной поддержки.

Как показывает практика довольно длительного применения комплекса, часто трудности заключаются в отсутствии у большинства пользователей (организаций, фирм и т.д.) установленного порядка и четких правил разграничения доступа к защищаемым ресурсам. Поэтому, именно выяснение того, что и кому в ПЭВМ (АС) доступно, а что нет, и какие действия с доступными ресурсами разрешено выполнять, а какие нет, является основным содержанием необходимой организационной поддержки. Для выполнения этих задач, а также для обеспечения непрерывной организационной поддержки работы применяемых технических средств защиты информации, в том числе и комплекса «Аккорд», необходима специальная служба (администрация) безопасности информации (СБИ), в небольших организациях и подразделениях — администратор безопасности информации (администратор БИ). На СБИ (администратора БИ) возлагаются задачи по осуществлению единого руководства, организации применения средств защиты и управления ими, а также контроля за соблюдением всеми категориями пользователей требований по обеспечению безопасности программно-информационных ресурсов автоматизированных систем. Правовой статус СБИ, обязанности и некоторые рекомендации по организации СБИ приведены в Приложении 4.

Внимание! Применение комплекса «Аккорд» совместно с сертифицированными программными СКЗИ и средствами разграничения доступа позволяет значительно снизить нагрузку на организационные меры, определяемые условиями применения этих средств, при этом класс защищенности не снижается.

Для эффективного применения комплекса «Аккорд» и поддержания уровня защищенности ПЭВМ (АС) необходимы:

— физическая охрана ПЭВМ и ее средств, в т.ч. обеспечение мер по неизвлечению контроллера комплекса;

— использование в ПЭВМ (АС) технических и программных средств, сертифицированных как в Системе ГОСТ Р, так и в системе Государственной системы безопасности информации (ГСЗИ);

— периодическое тестирование средств защиты комплекса;

— разработка и ведение учетной и объектовой документации (инструкция администратора, инструкций пользователей, журнал учета идентификаторов и отчуждаемых носителей пользователей и др.).

Все разработанные учетные и объектовые документы должны быть согласованы, утверждены у руководства и доведены до сотрудников (пользователей). Это необходимо для того, чтобы План защиты организации (предприятия, фирмы и т.д.) и действия СБИ (администратора БИ) получили юридическую основу.

5.1. Содержание работы администратора БИ по применению комплекса «Аккорд»

Основным содержанием работы администратора БИ по применению комплекса «Аккорд» являются следующие мероприятия:

— планирование применения комплекса;

— организация установки комплекса и настройка его защитных средств в соответствии с установленными ПРД;

— эксплуатация ПЭВМ (АС) с внедренным комплексом, в т.ч., организация контроля за правильностью применения защитных механизмов комплекса;

— снятие защиты.

5.1.1. Планирование применения комплекса

Планирование применения комплекса «Аккорд» осуществляется с учетом общей политики обеспечения безопасности в организации (на предприятии, фирме и т.д.). Основное содержание этой политики отражается в Плане защиты — документе, определяющем подходы к защите информации и фиксирующем состояние защищаемой автоматизированной системы. В части защиты информации в него включаются сведения о характере и составе обрабатываемой информации, составе технических и программных средств АС (ПЭВМ), возможных угрозах системе и наиболее вероятных способах их реализации, описание выбранных методов и средств защиты от этих угроз, правила разграничения доступа к информационным ресурсам и другие вопросы.

Для настройки средств защиты комплекса «Аккорд» в соответствии с принятыми в организации (фирме) ПРД администратору БИ необходимо предварительно выяснить и отразить в плане защиты следующие характеристики защищаемой системы (ПЭВМ):

— перечень задач, решаемых структурными подразделениями организации (сотрудниками) с использованием АС (ПЭВМ);

— детальный перечень используемых при решении каждой задачи программ;

— детальный перечень используемых при решении каждой задачи (совместно используемых несколькими задачами) данных с указанием мест их размещения, режимов обработки и правил доступа к ним;

— конфигурацию ПЭВМ с указанием перечня используемых технических средств (принтеров, сканеров и т.д.) и их характеристик;

— при использовании комплекса для защиты ЛВС - подробный перечень имеющихся в защищаемой сети серверов, рабочих станций и т.д. с указанием их состава, конфигурации, характеристик используемых технических средств и мест их размещения;

— перечень размещенных на ПЭВМ (каждой рабочей станции ЛВС и каждом файловом сервере) системных и прикладных программ, файлов и баз данных;

— перечень установленных на ПЭВМ (рабочих станциях и серверах) программных средств защиты (СКЗИ и СЗИ НСД);

— списки пользователей ПЭВМ (АС) с указанием решаемых ими задач из общего перечня задач и предоставленных им (в соответствии с их обязанностями) полномочий по доступу в ПЭВМ (рабочим станциям, серверам ЛВС) и информационным ресурсам.

На этапе организации системы защиты и применения комплекса «Аккорд» необходимо, исходя из целей защиты ПЭВМ (АС) и ее специфики, разработать ряд документов, определяющих:

— порядок и правила предоставления, изменения и утверждения конкретным должностным лицам необходимых полномочий по доступу к ресурсам ПЭВМ (АС);

— порядок организации учета, выдачи, использования и хранения съемных магнитных носителей информации, содержащих резервные копии программ и данных и т.п.;

— порядок обновления используемых версий, приема в эксплуатацию новых системных и прикладных программ на защищаемых ПЭВМ (рабочих станциях, серверах) — кто обладает правом разрешения таких действий, кто осуществляет, кто контролирует и что при этом они должны делать — гарантирующий их безопасность и отсутствие РПВ;

— порядок использования, хранения и контроля целостности программных продуктов;

— порядок замены и ремонта средств вычислительной техники на защищаемой ПЭВМ (в АС) — кто обладает правом разрешения таких действий, кто их осуществляет, кто контролирует и что при этом они должны делать;

— порядок и периодичность анализа системных журналов регистрации и принятия мер по зарегистрированным несанкционированным действиям пользователей ПЭВМ (АС).

Для реализации впоследствии возможности создания любому пользователю изолированной программной среды необходимо, чтобы вышеназванные документы и правила разграничения доступа к ресурсам гарантировали:

— исключение возможности доступа непривилегированных пользователей к находящихся в ПЭВМ (АС) инструментальным и технологическим программам, с помощью которых можно проанализировать работу СЗИ и предпринять попытки их «взлома» и обхода, внедрения разрушающих программных воздействий (РПВ);

— исключение возможности разработки программ в защищенном контуре ПЭВМ (системы);

— исключение возможности несанкционированной модификации и внедрения несанкционированных программ;

— жесткое ограничение круга лиц, обладающими расширенными или неограниченными полномочиями по доступу к защищаемым ресурсам.

С учетом вышесказанного необходимо также разработать и внести необходимые изменения во все организационно-распорядительные документы (положения о подразделениях, функциональные обязанности сотрудников, инструкции пользователей и т.д.) по вопросам информационной безопасности и правила работы на ПЭВМ (в АС) с внедренными средствами защиты комплекса, действиям в случае возникновения нештатных ситуаций.

5.1.2. Установка и настройка комплекса «Аккорд»

Администратор БИ организует установку комплекса исходя из принятой в организации политики информационной безопасности и осуществляет контроль за качеством ее выполнения. Порядок установки и настройки комплекса в соответствии с конфигурацией ПЭВМ(АС) содержится в «Руководстве по установке комплекса» (4012-001-11443195-95 98).

В настоящем разделе рассматривается порядок установки и настройки защитных средств комплекса в соответствии с правилами разграничения доступа (ПРД) к информации, принятыми в организации (на предприятии, фирме и т.д.). Содержанием этой работы является назначение пользователям ПЭВМ (АС) полномочий по доступу к ресурсам в соответствии с разработанными (и возможно уточненными в ходе настройки комплекса) организационно-распорядительными документами.

Требуемые полномочия назначаются пользователям путем соответствующей настройки (См. раздел 4.2):

— средств идентификации и аутентификации пользователей, с учетом необходимой длины пароля и времени его жизни, ограничением времени доступа субъекта к ПЭВМ (АС);

— механизма управления доступом к ресурсам с использованием атрибутов доступа, которые устанавливаются администратором БИ в соответствие каждой паре «субъект доступа - объект доступа» при регистрации пользователей исходя из их функциональных обязанностей;

— средств контроля целостности критичных с точки зрения информационной безопасности программ и данных;

— механизма функционального замыкания программной среды пользователей средствами защиты комплекса;

— механизмов применения специальных процедур печати, управления стандартными процедурами печати, процедурами ввода/вывода на отчуждаемые носители информации;

— дополнительных защитных механизмов, таких как блокирование экрана и клавиатуры, механизма контроля информационной безопасности объема конфиденциальной информации, выводимого на внешние устройства ПЭВМ, а также подачи соответствующих звуковых и визуальных сигналов при попытках несанкционированного доступа к ПЭВМ (АС) и ее ресурсам.

5.1.3. Эксплуатация комплекса

При эксплуатации комплекса администратор БИ решает следующие задачи:

— поддерживает средства защиты комплекса в работоспособном состоянии и контролирует правильность их работы;

— производит изменения в настройке средств защиты комплекса на основании и в полном соответствии с изменениями правил разграничения доступа. Они могут быть вызваны различными причинами, например, изменением состава пользователей, их должностных и функциональных обязанностей, расширением номенклатуры используемых технических и программных средств, задач и т.п.;

— осуществляет текущий контроль за работой пользователей ПЭВМ с установленными средствами защиты;

— анализирует содержимое журнала регистрации событий, формируемого средствами комплекса и на этой основе вырабатывает предложения по совершенствованию защитных механизмов, реализуемых средствами комплекса, принимает необходимые меры по совершенствованию системы защиты информации в целом.

Внимание! Непрерывная организационная поддержка функционирования средств защиты комплекса предполагает обеспечение строгого соблюдения всеми пользователями требований СБИ (администратора БИ).

5.2 . Некоторые особенности действия атрибутов и подготовки ПРД

Защитные механизмы в СЗИ «Аккорд» реализованы на основании требований, приведенных в [8,9] и рассмотренных в главе 2. Ниже остановимся лишь на некоторых особенностях их реализации.

Правила разграничения доступа (ПРД) в СЗИ НСД устанавливаются присвоением объектам доступа атрибутов доступа. Установленный атрибут означает, что определяемая атрибутом операция может выполняться над данным объектом.

Как отмечено в главе 2, в СЗИ «Аккорд» применяются следующие атрибуты:

R — открытие файлов для чтения;
W — открытие файлов для записи;
O — подмена атрибута R атрибутами RW на этапе открытия файла;
C — создание файлов;
D — удаление файлов;
N — переименование файлов и подкаталогов;
V — видимость файлов;
M — создание подкаталогов;
E — удаление подкаталогов;
G — доступность данного каталога (т.е. переход к нему);
X — исполнение задач;
S — наследование подкаталогами атрибутов каталога.

Установленные атрибуты определяют важнейшую часть ПРД пользователя. От правильности выбора и установки атрибутов во многом зависит эффективность работы СЗИ. В этой связи администратор службы безопасности информации должен ясно представлять, от чего и как зависит выбор атрибутов, назначаемых объектам, к которым имеет доступ пользователь. Как минимум, необходимо изучить принцип разграничения доступа с помощью данных атрибутов, а также особенности работы программных средств, которые будут применяться пользователем.

В частности, следует иметь ввиду следующее.

Часто перед нормальной попыткой открыть существующий файл программы выполняют просмотр содержимого каталога. В этом случае, если атрибут «V» не установлен, функции FindFirst и FindNext возвратят результат «ошибка». В некоторых случаях (при некорректной установке атрибутов) это может быть источником коллизий. Отметим, что все атрибуты, кроме атрибута «G», относятся к содержимому каталога. Атрибут «G» относится к собственно каталогу. При написании программ хорошим тоном считается, когда программа

создает «временные» файлы в каталоге, имя которого хранится в переменной окружения TEMP. Например, Windows 3.1 и все программы для Windows фирмы Microsoft вначале пытаются найти переменную окружения TEMP, и при успехе используют данный каталог для размещения промежуточных данных. Применительно к «Аккорд» можно рекомендовать следующее.

Если у вас задана переменная окружения TEMP, то доступ к этому каталогу должен быть максимально полным (можно исключить лишь атрибут «X»).

Отдельно стоит рассмотреть применение атрибута «O». Введение этого атрибута связано с тем, что ряд программ открывают файл на чтение и запись, хотя реально используют только операции чтения. В этом случае пользователю приходится разрешать Права доступа и на чтение, и на запись, что потенциально может служить источником информационных угроз. Чтобы избавиться от этой опасности, можно «разнести» данные по специальным каталогам. Это вполне возможный путь, однако его применение приводит к усложнению «Плана защиты» и увеличению количества каталогов. Введенный в СЗИ «Аккорд» атрибут «O» позволяет решить задачу другим методом, а именно: атрибут «O» подменяет (для задачи) атрибут «R» на совокупность атрибутов «R» и «W». При этом операция открытия файла проходит нормально, а попытка записи в этот файл классифицируется как НСД. Решение о применении для описания ПРД пользователей атрибута «O» принимается администратором БИ в том случае, когда файл по плану защиты должен быть доступен пользователю только для чтения, а применяемая для обработки данных программа пытается открыть этот файл и на чтение, и на запись.

Анализ ситуации может быть выполнен путем изучения журнала при тестировании программного обеспечения, планируемого для включения в состав программных средств АС. Некоторые редакторы текстов (в частности, NE и «Лексикон») сохраняют при редактировании информацию в файлах, имя которого отличается от имени редактируемого файла (указанные редакторы заменяют первый символ имени на символ «тильда» — волнистая линия). Если нет возможности не использовать такие редакторы, то необходимо по - крайней мере проследить, чтобы эти файлы оставались недоступными для пользователей, не имеющих на это полномочий. Обратите внимание на доставку информации в виде исполняемого файла — не важно, в каком виде осуществляется доставка — по сети, с помощью отчуждаемого носителя и др. Очень часто именно такой способ используется для внедрения программных закладок и очень часто покушения такого рода оказываются успешными.

Вот еще один вариант воздействия, выведший из строя не одну BBS — посылка архивированного файла, содержащего несколько Gb нулей. В архивированном виде такой файл занимает весьма немного места, а при раскрытии очень быстро занимает все пространство диска. Для того, чтобы избежать неприятностей такого рода, нужно запрещать запуск задач из всех каталогов, кроме тех, в которых хранятся проверенные модули. Следует также обратить внимание на использование отчуждаемых носителей — как в плане разрешения на использование (далеко не каждому пользователю это необходимо), так и в плане регистрации и учета.

5.3. Примеры ПРД для типовых ситуаций разграничения доступа

Будем считать, что физический диск на компьютере разбит на два логических диска. Как на диске C:, так и на D: размещены каталоги, доступ к которым могут иметь разные пользователи. Программные средства в основном размещены на C:.

Пример 1. Субъекту разрешено работать в каталоге C:\DOC.

Дополнительных ограничений нет. В этом случае ПРД для пользователя должны содержать следующий перечень атрибутов:

Права доступа
Диски
C: [RWCDNVMEGXA]
Каталоги
C:\[RWCDNVMEGX 0]
C:\MSDOS\[RWCDNVMEGX S]
C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
C:\DOC\[RWCDNVMEG S]
Файлы <Нет>

Описания ряда каталогов требуют пояснения. Так, каталог C:\NORTON\ должен быть описан, так как из него запускаются задачи — по - крайней мере, NC.EXE. Каталог ...\TEMP\ следует всегда описывать из общих соображений — часто он требуется для размещения временных файлов прикладных задач. Естественно, должен быть описан и корневой каталог, но наследование прав доступа отключается. Обратите внимание — при таких атрибутах видны файлы, размещенные в корневом каталоге диска C:. Для того, чтобы эти файлы были невидимы пользователю, из описания корневого каталога нужно удалить атрибут «V».

Права доступа
Диски
C: [RWCDNVMEGXA]
Каталоги
C:\ [RWCDN MEGX 0]
C:\MSDOS\[RWCDNVMEGX S]
C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
C:\DOC\[RWCDNVMEGX S]
Файлы <Нет>

Вот теперь мы получили то, что хотели.

Пример 2. Разрешено работать только с файлами и только в выделенном каталоге.

В этом случае пользователю необходимо запретить запуск задач, создание и удаление подкаталогов.

Права доступа
Диски
C: [RWCDNVMEGXA]
Каталоги
C:\ [RWCDN MEGX 0]
C:\MSDOS\ [RWCDNVMEGX S]
C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
C:\DOC\[RWCDNV G]
Файлы <Нет>

Теперь можно вернуться к Примеру 1, и убедиться, что создать каталог можно, но увидеть их, перейти к ним и вообще работать с ними будет трудновато — по крайней мере до тех пор, пока администратор БИ не установит вновь созданным каталогам необходимые атрибуты.

Пример 3. Применение атрибутов наследования

Есть и более простой способ — установить для разрешенного каталога атрибут наследования прав доступа.

Права доступа
Диски
C: [RWCDNVMEGXA]
Каталоги
C:\ [RWCDN MEGX 0]
C:\MSDOS\[RWCDNVMEGX S]
C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
C:\DOC\[RWCDNVMEG S]
Файлы <Нет>

Теперь с подкаталогами проблем быть не должно.

Пример 4. То же, но пользователю нельзя удалять файлы.

Права доступа
Диски
C: [RWCDNVMEGXA]
Каталоги
C:\ [RWCDN MEGX 0]
C:\MSDOS\[RWCDNVMEGX S]
C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
C:\DOC\[RWC NVMEG S]
Файлы <Нет>

Аналогичным образом можно проверить действие других атрибутов и сочетаний атрибутов. Этот эксперимент наверняка наведет на мысли об оптимальном применении разграничения доступа в Вашей ситуации.

Пример 5. Пользователю предоставляется право с полными полномочиями работать в выделенной директории на диске D:.

Основные параметры Идентификатор: G2
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска: C:\NORTON\NC.EXE
Детальность журнала: Низкая
Доступ к печати: Полный
Права доступа
Диски
C: [RWCDNVMEGXA]
D: [RWCDNVMEG A]
Каталоги
C:\ [RWCDN MEGX 0]
C:\MSDOS\ [RWCDNVMEGX S]
C:\MSDOS\TEMP\ [RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
D:\ [RWCDNVMEG 0]
D:\HHH\ [RWCDNVMEGX S]
Файлы <Нет>

Отметим, что в этом случае будут видны файлы, расположенные непосредственно в корне D:.

Точнее было бы сделать так:

Права доступа
Диски
C: [RWCDNVMEGXA]
D: [RWCDNVMEG A]
Каталоги
C:\ [RWCDN MEGX 0]
C:\MSDOS\ [RWCDNVMEGX S]
C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
D:\ [RWCDN MEG 0]
D:\HHH\ [RWCDNVMEGX S]
Файлы <Нет>

Пример 6. Конфиденциальное делопроизводство.

Обычно задача конфиденциального делопроизводства ставится так. Пользователю поручено исполнить документ, взяв исходные материалы из указанной ему директории, и передать готовый документ в другую указанную ему директорию. Это означает, что мы должны определить для пользователя различные права доступа к директориям. Так, из одной директории пользователь может только читать файлы (и, естественно, ознакомиться с ними), но не может ни редактировать их, ни удалять. В другой («своей») директории он может обрабатывать документы без ограничений, а в третью директорию может только записывать готовые материалы. Пусть для исполнения документа выделен каталог ...\HHH\, а пользователь с разными правами может использовать подкаталоги A1,A2 и A3. Эта задача реализуется следующими атрибутами.

Основные параметры
Идентификатор: G2
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска: C:\NORTON\NC.EXE
Детальность журнала: Низкая
Доступ к печати: Полный

Права доступа
Диски
C: [RWCDNVMEGXA]
D: [RWCDNVMEG A]
Каталоги
C:\ [RWCDN MEGX 0]
C:\MSDOS\[RWCDNVMEGX S]
C:\MSDOS\TEMP\ [RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
D:\ [RWCDN MEG 0]
D:\HHH\ [G 0]
D:\HHH\A1\[RV G 0]
D:\HHH\A2\[RWCDNVMEG 0]
D:\HHH\A3\[WC V G 0]
Файлы <Нет>

Пример 7. То же, но пользователь в своей работе использует редактор WORD.

Пусть редактор размещен на C:\WORD\, и путь к нему прописан в обычном порядке. Задачу можно решить с помощью следующего набора атрибутов.

Основные параметры
Идентификатор: G2
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска: C:\NORTON\NC.EXE
Детальность журнала: Низкая
Доступ к печати: Полный

Права доступа
Диски
C: [RWCDNVMEGXA]
D:[RWCDNVMEG A]

Каталоги
D:\ [RWCDN MEG 0]
D:\HHH\ [G 0]
D:\HHH\A1\[RV G 0]
D:\HHH\A2\[RWCDNVMEG 0]
D:\HHH\A3\[WC V G 0]
C:\ [RWCDN MEGX 0]
C:\MSDOS\ [RWCDNVMEGX S]
C:\MSDOS\TEMP\ [RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
C:\WORD\ [RWC V X S]
Файлы <Нет>

Атрибуты для C:\WORD\ можно прокомментировать так. R — файлы из подкаталогов читаются самим редактором. W — запись нужна для организации временных файлов. C — создание файлов — так же для временных файлов. V — WORD использует процедуры FindFirst и FindNext при работе с файлами. X — редактор должен быть запущен на исполнение. S — наследование необходимо, так как обычно каталог ...\WORD\ содержит подкаталоги, над содержимым которых выполняются те же операции. Из этого примера ясно, что перед включением некоторого программного средства в перечень используемых в системе разграничения доступа, нужно его особенности изучить. Изучение может основываться на использовании высокого уровня детальности журнала и анализе результатов. После того, как станет ясно, какие ресурсы требует то или иное программное средство, необходимо на основе анализа выявить, имеются ли возможности для нарушения ПРД и можно ли создать такие ПРД, которые обеспечат требуемый уровень безопасности информации. Только вслед за этим администратор БИ может принять решение о возможности применения тех или иных средств. Некоторые рекомендации по формированию ПРД при использовании наиболее распространенных программных средств обработки информации приведены в соответствующем разделе ниже.

Пример 8. То же, но пользователь имеет право читать файлы, размещенные в корневом каталоге на A:.

В этом случае атрибуты могут быть такими:
Права доступа
Диски
A: [RVA]
C: [RWCDNVMEGXA]
D:[RWCDNVMEG A]
Каталоги
D:\ [RWCDN MEG 0]
D:\HHH\ [G 0]
D:\HHH\A1\[RV G 0]
D:\HHH\A2\[RWCDNVMEG 0]
D:\HHH\A3\[WC V G 0]

A:\[R V 0]
C:\ [RWCDN MEGX 0]
C:\MSDOS\ [RWCDNVMEGX S]
C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
C:\WORD\ [RWC V X S]
Файлы <Нет>

Пример 9. То же, но пользователь может читать все файлы, размещенные на A:.

Это означает, что пользователю должны быть доступны все подкаталоги, т.е. нужно установить атрибут «G» (при наличии атрибута наследования).

Права доступа
Диски
A: [RV G A]
C: [RWCDNVMEGXA]
D:[RWCDNVMEG A]
Каталоги
D:\ [RWCDN MEG 0]
D:\HHH\ [G 0]
D:\HHH\A1\[RV G 0]
D:\HHH\A2\[RWCDNVMEG 0]
D:\HHH\A3\[WC V G 0]
A:\[RV G S]
C:\[RWCDN MEGX 0]
C:\MSDOS\TEMP\ [RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
C:\WORD\ [RWC V X S]
Файлы <Нет>

Пример 10. Доступ к печати.

Как уже отмечалось, администратор БИ устанавливает каждому пользователю параметры доступа к печати. Печать может быть не ограничена, и в этом случае основные параметры могут быть такими:

Основные параметры
Идентификатор: G2
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска: C:\NORTON\NC.EXE
Детальность журнала: Низкая
Доступ к печати: Полный

Пример 11. Печать недоступна
Основные параметры
Идентификатор: G2

Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска: C:\NORTON\NC.EXE
Детальность журнала: Низкая
Доступ к печати: Нет

Пример 12. Установка атрибутов выделенных файлов.

Как уже отмечалось, дополнительно могут определяться права доступа к отдельным файлам — с приоритетом, даже если файл расположен в каталоге, доступа к которому данный пользователь не имеет. Рассмотрим задачу, аналогичную приведенной в Примере 9, но с тем отличием, что исходный материал для исполнения документа расположен в файле C:\BOOK\BOOK.DOC. Этот материал должен быть доступен пользователю, но другие файлы из того же каталога должны быть недоступны. В этом случае атрибуты могут быть такими:

Права доступа
Диски
A: [RV G A]
C: [RWCDNVMEGXA]
D:[RWCDNVMEG A]
Каталоги
D:\ [RWCDN MEG 0]
D:\HHH\ [G 0]
D:\HHH\A1\[RV G 0]
D:\HHH\A2\[RWCDNVMEG 0]
D:\HHH\A3\[WC V G 0]
A:\[RV G S]
C:\[RWCDN MEGX 0]
C:\MSDOS\ [RWCDNVMEGX S]
C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
C:\WORD\ [RWC V X S]
Файлы
C:\BOOK\BOOK.DOC [RWC V]

В результате работы монитора разграничения доступа с такими атрибутами пользователь не сможет увидеть исходный каталог и файл, но вполне может скопировать его командой copy c:\book\book.doc d:\hhh\a2\c1\book.doc. Обратите внимание на наличие атрибутов W,C и V. Функции, соответствующие этим атрибутам, используются встроенной командой DOS «COPY», и в этой связи, установка их является обязательной. Чтобы избавится от таких неоднозначностей, вместо команды «COPY» лучше использовать специально подготовленную программу. Если Вы используете не command.com, а другой интерпретатор командной строки, то перечень атрибутов может быть другим. Так,

для NDOS достаточно атрибутов R и V — он написан корректней. При использовании ndos.com состав атрибутов может быть таким:

Права доступа
Диски
A: [RV G A]
C: [RWCDNVMEGXA]
D:[RWCDNVMEG A]
Каталоги
A:\ [RV G S]
C:\ [RWCDN MEGX 0]
C:\MSDOS\ [RWCDNVMEGX S]
C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
C:\WORD\ [RWCD VS]
D:\ [RWCDN MEG 0]
D:\HHH\ [G 0]
D:\HHH\A1\[RV G 0]
D:\HHH\A2\[RWCDNVMEGX 0]
D:\HHH\A3\[WC V G 0]
Файлы
C:\BOOK\BOOK.DOC [RV]

Такие же атрибуты можно использовать и при применении 4DOS.COM. Естественно, на аналогичном принципе можно построить и запись в заранее определенный файл.

Пример 13. Установка атрибутов для выделенных программ.

Предусмотрена также возможность установки разрешения на исполнение задач, размещенных в выделенном файле — выделенных задач. Пусть необходимо запустить утилиту MEMSCAN.EXE. Для этого установим следующие атрибуты:

Права доступа
Диски
C: [RWCDNVMEGXA]
D:[RWCDNVMEGXA]
Каталоги
D:\ [RWCDN MEG 0]
D:\HHH\ [G 0]
D:\HHH\A1\[RV G 0]
D:\HHH\A2\[RWCDNVMEGX 0]
D:\HHH\A3\[WC V G 0]
C:\ [RWCDN MEGX 0]
C:\MSDOS\[RWCDNVMEGX S]

C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
Файлы
C:\BOOK\BOOK.DOC [RWC V]
C:\ACCORD\MEMSCAN.EXE [V X]

Обратите внимание — кроме атрибута «X» необходимо установить и «V». Это связано с особенностями запуска задач интерпретатором командной строки «COMMAND.COM» (как, впрочем, и NDOS.COM и 4DOS.COM).

Пример 14. Еще один пример установки атрибутов для выделенных задач — на этот раз это будет WORD.

Права доступа
Диски
A: [RV G A]
C: [RWCDNVMEGXA]
D:[RWCDNVMEGXA]
Каталоги
D:\ [RWCDN MEG 0]
D:\HHH\ [G 0]
D:\HHH\A1\[RV G 0]
D:\HHH\A2\[RWCDNVMEGX 0]
D:\HHH\A3\[WC V G 0]
A:\ [RV G S]
C:\ [RWCDN MEGX 0]
C:\MSDOS\ [RWCDNVMEGX S]
C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
C:\WORD\ [RWCD VS]
Файлы
C:\BOOK\BOOK.DOC [RWC V]
C:\WORD\WORD.EXE [RW V X]
C:\ACCORD\MEMSCAN.EXE [V X]

Сравнивая последние две строки, можно заметить, что для запуска WORD необходимо установить атрибуты «R» и «W». Так уж WORD устроен. Узнать, какие ресурсы требует WORD, как, впрочем, и «COPY», нам удалось, только анализируя с помощью журнала возникающие проблемы. Именно так и придется поступать администратору БИ, если возникнет необходимость включения в защищенную АС новых (ранее не анализировавшихся) процедур.

Пример 15. Использование «черного» списка.

Пусть пользователю A4 запрещен доступ к файлам с расширением .BAT и .SYS, размещенным в корневом каталоге диска C:. ПРД в этом случае могут выглядеть так:

Пользователь: A4
Основные параметры
Идентификатор: A4
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска:
Детальность журнала: Низкая
Доступ к печати: LPT1: Без ограничений
LPT2: Без ограничений
LPT3: Без ограничений
Права доступа
Диски
C: [RWCDNVMEGXO A]
D:[RWCDNVMEGXO A]
Каталоги
C:\ [RWCDNVMEGX 0]
C:\MSDOS\ [RWCDNVMEGX S]
C:\MSDOS\TEMP\[RWCDNVMEG]
C:\NORTON\[RWCDNVME X]
Файлы <Нет>
Скрытые файлы
C:*.BAT [ ]
C:*.SYS [ ]

Пример 16. Анализ ресурсов.

Перед тем, как включить в АС новое программное средство, администратор БИ должен изучить его особенности в части доступа к ресурсам. Не исключено, что требуемые ресурсы не позволят применять изучаемые программы в составе АС, или, возможно, возникнет необходимость пересмотреть «План защиты». Для анализа ресурсов целесообразно установить в программе «ACED» для некоторого пользователя высокий уровень детальности журнала и полный доступ к каталогам и файлам, провести сеанс работы с изучаемым программным средством, а затем посредством программы «ACLOG» изучить требуемые для работы программы ресурсы. Необходимо помнить, что объем журналов при высоком уровне детальности будет очень большим, и, в этой связи, сеанс работы должен быть не слишком длинным. Лучше, в случае необходимости, изучение провести несколькими небольшими сеансами.

Пример 17. Использование атрибута «O».

Пусть в состав АС включена некоторая специализированная процедура копирования информации mycopy.exe (естественно, это только пример). Пусть, также, по Плану защиты пользователю доступны каталоги, как в Примере 6. Попытаемся воспользоваться этой процедурой при следующих установленных атрибутах:

```
Права доступа
Диски
C: [RWCDNVMEGXA]
D: [RWCDNVMEG A]
Каталоги
C:\ [RWCDN MEGX 0]
C:\MSDOS\ [RWCDNVMEG X ]
C:\MSDOS\TEMP\[RWCDNVMEG ]
C:\NORTON\[RWCDNVME X ]
D:\ [RWCDN MEG 0]
D:\HHH\ [G 0]
D:\HHH\A1\[RV G 0]
D:\HHH\A2\[RWCDNVMEG 0]
D:\HHH\A3\[ WC V G 0]
Файлы <Нет>
```

Воспользуемся для копирования имеющейся в АС программой: mycopy d:\hhh\a1\test.txt d:\hhh\a2\test.tst. При этом будет выдано сообщение об ошибке — о невозможности открыть файл. Изменим теперь атрибуты доступа, а именно в части описания прав доступа к каталогу d:\hhh\a1: D:\HHH\A1\ [RV G O 0]. При этом процедура будет успешно выполнена. Для анализа приведем текст программы mycopy — как образец неправильно написанной программы — вот такие программы не надо использовать — но, увы, иногда приходится.

```
program mycopy; {$I-}
var F1,F2 : File;
IO : word;
C : byte;
begin
 if ParamCount <> 2 then begin Writeln(" Должно быть два параметра на
входе "); Halt; end;
Assign(F1,ParamStr(1));
Assign(F2,ParamStr(2));
Reset(F1,1);
IO := IOResult;
 if IO <> 0 then begin Writeln(" Входной файл не найден "); Halt; end;
Rewrite(F2,1);
IO := IOResult;
if IO <> 0 then begin Writeln(" Ошибка создания выходного файла "); Halt;
end;
while Not Eof(F1) do begin
BlockRead(F1,C,1);
IO := IOResult;
if IO <> 0 then begin Writeln(" Ошибка чтения входного файла "); Halt;
end;
BlockWrite(F2,C,1);
```



```
IO := IOResult;
 if IO <> 0 then begin Writeln(" Ошибка записи выходного файла "); Halt;
end;
end;
Close(F2);
IO := IOResult;
if IO <> 0 then begin Writeln(" Ошибка закрытия выходного файла ");
Halt; end;
Close(F1);
end.
```

Ошибка, допущенная в этой программе, состоит в том, что при открытии файла не был установлен режим открытия FileMode. Если FileMode = 0, то файл будет открыт Read Only, FileMode = 1, то файл будет открыт Write Only, FileMode = 2, то файл будет открыт Read/Write. По умолчанию FileMode = 2, то есть осуществляется попытка открыть файл и на чтение, и на запись, что и приводит к конфликтной ситуации. Естественно, если есть возможность использовать правильно подготовленные программы, то их и нужно использовать. Если же выбора нет — необходимо использовать атрибут «O».

Пример 18. Описание сетевого ресурса.

Иллюстрацию использования сетевых ресурсов продемонстрируем на задаче, аналогичной приведенной в Примере 6. Здесь, однако, каталоги размещаются на сервере, а пользователи имеют различные права. Приводимая цепочка показывает, как можно организовать конфиденциальное делопроизводство в сети. Необходимо обратить внимание на описание каталогов, расположенных на сервере.

```
Пользователь: A1
Основные параметры
Идентификатор: A1
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска:
Детальность журнала: Низкая
Доступ к печати: LPT1: Без ограничений LPT2: Без ограничений LPT3:
Без ограничений
Права доступа
Диски
C: [RWCDNVMEGXO A ]
D: [RWCDNVMEGXO A ]
S: [RWCDNVMEGXO A ]
Каталоги
C:\ [RWCDNVMEGXO S]
D:\ [RWCDNVMEGXO 0]
\SERVER\VOL2\[G 0]
\SERVER\VOL2\A1\[RV GX A 0]
```

\SERVER\VOL2\A2\[RWCDNVMEGXO 0]
\SERVER\VOL2\A3\[WC V G O 0]
Файлы <Нет>
Скрытые файлы <Нет>
Пользователь: A2
Основные параметры
Идентификатор: A2
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска:
Детальность журнала: Низкая
Доступ к печати: LPT1: Без ограничений LPT2: Без ограничений LPT3: Без ограничений
Права доступа
Диски
C: [RWCDNVMEGXO A]
D: [RWCDNVMEGXO A]
S: [RWCDNVMEGXO A]
Каталоги
C:\ [RWCDNVMEGXO S]
D:\ [RWCDNVMEGXO S]
\SERVER\VOL2\[G 0]
\SERVER\VOL2\A2\[RV GX A 0]
\SERVER\VOL2\A3\[RWCDNVMEGXO 0]
\SERVER\VOL2\A4\[WC V G O 0]
Файлы <Нет>
Скрытые файлы <Нет>
Пользователь: A3
Основные параметры
Идентификатор: A3
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска: C:\VC\VC.COM
Детальность журнала: Низкая
Доступ к печати: LPT1: Без ограничений LPT2: Без ограничений LPT3: Без ограничений
Права доступа
Диски
C: [RWCDNVMEGXO A]
D: [RWCDNVMEGXO A]
S: [RWCDNVMEGXO A]
Каталоги
C:\ [RWCDNVMEGXO S]
D:\ [RWCDNVMEGXO S]
\SERVER\VOL2\[G 0]

\SERVER\VOL2\A3\[RV GX A 0]
\SERVER\VOL2\A4\[RWCDNVMEGXO 0]
\SERVER\VOL2\A5\[WC V G O 0]
Файлы <Нет>
Скрытые файлы <Нет>

6. РЕКОМЕНДАЦИИ ПО СОЗДАНИЮ ИЗОЛИРОВАННОЙ ПРОГРАММНОЙ СРЕДЫ (ИПС) В ОС WINDOWS 95/98

Windows 95/98 обладает достаточно обширным набором функций и утилит для изменения конфигурации и подключения новых устройств и ресурсов. С одной стороны эти функции облегчают работу квалифицированному пользователю, но с другой — могут служить источником НСД. В каталоге \Windows\System находятся файлы с расширением .cpl. Это файлы, которые позволяют изменять конфигурацию и настройку системы и отдельных ее компонент. «Иконки» этих файлов Вы можете увидеть на Панели Управления. Ниже приводится список файлов .cpl для типичной конфигурации Windows 95/98 и функции, выполняемые каждым файлом.

APPWIZ.CPL — Applications Setup Wizard (Установка и удаление программ)
DESK.CPL — Desktop Control Panel (Экран)
INTL.CPL — International Control Panel (Языки и стандарты)
JOY.CPL — Joystic Control Panel (Джойстик)
MAIN.CPL — Main Control Panel (Клавиатура, Мышь, Принтеры)
MLCFG32.CPL — Microsoft Exchange (Почта и факс)
MMSYS.CPL — Multimedia Control Panel (Мультимедиа)
MODEM.CPL — Microsoft Modem Control Panel (Модемы)
NETCPL.CPL — Network Control Panel (Сеть)
ODBCCP32.CPL — ODBC Control Panel (32bit ODBC)
PASSWORD.CPL — Network Security and Administration UI (Пароли)
SYSDM.CPL — System Setting Device Manager Control Panel (Система и Установка оборудования)
TELEPHON.CPL — Telephon Control Panel (Телефон)
TIMEDATE.CPL — Time/date Setting Control Panel (Дата/время).

Любой из этих файлов, внесенный в список «скрытых» файлов в редакторе прав доступа ACED.EXE, становится недоступным (для конкретного пользователя) и соответствующая функция исключается из Панели Управления.

Для защиты от НСД также большое значение имеет файл MSSHRUI.DLL в каталоге \WINDOWS\SYSTEM. Данный файл управляет выделением ресурсов компьютера в совместное пользование.

Рассмотрим более подробно методику создания ИПС при использовании Windows 95/98.

6.1. «Аккорд» установлен на локальном компьютере

В этом случае в редакторе прав доступа ACED.EXE для конкретного пользователя следует внести в список «скрытых» файлов APPWIZ.CPL, MLCFG32.CPL, MODEM.CPL, NETCPL.CPL, PASSWORD.CPL и SYSDM.CPL.

Пользователю запрещено:

— установка и удаление программ;

— работа с Microsoft Network и Microsoft Exchange;

— установка и настройка модемов;

— установка и настройка сетевых карт;

— изменение способа входа в Windows и пароля, а также удаленное управление данным компьютером;

— установка/удаление оборудования.

Стартовой задачей следует указать C:\WINDOWS\win.com. Доступ к каталогам и файлам прописать в соответствии с полномочиями, установленными для данного пользователя. После перезагрузки компьютера и входа в систему зарегистрированного пользователя запустится Windows 95, в которой пользователь может работать только в разрешенных каталогах и только с установленным ПО. ПРД в этом случае могут выглядеть так:

```
Пользователь: MAIN_USER
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска: C:\WINDOWS\WIN.COM
Детальность журнала:  Низкая
 Диски
A: [RWCDNVMEGXO  A  ]
C: [RWCDNVMEGXO  A  ]
D: [RWCDNVMEGXO  A  ]
Каталоги
A:\[RWCDNVMEGXO    S]
C:\[RW  V GXO    0]
D:\ [RW  V GO    0]
C:\ACCORD\  [RW  V GXO    S]
C:\DRWEB_~1\[RW  V GXO    S]
C:\MSOFFICE\[RWCDNVMEGX O S]
C:\PROGRA~1\[RWCDNVMEGX O S]
C:\TEMP\[RWCDNV  G O    S]
C:\WINDOWS\  [RWCDNVMEGXO    S]
D:\OPEN_DOC\[RWCDNVMEG O    S]
```



```
Файлы
Скрытые файлы
C:\WINDOWS\REGEDIT.EXE                    [        ]
C:\WINDOWS\SYSTEM\APPWIZ.CPL              [        ]
C:\WINDOWS\SYSTEM\MLCFG32.CPL             [        ]
C:\WINDOWS\SYSTEM\MODEM.CPL               [        ]
C:\WINDOWS\SYSTEM\NETCPL.CPL              [        ]
C:\WINDOWS\SYSTEM\PASSWORD.CPL            [        ]
C:\WINDOWS\SYSTEM\SYSDM.CPL               [        ]
```

Пользователь сможет работать с документами в каталоге D:\OPEN_DOC средствами Windows или MSOffice, но не может изменить конфигурацию системы. Обратите внимание, что при описании ПРД длинные имена файлов и каталогов прописываются в «коротком» виде, как они отображаются в MS DOS.

6.2. «Аккорд» установлен на компьютере, подключенном к ЛВС

В этом случае в редакторе прав доступа ACED.EXE для конкретного пользователя следует внести в список «скрытых» файлов APPWIZ.CPL, MLCFG32.CPL, MODEM.CPL, NETCPL.CPL, PASSWORD.CPL, SYSDM.CPL и MSSHRUI.DLL.

Пользователю запрещено:

— установка и удаление программ;

— работа с Microsoft Network и Microsoft Exchange;

— установка и настройка модемов;

— установка и настройка сетевых карт;

— изменение способа входа в Windows и пароля, а также удаленное управление данным компьютером;

— установка/удаление оборудования;

— предоставления дисков, файлов и принтеров компьютера в совместное пользование (MSSHRUI.DLL).

ПРД в этом случае могут выглядеть так:

```
Пользователь: MAIN_USER
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска: C:\WINDOWS\WIN.COM
Детальность журнала:  Низкая
 Диски
A: [RWCDNVMEGXO  A  ]
C: [RWCDNVMEGXO  A  ]
D: [RWCDNVMEGXO  A  ]
```

```
Каталоги
A:\[RWCDNVMEGXO    S]
C:\[RW  V GXO    0]
D:\[RW  V GO    0]
C:\ACCORD\[RW  V GXO    S]
C:\DRWEB_~1\[RW  V GXO    S]
C:\MSOFFICE\[RWCDNVMEGX O S]
C:\PROGRA~1\ [RWCDNVMEGX O S]
C:\TEMP\[RWCDNV G O    S]
C:\WINDOWS\[RWCDNVMEGXO    S]
D:\OPEN_DOC[RWCDNVMEG O    S]
Файлы
Скрытые файлы
C:\WINDOWS\REGEDIT.EXE                    [        ]
C:\WINDOWS\SYSTEM\ MSSHRUI.DLL            [        ]
C:\WINDOWS\SYSTEM\APPWIZ.CPL              [        ]
C:\WINDOWS\SYSTEM\INETCPL.CPL             [        ]
C:\WINDOWS\SYSTEM\MAIN.CPL                [        ]
C:\WINDOWS\SYSTEM\MLCFG32.CPL             [        ]
C:\WINDOWS\SYSTEM\MMSYS.CPL               [        ]
C:\WINDOWS\SYSTEM\MODEM.CPL               [        ]
C:\WINDOWS\SYSTEM\NETCPL.CPL              [        ]
C:\WINDOWS\SYSTEM\PASSWORD.CPL            [        ]
C:\WINDOWS\SYSTEM\SYSDM.CPL               [        ]
C:\WINDOWS\SYSTEM\TELEPHON.CPL            [        ]
```

6.3. Конфиденциальное делопроизводство в среде Windows 95/98 и Microsoft Office

Реализация технологии конфиденциального делопроизводства в среде Windows осложняется тем, что операционная система и программы MSOffice в процессе работы создают, используют, удаляют и переименовывают множество временных служебных файлов. При этом основная задача администратора БИ — разрешить пользователю работать с конфиденциальными документами только в выделенных каталогах и исключить возможность сохранения документов в любых иных областях дискового пространства.

ПРД в этом случае могут выглядеть так:

```
Пользователь: MAIN_USER
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска: C:\WINDOWS\WIN.COM
Детальность журнала:  Низкая
```



```
Диски
A: [RWCDNVMEGXO  A  ]
C: [RWCDNVMEGXO  A  ]
D: [RWCDNVMEGXO  A  ]
Каталоги
A:\[RWCDNVMEGXO    S]
C:\[RW  V GXO    0]
D:\[RW  V G  O    0]
C:\ACCORD\[RW  V GXO    S]
C:\DRWEB_~1\[RW  V GXO    S]
C:\MSOFFICE\[RW  V GXO    S]
C:\PROGRA~1\[RW  V GXO    S]
C:\TEMP\[RWCDNV G O    S]
C:\WINDOWS\[RW  V GXO    S]
D:\SECRET_1\[RWCDNVMEG O    S]
D:\SECRET_2\[RWCDNVMEG O    S]
D:\SECRET_3\[RWCDNVMEG O    S]
Файлы
C:\MSOFFICE\OFFICE\ПАНЕЛЬ~1\OFFICE.TBB        [RWCDNV  O]
C:\MSOFFICE\ШАБЛОНЫ\NORMAL.DOT                [RWCDNV O]
C:\MSOFFICE\ШАБЛОНЫ\~$NORMAL.DOT              [RWCDNV O]
C:\PROGRAM\COMMON F\MICROSOF\PROOF\MSSP2_EN.EXC
                                              [RWCDNV XO]
C:\PROGRAM\COMMON F\MICROSOF\PROOF\MSSP_RU.EXC
                                              [RWCDNV XO]
C:\SCANDISK.LOG                               [RWCDNV  O]
C:\WINDOWS\*.INI                                  [R  V  O]
C:\WINDOWS\HELP\*.TMP                         [RWCDNV O]
C:\WINDOWS\IOS.LOG                            [RWCDNV O]
C:\WINDOWS\MSAPPS\PROOF\CUSTOM.BAK            [RWCDNV O]
C:\WINDOWS\MSAPPS\PROOF\CUSTOM.DIC            [RWCDNV O]
C:\WINDOWS\MSAPPS\PROOF\~$CUSTOM.DIC          [RWCDNV O]
C:\WINDOWS\RECENT\*.LNK                       [RWCDNV O]
C:\WINDOWS\SHELLI~1                           [RWCDNV XO]
C:\WINDOWS\SPOOL\PRINTERS\*.SHD               [RWCDNV O]
C:\WINDOWS\SPOOL\PRINTERS\*.SPL               [RWCDNV O]
C:\WINDOWS\SYSTEM.DA0                         [RWCDNV O]
C:\WINDOWS\SYSTEM.DAT                         [RWCDNV O]
C:\WINDOWS\TEMP\*.TMP                         [RWCDNV O]
C:\WINDOWS\TEMP\~$CUSTOM.DIC                  [RWCDNV O]
C:\WINDOWS\USER.DA0                           [RWCDNV O]
C:\WINDOWS\USER.DAT                           [RWCDNV O]
C:\WINDOWS\VDDASD.DAT                         [RWCDNV O]
C:\WINDOWS\WNBOOTNG.STS                       [RWCDNV O]
```

Скрытые файлы
C:\WINDOWS\REGEDIT.EXE [ ]
C:\WINDOWS\SYSTEM\APPWIZ.CPL [ ]
C:\WINDOWS\SYSTEM\INETCPL.CPL [ ]
C:\WINDOWS\SYSTEM\JOY.CPL [ ]
C:\WINDOWS\SYSTEM\MAIN.CPL [ ]
C:\WINDOWS\SYSTEM\MLCFG32.CPL [ ]
C:\WINDOWS\SYSTEM\MMSYS.CPL [ ]
C:\WINDOWS\SYSTEM\MODEM.CPL [ ]
C:\WINDOWS\SYSTEM\NETCPL.CPL [ ]
C:\WINDOWS\SYSTEM\PASSWORD.CPL [ ]
C:\WINDOWS\SYSTEM\SYSDM.CPL [ ]
C:\WINDOWS\SYSTEM\TELEPHON.CPL [ ]

Пользователь MAIN_USER имеет право работать со всеми тремя каталогами на диске D:\ и диском A:\. Файлы операционной системы и прикладного ПО находятся на диске C:\. При этом запрещено создание, удаление и переименование любых файлов и каталогов диска C:. Данный пользователь по должностной инструкции может выполнять функции начальника канцелярии. Для обычного пользователя, который работает с документами только в выделенном ему каталоге на диске D:\ и не имеет доступа к сменным дискетам на диске A:\, ПРД выглядят следующим образом:

Пользователь: USER_1
Права администратора: Нет
Стартовый каталог: C:\
Задача для запуска: C:\WINDOWS\WIN.COM
Детальность журнала: Низкая
Диски
A: [RWCDNVMEGXO]
C: [RWCDNVMEGXO A]
D: [RWCDNVMEGXO A]
Каталоги
A:\[RWCDNVMEGXO 0]
C:\ [RW V GXO 0]
D:\ [RW V G O 0]
C:\ACCORD\[RW V GXO S]
C:\DRWEB_~1\[RW V GXO S]
C:\MSOFFICE\[RW V GXO S]
C:\PROGRA~1\[RW V GXO S]
C:\TEMP\[RWCDNV G O S]
C:\WINDOWS\[RW V GXO S]
D:\SECRET_1\[RWCDNVMEG O S]

Файлы
C:\MSOFFICE\OFFICE\ПАНЕЛЬ~1\OFFICE.TBB [RWCDNV O]
C:\MSOFFICE\ШАБЛОНЫ\NORMAL.DOT [RWCDNV O]
C:\MSOFFICE\ШАБЛОНЫ\~$NORMAL.DOT [RWCDNV O]
C:\PROGRAM\COMMON F\MICROSOF\PROOF\MSSP2_EN.EXC [RWCDNV XO]
C:\PROGRAM\COMMON F\MICROSOF\PROOF\MSSP_RU.EXC [RWCDNV XO]
C:\SCANDISK.LOG [RWCDNV O]
C:\WINDOWS*.INI [R V O]
C:\WINDOWS\HELP*.TMP [RWCDNV O]
C:\WINDOWS\IOS.LOG [RWCDNV O]
C:\WINDOWS\MSAPPS\PROOF\CUSTOM.BAK [RWCDNV O]
C:\WINDOWS\MSAPPS\PROOF\CUSTOM.DIC [RWCDNV O]
C:\WINDOWS\MSAPPS\PROOF\~$CUSTOM.DIC [RWCDNV O]
C:\WINDOWS\RECENT*.LNK [RWCDNV O]
C:\WINDOWS\SHELLI~1 [RWCDNV XO]
C:\WINDOWS\SPOOL\PRINTERS*.SHD [RWCDNV O]
C:\WINDOWS\SPOOL\PRINTERS*.SPL [RWCDNV O]
C:\WINDOWS\SYSTEM.DA0 [RWCDNV O]
C:\WINDOWS\SYSTEM.DAT [RWCDNV O]
C:\WINDOWS\TEMP*.TMP [RWCDNV O]
C:\WINDOWS\TEMP\~$CUSTOM.DIC [RWCDNV O]
C:\WINDOWS\USER.DA0 [RWCDNV O]
C:\WINDOWS\USER.DAT [RWCDNV O]
C:\WINDOWS\VDDASD.DAT [RWCDNV O]
C:\WINDOWS\WNBOOTNG.STS [RWCDNV O]
Скрытые файлы
C:\WINDOWS\REGEDIT.EXE [ ]
C:\WINDOWS\SYSTEM\APPWIZ.CPL [ ]
C:\WINDOWS\SYSTEM\INETCPL.CPL [ ]
C:\WINDOWS\SYSTEM\JOY.CPL [ ]
C:\WINDOWS\SYSTEM\MAIN.CPL [ ]
C:\WINDOWS\SYSTEM\MLCFG32.CPL [ ]
C:\WINDOWS\SYSTEM\MMSYS.CPL [ ]
C:\WINDOWS\SYSTEM\MODEM.CPL [ ]
C:\WINDOWS\SYSTEM\NETCPL.CPL [ ]
C:\WINDOWS\SYSTEM\PASSWORD.CPL [ ]
C:\WINDOWS\SYSTEM\SYSDM.CPL [ ]
C:\WINDOWS\SYSTEM\TELEPHON.CPL [ ]

СЗИ «Аккорд» обладает широким набором атрибутов доступа, которые позволяют администратору БИ реализовать любую непротиворечивую политику безопасности информации.

7. УПРАВЛЕНИЕ ЗАЩИТОЙ ИНФОРМАЦИИ В ЛВС

Для полноценной защиты локальной вычислительной сети необходимо применять комплексную технологию, дополняющую рассмотренные выше механизмы. Эта технология защиты обеспечивается:

— установкой СЗИ «Аккорд 4+»/ «Аккорд 4++» с ПО Аккорд v.1.35, v.1.95, v.2.03 на рабочих станциях;

— установкой подсистемы контроля целостности файл-сервера на каждом сервере;

— установкой подсистемы усиленной аутентификации.

— установкой подсистемы распределенного аудита и управления.

Контроль целостности файл-сервера осуществляется аналогично контролю целостности на рабочей станции. Отличие состоит в том, что дополнительно контролируются файлы на томах NetWare за счет расширения пошагового механизма контроля целостности.

Подсистема усиленной аутентификации предоставляет дополнительный механизм проверки подлинности рабочих станций. Процедура проверки подлинности выполняется не только в момент подключения станции, но и в ходе функционирования ЛВС с установленной администратором периодичностью. Подсистема предотвращает как подмену локальной станции или сервера, так и подключение в ЛВС нелегальных станций/серверов. Подсистема усиленной аутентификации построена на применении механизма проверки подлинности запросов/ответов на основе технологии кода аутентификации, при этом, за счет использования аппаратного генератора случайных чисел обеспечивается уникальность каждого запроса/ответа, что предотвращает сканирование и подмену кодов при прослушивании трафика сети.

Система усиленной аутентификации в ЛВС основана на применении математических методов, позволяющих однозначно опознать участников диалога. Стойкость применяемых методов в значительной степени определяется параметрами ключа, в качестве которого используется случайная двоичная последовательность. При программной генерации псевдослучайных последовательностей создается ключ, потенциально обладающий такими недостатками, как периодичность и предсказуемость. Использование аппаратного генератора случайных чисел позволяет получать двоичную последовательность, лишенную вышеуказанных недостатков. В этой связи для защиты информации в ЛВС рекомендуется применять те модификации СЗИ «Аккорд», которые снабжены аппаратным генератором случайных чисел. Особенно это важно для одноранговых сетей, так как в этом случае любая из рабочих станций может выступать в качестве сервера.

Подсистема распределенного аудита и управления предполагает наличие выделенного рабочего места сетевого администратора безопасности. Подсистема позволяет администратору БИ отслеживать все действия пользователей на рабочих станциях и запросы пользователей к любым ресурсам, в том числе и сетевым. При этом полный журнал регистрации событий ведется на рабочей станции, а на АРМ администратора БИ выводятся сообщения о всех попытках несанкционированного доступа в реальном масштабе времени. Для оперативного анализа администратор БИ имеет возможность просматривать экран любой локальной станции, послать сообщение на любую станцию, получить журнал регистрации событий с локальной станции, блокировать работу на контролируемой станции вплоть до ее перезагрузки. Кроме оперативного наблюдения администратор БИ может назначать правила разграничения доступа для любого пользователя на любой рабочей станции.

Подсистема распределенного аудита и управления функционирует только при установленной подсистеме усиленной аутентификации.

7.1. Подсистема усиленной аутентификации

Усиленная аутентификация осуществляется на базе программно-аппаратных комплексов средств защиты информации от НСД семейства «Аккорд» с версией ПО 1.35, 1.95 или 2.03.

В качестве аппаратной части комплекса на рабочих станциях может использоваться контроллер «Аккорд 4+» с версией BIOS 1.12 или 1.40, «Аккорд 4++» с версией BIOS 1.40, а на сервере только «Аккорд 4+»/«Аккорд 4++» с BIOS 1.40 (АМДЗ).

Подсистема усиленной аутентификации (УА) предназначена для контроля целостности состава технических средств ЛВС. При использовании данной технологии, состав технических средств ЛВС фиксируется в момент установки подсистемы и его целостность контролируется при каждой попытке подключения рабочей станции к сети и периодически через интервал времени, установленный администратором безопасности информации. Например, в момент входа пользователя происходит диалог (сеанс УА) между сервером и рабочей станцией. В результате этого диалога и сервер и рабочая станция проверяют подлинность друг друга.

Подсистема усиленной аутентификации построена на применении механизма контроля подлинности запросов/ответов рабочей станции на основе технологии кода аутентификации, при этом за счет использования аппаратного генератора случайных чисел обеспечивается уникальность каждого запроса/ответа, что предотвращает раскрытие кодов при прослушивании запросов/ответов.

В защищенной ЛВС допускается работа только зарегистрированных рабочих станций и серверов.

7.1.1. Принцип работы

Аутентификация или установление подлинности чрезвычайно важны при работе пользователей на электронно-вычислительных машинах, особенно в составе локальной вычислительной сети. Пользователю чрезвычайно важно установить подлинность того, что операционная система или аппаратные средства, которые он использует для обработки данных, соответствуют тому, что должно было быть. В процессе аутентификации происходит проверка: является ли проверяемое лицо или объект на самом деле тем, за кого себя выдаёт.

Допустим, некая станция хочет получить доступ к ресурсам файлового сервера. Тогда будет произведён некоторый диалог между рабочей станцией, желающей воспользоваться ресурсами файлового сервера и этим файловым сервером (Рис. 3.47).

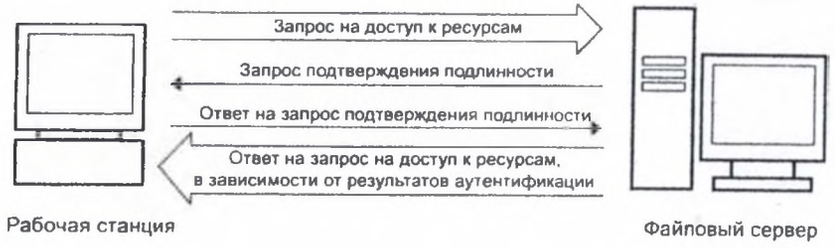


Рисунок 3.47. Схема диалога «Рабочая станция — файловый сервер»

Работа подсистемы усиленной аутентификации основана на том, что обладатель секретного ключа доказывает, что он может вычислять некоторую функцию, зависящую как от секретного ключа, так и от аргументов, задаваемых проверяющим. Проверяющий, даже зная эти аргументы, не может по значению функции восстановить секретный ключ. Но проверяющий может удостовериться в правильности её вычисления.

Код аутентификации (КА) зависит от текста сообщения, секретного ключа и несекретного ключа. При отсутствии секретного ключа подделка КА невозможна. В то же время, правильность КА можно проверить, зная лишь только несекретный ключ. Несекретный ключ используется для проверки подлинности сообщения и КА, а также для предупреждения мошенничества со стороны заверяющего в виде отказа его от подписи сообщения.

Каждая рабочая станция в сети обладает уникальным набором параметров с точки зрения усиленной аутентификации, а именно:

— пара ключей (секретный и несекретный), причём секретный ключ данной станции хранится зашифрованный на секретном ключе пользователя;

— уникальное имя станции;

— номер сетевой карты.

Список всех станций с несекретными параметрами есть на каждой станции в составе локальной вычислительной сети. Причём, этот список также снабжается КА рабочей станции, которая проверяется при каждом обращении к этому списку за открытыми ключами других станций и серверов.

Предлагаемая система усиленной аутентификации предоставляет дополнительный механизм проверки подлинности рабочих станций в момент запроса доступа к ресурсам файлового сервера с учётом всех перечисленных требований. Секретный ключ станции хранится в закодированном виде, причём кодируется он на секретном ключе пользователя, который, в свою очередь, хранится вне ЭВМ — в Touch Memory пользователя.

Потенциально слабым местом во всей этой системе является генерируемая случайная последовательность, так как в случае её повторения существует вероятность успешного преодоления всей системы усиленной аутентификации. Для того чтобы получить какие-либо гарантии того, что вероятность этого события достаточно мала, необходимо провести тщательный выбор генератора случайных чисел. В контроллерах СЗИ «Аккорд» используется аппаратный датчик случайных чисел (ДСЧ).

В отличии от многих систем безопасности, возлагающих на конечного пользователя ответственность за соблюдение секретности, предлагаемая система автоматически выполняет все необходимые операции и не требует активного участия конечного пользователя в процессе прохождения информации.

7.1.2. Основы функционирования подсистемы усиленной аутентификации

Типичный сеанс усиленной аутентификации выглядит так:

— пользователь стандартными средствами подсоединяется к файловому серверу. В случае успешного подсоединения происходит вызов серверной части подсистемы усиленной аутентификации;

— на основании данных о соединении серверная часть подсистемы усиленной аутентификации проверяет корректность существования данной станции в защищенной сети;

— если данная станция имеет право на существование, то происходит

генерация случайной последовательности длиной сто двадцать восемь байт, которая подписывается на секретном ключе серверной части. Подписанная случайная последовательность посылается на рабочую станцию. Если в течение некоторого времени, ответ от данной станции не приходит, то фиксируется попытка несанкционированного доступа к файловому серверу и соединение разрывается;

— клиентское ПО, установленное на рабочей станции, получает подписанную серверной частью случайную последовательность и, используя несекретный ключ сервера, проверяет КА. В случае успешной проверки клиентская часть уверена в том, что диалог ведется действительно с сервером. Далее необходимо подтвердить, что доступ к серверу пытается получить легальная станция. Для этого секретный ключ станции расшифровывается на секретном ключе пользователя, и полученная от серверной части случайная последовательность подписывается на нём. После чего происходит передача подписанного пакета к серверной части;

— получив от клиентской части подсистемы усиленной аутентификации подписанный ответ, и проверив КА станции, серверная часть разрешает данное соединение, так как есть уверенность в том, что рабочая станция именно та, за кого себя выдаёт.

Таким образом, в процессе сеанса усиленной аутентификации и сервер, и рабочая станция получили достаточные основания для доверия друг другу.

Кроме подсоединения пользователя к файловому серверу подсистема усиленной аутентификации может быть задействована во время сеансных опросов организуемых администратором безопасности информации. Сеансные опросы происходят также как обычный сеанс.

7.1.3. Установка и администрирования подсистемы УА

Программное обеспечение УА запускается из autoexec.bat перед стартом ПО Аккорд 1.95./1.35:

c:\accord\cauth16.exe.

Рабочая станция

Операционная система MS DOS

Программное обеспечение УА запускается из autoexec.bat после загрузки клиентской части Novell NetWare перед входом в сеть. Фрагмент autoexec.bat может выглядеть следующим образом:

```
rem Клиентская часть Novell NetWare
c:\nwclient\lsl.com
c:\nwclient\ne2000.com
c:\nwclient\ipxodi.com
c:\nwclient\vlm.exe

rem Программное обеспечение дополнительного кодирования сетевых пакетов
c:\accord\acp.exe

rem Программное обеспечение подсистемы распределенного аудита
c:\accord\accrypt.exe
c:\accord\acmodipx.exe
c:\accord\acfile.exe
c:\accord\acshell.exe

rem Программное обеспечение подсистемы усиленной аутентификации
c:\accord\acauth16.exe

rem Вход в сеть
f:\login\login.exe

rem Аккорд 1-95
c:\accord\tmac4.exe auto
c:\accord\acrun.exe /r
```

Операционная система Windows 95/98

Программное обеспечение УА запускается из autoexec.bat. Фрагмент autoexec.bat может выглядеть следующим образом:

```
rem Усиленная аутентификация
c:\accord\acshell.exe

rem Аккорд 1-95
c:\accord\tmac4.exe auto
c:\accord\acrun.exe /r
```

Сервер NetWare

Программное обеспечение УА на сервере NetWare должно загружаться из файла autoexec.ncf сразу после загрузки драйверов сетевых карт. Модули УА загружаются следующим образом:

Если используется сервер Novell NetWare v.3.1x, то load before4x.nlm,

```
load acxauth.nlm p:xxxxx,
load aclogin.nlm, где xxxxx — базовый адрес контролера АМДЗ.
```

Ключи для aclogin.nlm :

/T:tt - где tt- временной интервал;
/C - проверка уже существующих соединений.

7.2. Подсистема распределенного аудита

Распределенный аудит осуществляется на базе программно-аппаратных комплексов средств защиты информации от НСД семейства «Аккорд» и подсистемы усиленной аутентификации.

Общие сведения

Автоматизированное рабочее место администратора безопасности информации (АРМ АБИ) на базе комплекса «Аккорд» предназначено для оперативного наблюдения за работой пользователей, оперативного управления работой пользователей, централизованного сбора журналов регистрации работы комплекса «Аккорд», управления составом рабочих станций и серверов.

СЗИ «Аккорд» обеспечивает для пользователя «прозрачный» режим работы, при котором пользователь, как правило, не замечает внедренной системы защиты. При этом, дополнительная нагрузка, связанная с эксплуатацией СЗИ, не ложится на пользователя, а замыкается на администраторе безопасности информации (БИ). В этой связи для обеспечения эффективности работы АС администратор БИ обязан досконально изучить и правильно применять возможности системы защиты информации на базе СЗИ «Аккорд».

Использование ПЭВМ с внедренными средствами защиты комплекса не требует изменения существующего программного обеспечения, необходимы лишь квалифицированное применение комплекса (правильная установка, настройка и эксплуатация в соответствии с принятыми на предприятии ПРД) и обеспечение некоторой организационной поддержки.

Как показывает практика довольно длительного применения комплекса, часто трудности заключаются в отсутствии у большинства пользователей (организаций, фирм и т.д.) установленного порядка и четких правил разграничения доступа к защищаемым ресурсам. Поэтому, именно выяснение того, что и кому в ПЭВМ (АС) доступно, а что нет, и какие действия с доступными ресурсами разрешено выполнять, а какие нет, является основным содержанием необходимой организационной поддержки.

Для выполнения этих задач, а также для обеспечения непрерывной организационной поддержки работы применяемых технических средств защиты информации, в том числе и комплекса «Аккорд», необходима специальная служба (администрация) безопасности информации (СБИ), в небольших организациях и подразделениях — администратор безопасности информации (администратор БИ). На СБИ (администратора БИ) возлагаются задачи по осуществлению единого руководства, организации применения средств защиты

и управления ими, а также контроль за соблюдением всеми категориями пользователей требований по обеспечению безопасности программно-информационных ресурсов автоматизированных систем.

АРМ АБИ предназначена для оперативного наблюдения и управления за работой пользователей ПАК СЗИ «Аккорд», работающих в составе ЛВС. В любой момент времени администратор БИ может получить информацию о том, кто работает на данной станции, версию операционной системы, под управлением которой идет работа, список задач, которые выполняются на этой станции в текущий момент времени.

Кроме того, на АРМ АБИ происходит получение журналов регистрации работ ПАК СЗИ «Аккорд» в режиме реального времени, то есть все попытки НСД тут же отображаются на экране АРМ АБИ.

Администратор БИ может просматривать все события со всех станций в одном окне. Но если возникает необходимость детального анализа работы одной станции, то можно все поступающие события выводить в отдельное окно.

Для улучшения восприятия информации, АБИ может воспользоваться системой фильтров, которые позволят выбрать только те рабочие станции или только те события, которые вызывают в данный момент времени особенных интерес.

Для лучшего понимания того, что происходит на какой-либо станции, администратор БИ может оперативно изменить уровень детальности журнала. Или, в случае необходимости, просмотреть экран выбранной рабочей станции.

С помощью АРМ администратор безопасности информации может выполнять следующие функции:

— оперативное наблюдение за работой пользователей,
— оперативное управление работой пользователей,
— централизованный сбор журналов регистрации работ СЗИ Аккорд,
— управление составом рабочих станций и серверов.

Технические требованиям

	Контроллер	Версия ПО	Версия BIOS
АРМ администратора	Аккорд 4+/4++	1.95	v.1.40
Рабочая станция	Аккорд 4+/4++	1.35,1.95,2.03	v.1.40
Сервер	Аккорд 4+/4++		v.1.40

Система функционирует в сетях Novell NetWare V3.1X, V4.X и одноранговых сетях Windows 95/98, Windows NT.

Управление работой пользователей

В случае обнаружения попытки НСД АБИ имеет возможность:

— послать сообщение пользователю;

— включить ему хранитель экрана, который может быть разблокирован только ТМ-идентификатором АБИ;

— перегрузить рабочую станцию.

Централизованный сбор журналов регистрации событий СЗИ «Аккорд»

Администратор БИ может, со своего рабочего места, получать журналы регистрации работ комплекса «Аккорд». Для этого ему достаточно выбрать соответствующий пункт меню АРМ АБИ и выбрать станции, с которых необходимо получить журналы. Все полученные журналы будут сложены в соответствующих подкаталогах с делением по датам сбора.

Тем самым администратор БИ освобождается от рутинной работы обхода всех станций и сбора локальных журналов регистрации событий

Внимание!

Доступ к журналу имеет только администратор

Управление составом рабочих станций и серверов

Все рабочие станции и сервера ЛВС согласно технологии усиленной аутентификации должны содержать файл acnode.lst. Синхронизация содержимого этого файла выполняется на АРМ АБИ. Для того, чтобы переслать этот файл необходимо выбрать соответствующий пункт меню и станции, на которые необходимо его передать.

Установка подсистемы распределенного аудита

Для установки программного обеспечения необходимо наличие установочной дискеты и идентификатора TouchMemory типа DS1996 (ТМ DS1996). На рабочих станциях должен быть установлен комплекс «Аккорд 4+/4++» с BIOS v.1.40 и ПО версии 1.95 или 1.35 , на АРМ АБИ должен быть установлен комплекс «Аккорд 4+» с BIOS v.1.12 или 1.40 и ПО версии 1.95. На сервере — комплекс «Аккорд 4М1+» (АМДЗ) с ПО версии 1.95 и BIOS v.1.40. Система функционирует в сетях Novell Netware V3.1X и V4.X.

Перед установкой комплекса необходимо добиться корректного функционирования драйвера аппаратной части комплекса. Версия драйвера должна быть 3.0 или выше.

АРМ АБИ функционирует под управлением ОС Windows 95/98 или Windows NT.

Порядок установки

Установка платы контроллера в свободный слот ПЭВМ производится в соответствии с «Руководством по установке» того типа контроллера, который входит в комплект поставки.

Установка программного обеспечения

На АРМ АБИ.

1. Установить ПО СЗИ «Аккорд 1.95» согласно «Руководству по установке»;

2. Запустить с установочной дискеты ПРА программу install.exe;

3. Выбрать вариант установки «АРМ администратора безопасности информации». Файлы, необходимые для работы, будут установлены в каталог C:\ACCORD, созданный при установке ПО СЗИ «Аккорд 1.95»;

4. Загрузив драйвер аппаратной части комплекса из autoexec.bat, запустить Windows 95;

5. Из каталога C:\ACCORD запустить программу ACSETCON.EXE. Выбрать пункт меню «Создать». На запрос ключа прикоснуться идентификатором ТМ DS1996 к съемнику информации. В идентификатор при этом заносится информация, которая будет использоваться при конфигурации рабочих станций и сервера. В каталоге C:\ACCORD создается файл ACNODE.LST. В этом файле содержатся данные об АРМ АБИ. Выйти из программы ACSETCON.EXE.

На рабочей станции Windows 95/98.

1. Установить ПО СЗИ «Аккорд 1.95»;

2. Запустить с установочной дискеты программу install.exe;

3. Выбрать вариант установки «Рабочая станция (Windows'95)». Файлы, необходимые для работы, будут установлены в каталог C:\ACCORD;

4. Загрузив драйвер аппаратной части, запустить Windows"95;

5. Из каталога C:\ACCORD запустить программу ACSETWS.EXE. В предложенном диалоге необходимо указать уникальное имя станции. В дальнейшем с АРМ АБИ станция будет доступна под этим именем. На запрос ключа прикоснуться идентификатором ТМ к съемнику информации. В идентификатор при этом заносится информация о рабочей станции и открытый ключ станции. В каталоге C:\ACCORD создается файл ACNODE.LST;

6. Эту операцию необходимо произвести на каждой рабочей станции.

На рабочей станции DOS.

1. Установить ПО СЗИ «Аккорд 1.35»

2. Запустить с установочной дискеты программу install.exe.

3. Выбрать вариант установки «Рабочая станция (DOS)». Файлы, необходимые для работы, будут установлены в каталог C:\ACCORD.

4. Загрузить драйвер аппаратной части, затем, используя программное обеспечение клиентской части Novell NetWare, войти на сервер.

5. Из каталога C:\ACCORD запустить программу ACSETWS.EXE. В предложенном диалоге необходимо указать уникальное имя станции. В дальнейшем с АРМ АБИ станция будет доступна под этим именем. На запрос ключа прикоснуться идентификатором ТМ к съемнику информации. В идентификатор при этом заносится информация о рабочей станции и открытый ключ станции. В каталоге C:\ACCORD создается файл ACNODE.LST.

6. Эту операцию необходимо произвести на каждой рабочей станции.

На сервере Novell NetWare.

1. Если используется сервер Novell NetWare V3.1x, то необходимо скопировать с установочной дискеты из каталога /SERVER модуль BEFORE4X.NLM на том SYS в каталог /SYSTEM. Загрузить на сервере модуль BEFORE4X.NLM.

2. Скопировать с установочной дискеты из каталога /SERVER все модули NLM на том SYS в каталог /SYSTEM.

3. Загрузить на сервере модуль ACSETNW командой:

LOAD ACSETNW P:xxxx, где xxxx — адрес ПЗУ контроллера «Аккорд» в шестнадцатеричном виде.

После загрузки модуля на консоли сервера выводится запрос идентификатора Touch Memory. На запрос ключа прикоснитесь идентификатором ТМ к съемнику информации. В идентификатор при этом заносится информация о сервере и открытый ключ сервера. На томе SYS создается файл ACNODE.LST.

4. Если в Вашей сети несколько серверов, повторите эту операцию для каждого сервера.

На АРМ АБИ.

1. Из каталога C:\ACCORD вновь запустить программу ACSETCON.EXE. Выбрать пункт меню «Добавить». На запрос ключа прикоснуться идентификатором ТМ к съемнику информации. Объем идентификатора DS1996 позволяет хранить данные о 31 рабочей станции и их открытые ключи. Если Вы зарегистрировали 31 станцию, то при попытке зарегистрировать следующую выдается сообщение: «В идентификаторе нет свободных страниц для записи». Информация о рабочих станциях и серверах будет считана из идентификатора ТМ и память идентификатора очищается. На экран выводится информация о станциях:

— имя станции;

— открытый ключ станции;

— список номеров сетевых карт;

— номера сети для каждой сетевой карты (если в Вашей сети несколько серверов, то для каждого сервера номер сети будет индивидуальным).

2. Если в сети остались незарегистрированные станции, повторить операцию для остальных рабочих станций.

Синхронизация файлов ACNODE.LST

Для нормального функционирования системы необходимо синхронизировать содержимое ACNODE.LST на всех серверах и рабочих станциях ЛВС. Для этого необходимо выполнить следующее:

На каждой рабочей станции Windows 95/98:

— в autoexec.bat добавить вызов :c:\accord\acshell.exe до запуска acrun.exe.

На каждой станции DOS:

— до запуска acrun.exe добавить в autoexec.bat вызов клиентской части программного обеспечения Novell NetWare, а также ACCRYPT.EXE, ACMODIPX.EXE, ACFILE.EXE, ACSHELL.EXE /F, ACAUTH16.EXE;

На каждом сервере Novell NetWare:

— загрузить на сервере модуль ACCLIENT командой:

LOAD ACCLIENT P:xxxx, где xxxx — адрес ПЗУ контроллера «Аккорд» в шестнадцатеричном виде.

На АРМ АБИ:

— загрузить ACCONNET.EXE;

— выбрать пункт меню «Разослать список станций» и разослать новый ACNODE.LST на все рабочие станции и сервера.

Перегрузить все рабочие станции ЛВС и АРМ АБИ.

Эксплуатация подсистемы

Эксплуатация подсистемы распределенного аудита (ПРА) обеспечивается запуском специального ПО на рабочих станциях и АРМ АБИ.

Рабочие станции:

DOS

Для функционирования ПРА на рабочих станциях необходимо после старта клиентской части Novell NetWare и до старта программного обеспечения «Аккорд» загрузить следующие модули:

c:\accord\ACP.EXE

c:\accord\ACCRYPT.EXE

c:\accord\ACMODIPX.EXE

c:\accord\ ACFILE.EXE
c:\accord\ACSHELL.EXE
c:\accord\ ACAUTH16.EXE.

Windows 95

Программное обеспечение ПРА запускается из autoexec.bat. Фрагмент autoexec.bat может выглядеть следующим образом:

```
rem ПРА
c:\accord\acshell.exe

rem Аккорд 1.95
c:\accord\tmac4.exe auto
c:\accord\acrun.exe /r
```

АРМ АБИ

Функции АРМ АБИ реализуются в программе console.exe. Общий вид АРМ администратора безопасности приведен на Рис.3.48.

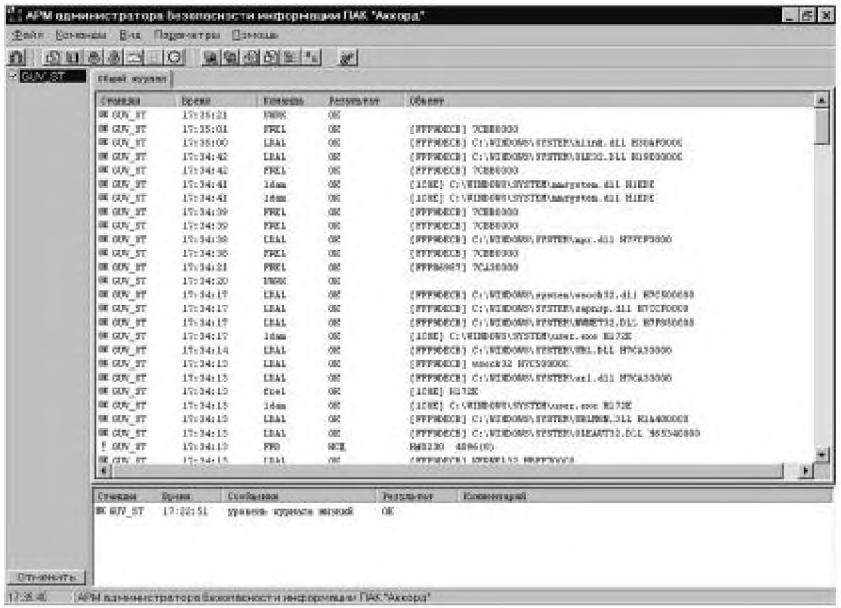


Рисунок 3.48. Общий вид АРМ администратора безопасности

Общий вид панели команд

Где:

 — Поиск станций в сети

 — Получить информацию о станции

 — Установить уровень детальности журнала

 — Заблокировать станцию

 — Разблокировать станцию

 — Отправить сообщение

 — Посмотреть экран (выбранной станции)

 — Отключить станцию

 — Получить журналы

 — Просмотреть журнал

 — Рассылка списка станций

 — Получение и редактирование файлов конфигурациикиента

 — Проводник сети «Аккорд»

 — Работа со станциями по группам

 — Очистка окон

В окне «Общий журнал» можно установить список команд DOS или Windows API, выполнение которых будет выводится на экран, а также фильтр, который устанавливает критерий вывода результатов (Рис. 3.49).

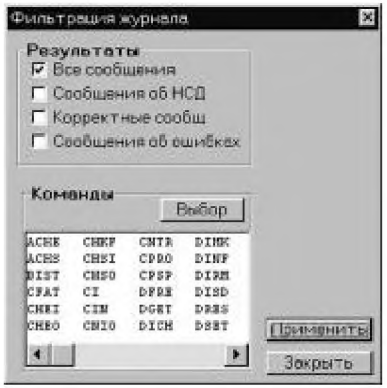


Рисунок 3.49. Выбор параметров фильтрации журнала

Меню команд

Файл

«Выход» - завершение работы с программой.

Команды

Здесь приводятся команды, которые могут быть выполнены в процессе работы администратора БИ на АРМ.

Очистка — стирание информации в выбранном окне, делится на:

— «Очистку окна вывода сообщений от станций», в результате которой стирается информация в окне вывода сообщений от станций.

— «Очистку окна вывода журнала от станций», в результате которой стирается информация в окне вывода журналов рабочих станций.

Опрос сети — с помощью этой команды можно проверить сеть на наличие подключенных станций и поиск новых.

Заблокировать станции — включить хранитель экрана на выбранных станциях.

Разблокировать станции — выключить хранитель экрана на выбранных станциях.

Получить информацию о станциях

Получение следующей информации с выбранных станций.

Окно вывода показано на рисунке 3.50.

Сетевой адрес — адрес станции в сети (IP или NIC);

Пользователь — имя пользователя, работающего на выбранной станции или сообщение «No_Acrun !»;

Тип ОС — тип операционной системы (Dos, Win95, WinNT);

Протокол — сетевой протокол (IPX или TCP/IP);

Список запущенных задач. Сетевой адрес — адрес станции в сети (IP или NIC);

Пользователь — имя пользователя, работающего на выбранной станции или сообщение «No_Acrun !»;

Тип ОС — тип операционной системы (Dos, Win95, WinNT);

Протокол — сетевой протокол (IPX или TCP/IP);

Список запущенных задач.

Рисунок 3.50. Информация о рабочей станции

Послать сообщение станциям (Рис. 3.51)

С помощью этой команды можно написать и отправить сообщение операторам выбранных станций. *Текст сообщения* — в этом окне администратор набирает сообщение, которое хочет отправить на выбранную станцию. Затем после нажатия кнопки «Отправить» сообщение будет передано на станцию.

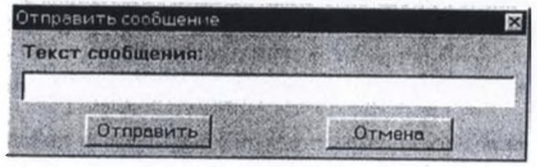


Рисунок 3.51. Рассылка сообщений рабочим станциям

Получить экран станции

Эта команда позволяет визуально наблюдать за работой пользователей. Администратор БИ получает копию графического экрана с выбранной станции.

Установить уровень детальности журнала

Отключить — не выводить журнал событий на экран;

Высокий — выводить все события, происходящие на станциях;

При работе станции, происходят обращения к функциям операционной системы, которые заносятся в журнал событий данной станции. Отбор событий происходит в зависимости от выбранного уровня детальности журнала. При максимальном уровне журнала, записываются все обращения к файловым функциям ОС. При минимальном — только запуск программ и все попытки несанкционированного доступа.

События НСД (несанкционированного доступа) фиксируются при любом уровне детальности журнала

Средний — выводить основные события, происходящие на станциях;

Низкий — выводить главные события, происходящие на станциях.

Отключить станцию

Данная команда выполняет перезагрузку выбранной станции через заданное время(Рис.3.52)

Если указано время 0 мин. — перезагрузка происходит немедленно.

Отключить — выполнить операцию.

Отменить отключение — отменить перезагрузку выбранной станции.

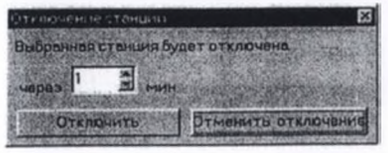


Рисунок 3.52. Принудительная перезагрузка рабочей станции

Получить журналы от станций

Эта команда позволяет переписать локальные журналы с выбранных станций на АРМ администратора, для проведения последующего их анализа.

Разослать список станций

Переслать обновлённый список станций всем станциям в сети (Рис.3.53).

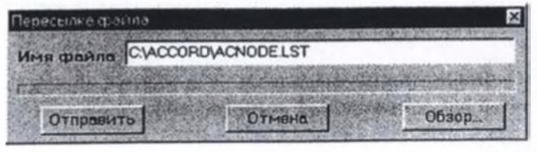


Рисунок 3.53. Рассылка списка зарегистрированных станций

Имя файла — файл, в котором находится информация о станциях.

Отправить — выполнить операцию.

Проводник сети «Аккорд»

Вызов проводника сети «Аккорд» для работы с дисками выбранной станции (сокращённый аналог проводника Windows) (Рис. 3.54).

Позволяет:

— просматривать диски выбранной станции;

— копировать и удалять файлы;

— просматривать графические файлы в формате JPEG;

— посылать сообщения выбранной станции.

Рисунок 3.54. Проводник сети «Аккорд»

Получение и редактирование файлов конфигураций станции

По этой команде можно получить и отредактировать файлы конфигурации выбранной станции (Рис. 3.55).

К файлам конфигурации относятся — config.sys, config.dos, config.win, config.w40, autoexec.bat, autoexec.dos, autoexec.w40.

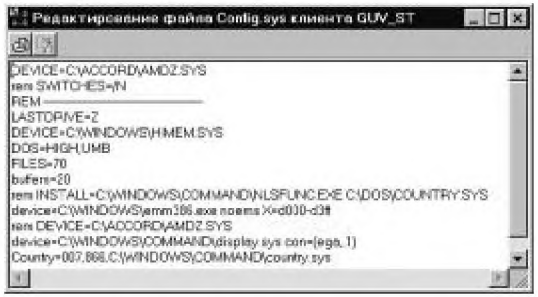


Рисунок 3.55. Редактирование файлов конфигурации выбранной станции

Вид экрана АРМ администратора БИ

Определяет способ контроля рабочих станций по списку станций, либо по группам станций (Рис. 3.56).

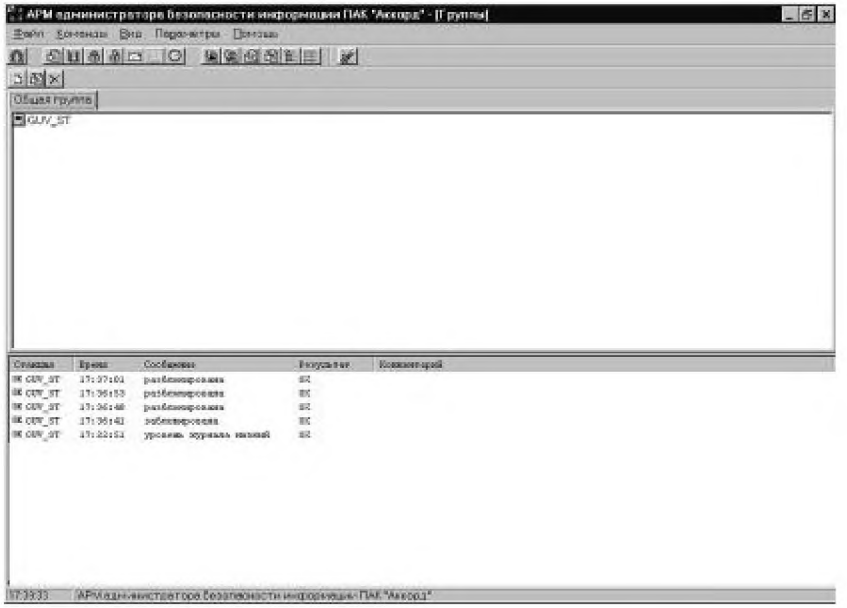


Рисунок 3.56. Работа в режиме контроля групп станций

Вид журнала

Во время работы каждого пользователя ведется журнал, в котором регистрируются его действия, которые он совершает. Администратору БИ рекомендуется в текущей работе использовать низкую детальность ведения журнала. Среднюю и высокую детальность следует использовать при изучении работы вновь используемых задач с целью определения особенностей задачи, а именно: создание новых постоянных и временных каталогов и файлов, используемых прерываний и т.д.

Общий — вывод журнала событий, приходящих от всех станций.

Персональный — вывод журнала событий, приходящих от выбранных станций.

Параметры

Конфигурация

По этой команде можно изменить следующие параметры настройки АРМ(Рис. 3.57).

1. *Рабочий каталог* — каталог, из которого запускается АРМ. Необходим для хранения временных файлов и файлов журнала со станций.

2. *Интервал смены ключа* — время, через которое изменяется сеансовый ключ шифрования.

3. *Интервал опроса станций* — время, через которое происходит автоматический поиск станций в сети.

Автосохранение конфигурации — сохранение конфигурации при выходе из программы.

Применить — выполнить операцию.

Закрыть — прервать операцию.

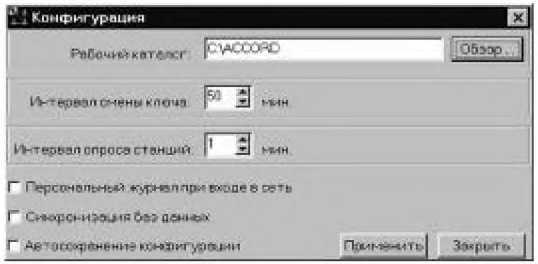


Рисунок 3.57. Настройка конфигурации АРМ АБИ

Сохранить — сохранить текущие установки АРМ.

ПРИЛОЖЕНИЯ

Приложение 1

Перечень сертификатов соответствия, выданных на продукцию ОКБ САПР

Наименование изделия, на который выдан сертификат	**Предназначение средства (область применения), краткая характеристика параметров**	**Документ, на соответствие требованиям которого выдан сертификат**
В Системе сертификации средств защиты информации по требованиям безопасности информации № РОСС RU.0001.01БИ00, Гостехкомиссия России		
Сертификат № 34 выдан 29.04.1996, действителен по 29.04.1999		
Программно-аппаратный комплекс СЗИ НСД «Аккорд» (версия ПО и БИОС1.3/1.10)	Комплекс, является средством защиты информации в автоматизированных системах от несанкционированного доступа и соответствует требованиям для классов защищенности АС: **«1Д»** — для произвольной программной среды ПЭВМ; **«1В»** — для функционально-замкнутой программной среды ПЭВМ с установленными алгоритмами обработки информации в автоматизированных системах.	РД «АС. Защита от несанкционированного доступа к информации. Классификация АС и требования по защите информации», 1992 РД «СВТ. Защита от несанкционированного доступа к информации. Показатели защищенности от НСД к информации»,1992
Сертификат № 36 выдан 7.05.1996, действителен по 7.05.1999		
Программно-аппаратный комплекс СЗИ НСД «Svet&Q»	Является средством защиты информации в автоматизированных системах от несанкционированного доступа и соответствует требованиям для классов защищенности: **«1В»** — для автоматизированных систем **«4»** — для сертифицированных СВТ	РД «АС. Защита от несанкционированного доступа к информации. Классификация АС и требования по защите информации», 1992 РД «СВТ. Защита от несанкционированного доступа к информации. Показатели защищенности от НСД к информации», 1992

Сертификат № 90 выдан 16.05.1997, действителен по 16.05.2000		
Программно-аппаратный комплекс СЗИ НСД «Аккорд1.35»	Комплекс (и его модификации, приведенные в приложении) является программно-техническим средством защиты информации, функционирует в ОС MS-DOS с программными средами Windows 3.1, Windows 3.11, обеспечивает защиту сетевых ресурсов сетей Novell NetWare v.3.12, 4.1, IntraNetware для класса защищенности **«1В»**.	РД «АС. Защита от несанкционированного доступа к информации. Классификация АС и требования по защите информации», 1992
Сертификат № 108 выдан 21.08.1997, действителен по 21.08.2000		
Программно-аппаратный комплекс СЗИ НСД «Аккорд Сеть-NetWare4»	Комплекс является программно-техническим средством защиты информации, функционирует в ОС Novell NetWare v.3.12, 4.1, IntraNetware и соответствует требованиям для класса защищенности: **«1В» —** для автоматизированных систем; **«4» —** для сертифицированных СВТ.	РД «АС. Защита от несанкционированного доступа к информации. Классификация АС и требования по защите информации», 1992 РД «СВТ. Защита от несанкционированного доступа к информации. Показатели защищенности от НСД к информации», 1992
Сертификат № 153 выдан 19.02.1998, действителен по 19.02.2001		
Программно-аппаратный комплекс СЗИ НСД «Аккорд 1.95»	Комплекс (и его модификации, приведенные в приложении) является средством защиты от несанкционированного доступа, функционирует в среде ОС MS-DOS v.5.0, 6.0, 6.20, 6.22, Windows 3.1, Windows 3.11, Windows 95 с интерфейсами ЛВС ОС Novell NetWare v.3.11, 3.12, 4.10, 4.11, IntraNetware и соответствует требованиям для класса защищенности **«1В»** при условии соблюдения требований эксплуатационной документации	РД «АС. Защита от несанкционированного доступа к информации. Классификация АС и требования по защите информации», 1992

Сертификат № 167 выдан 26.03.1998, действителен по 26.03.2001		
СЗИ НСД на изолированном рабочем месте и в ЛВС «Аккорд-Рубеж» (версии 1.3)	Является программно-техническим средством защиты от несанкционированного доступа к информации, функционирует в среде ОС MS-DOS v.5.0, 6.0, 6.20, 6.22, Windows 3.1, Windows 3.11, Windows 95, Windows 98 с интерфейсами ЛВС ОС Novell NetWare v.3.11, 3.12, 4.10, 4.11, IntraNetware и соответствует требованиям для класса защищенности **«1Г»**	РД «АС. Защита от несанкционированного доступа к информации. Классификация АС и требования по защите информации», 1992
Сертификат № 191 выдан 17.07.1998, действителен по 17.07.2001		
Программно-аппаратный комплекс СЗИ НСД «Аккорд АМДЗ»	Комплекс (и его модификации, приведенные в приложении) является программно-аппаратным средством идентификации и аутентификации пользователей и средством контроля целостности программной среды, функционирует на ПЭВМ типа IBM PC AT с системной шиной ISA и файловыми системами FAT12, FAT16, FAT32, NTFS, HPFS, FreeBSD и может быть использован для создания средств защиты от несанкционированного доступа к информации, соответствующих требованиям до класса защищенности **«1Б»** включительно	РД «АС. Защита от несанкционированного доступа к информации. Классификация АС и требования по защите информации», 1992
Сертификат № 219 выдан 17.03.1999, действителен по 17.03.2002		
Программно-аппаратный комплекс СЗИ НСД «Аккорд АМДЗ» (версия 1.1)	Комплекс (и его модификации, приведенные в приложении) является программно-техническим средством защиты от несанкционированного доступа к информации, функционирует на ПЭВМ типа IBM PC AT с системной шиной ISA и файловыми системами FAT12, FAT16, FAT32, NTFS, HPFS, FreeBSD и соответствует требованиям по классу защищенности **«1Д»** при условии соблюдения требований эксплуатационной документации.	РД «АС. Защита от несанкционированного доступа к информации. Классификация АС и требования по защите информации», 1992

Сертификат № 153/1 выдан 5.07.1999, действителен по 5.07.2001		
Программно-аппаратный комплекс СЗИ НСД «Аккорд 1.95»	Комплекс (и его модификации, приведенные в приложении) является программно-техническим средством защиты от несанкционированного доступа к информации, функционирует в среде ОС MS-DOS v.5.0, 6.0, 6.20, 6.22, Windows 3.1, Windows 3.11, Windows 95, Windows 98 с интерфейсами ЛВС ОС Novell NetWare v.3.11, 3.12, 4.10, 4.11, IntraNetware и соответствует тербованиям по классу защищенности **«1В»** при условии соблюдения требований эксплуатационной документации.	РД «АС. Защита от несанкционированного доступа к информации. Классификация АС и требования по защите информации», 1992
Сертификат № 246 выдан 5.07.1999 действителен по 5.07.2001		
Программно-аппаратный комплекс СЗИ НСД «Аккорд АМДЗ» (версия 2.01)	Комплекс (и его модификации, приведенные в приложении) является программно-техническим средством защиты от несанкционированного доступа к информации, функционирует на ПЭВМ типа IBM PC AT с системной шиной ISA и файловыми системами FAT12, FAT16, FAT32, NTFS, HPFS, FreeBSD и соответствует требованиям по классу защищенности **«1Д»** при условии соблюдения требований эксплуатационной документации, при использовании в качестве средства идентификации и аутентификации прользователей и средства контроля целостности программной среды может быть использован для создания средств защиты, соответствующих требованиям по классу защищенности до **«1Б»** включительно.	РД «АС. Защита от несанкционированного доступа к информации. Классификация АС и требования по защите информации», 1992

Сертификат № 258 выдан 5.08.1999 действителен по 5.08.2002		
Плата коррекции дат «Y2K RTC CORRECTION»	Предназначена для применения в ПЭВМ типа IBM PC с системной шиной ISA, соответствует **3 классу** по требованиям уровня контроля отсутствия недекларированных возможностей.	РД «АС. Защита от несанкционированного доступа к информации. Часть 1. Программное обеспечение средств защиты информации. Классификация по уровню контроля отсутствия недекларированных возможностей», 1998
Сертификат № 262 выдан 19.08.1999 действителен по 19.08.2002		
Программно-аппаратный комплекс СЗИ НСД «Аккорд Сеть-NDS»	Комплекс является программно-техническим средством защиты информации, функционирует в вычислительных сетях под управлением ОС Novell NetWare v.4.11, IntraNetware, NetWare 4.2, NetWare 5, BorderManager 3.5, Windows NT Server 4.0, HP-UX 10.20, HP-UX 10.30, Sun Solaris 2.6, Sun Solaris 7, Linux Red Hat, NCR UNIX SVR4 MP-RAS 3.0 и соответствует требованиям для класса защищенности: **«1В»** — для автоматизированных систем; **«4»** — для сертифицированных СВТ.	РД «АС. Защита от несанкционированного доступа к информации. Классификация АС и требования по защите информации», 1992. РД «СВТ. Защита от несанкционированного доступа к информации. Показатели защищенности от НСД к информации
В Системе сертификации средств и систем в сфере информатизации «РОСИНФОСЕРТ», Гостелеком России		
Сертификат № РОСИ.1003.643.С0022 выдан 27.05.1999, действителен по 1.06.2000		
Плата коррекции дат «Y2K RTC CORRECTION»	Обеспечивает корректную работу Real Time Clock (RTC) и BIOS ПЭВМ при переходе даты к 2000 году и правильное функционирование этих параметров в XX и XXI столетии, удовлетворяет требованиям нормативного документа по «Проблеме-2000» ВТУ 115.005-1999. Перечень сертифицированных характеристик приведен в приложении.	ВТУ. Информационная технология. Сертификация средств и систем в сфере информатизации. Вычислительные и программные средства и базы данных информационно-вычисли-тельных систем. Характеристики корректности функционирования при “наступлении 2000 года”. Общие технические требования»,1999

Сертификат № РОСИ. 1003.643.С0040 выдан 26.07.1999, действителен по 1.08.2000		
Прикладные программные средства комплекса СЗИ НСД «Аккорд 1.95»	Обеспечивается корректное функционирование при «наступлении 2000 года» в соответствии с требованиям нормативного документа по «Проблеме-2000» ВТУ 115.006-1999. Перечень сертифицированных характеристик приведен в приложении к сертификату.	ВТУ. Информационная технология. Сертификация средств и систем в сфере информатизации. Вычислительные и программные средства и базы данных информационно-вычисли-тельных систем. Характеристики корректности функционирования при “наступлении 2000 года”. Общие технические требования»,1999
Сертификат № РОСИ .1003. 643.С0041 выдан 26.07.1999, действителен по 1.08.2000		
Прикладные программные средства комплекса СЗИ НСД «Аккорд АМДЗ»	Обеспечивается корректное функционирование при «наступлении 2000 года» в соответствии с требованиям нормативного документа по «Проблеме-2000» ВТУ 115.006-1999. Перечень сертифицированных характеристик приведен в приложении к сертификату.	ВТУ. Информационная технология. Сертификация средств и систем в сфере информатизации. Вычислительные и программные средства и базы данных информационно-вычисли-тельных систем. Характеристики корректности функционирования при “наступлении 2000 года”. Общие технические требования»,1999
В системе сертификации ГОСТ Р, Госстандарт России		
Сертификат № РОСС RU. МЕ67. В00610 выдан 26.04.1999, действителен по 26.04.2000		
Плата коррекции дат «Y2K RTC CORRECTION»	Соответствует требованиям электробезопасности по ГОСТ Р 50377-92, нормам индустриальных помех по ГОСТ 29216-91, нормам устойчивости к индустриальным помехам по ГОСТ Р 50628-93	ГОСТ Р 50377-92, ГОСТ 29216-91 ГОСТ Р 50628-93

Приложение 2
Алгоритм вычисления хэш-функции, применяемый в комплексе «Аккорд»

В СЗИ «Аккорд» применяется специальный алгоритм вычисления хэш-функции. Схема, реализующая алгоритм хэширования, состоит из двух регистров W и H, управляющих друг другом. Регистр W содержит 16 ячеек W[0],W[1],...,W[15], а регистр H - 17 ячеек H[0],H[1],...,H[16], каждая длиной 8 бит (один байт). За один такт работы схемы ячейки регистров W и H сдвигаются в сторону младших номеров, а в ячейки W[15] и H[16] записывается соответственно:

W[15] = (W[0]^W[2]^W[8]^W[13]) + S(5, H[15])

H[16] = W[0] + S(3, H[0]) + f[k](H[1], H[6], H[16]), где:

^ — сложение по модулю 2;
+ — сложение по модулю 256;
S(L,A) — циклический сдвиг байта A на L разрядов в сторону старших разрядов;
& — логическое поразрядное “И”;
| — логическое поразрядное “ИЛИ”;
f[0](A,B,C) = {A & [C ^ 0xFF)] | [C & (B ^ 0xFF)]};
f[1](A,B,C) = [(A & B) | (B & C) | (A & C)]; f[2](A,B,C) = (A ^ B ^ C);

Выбор функции определяется номером такта.

Кроме того, при сдвиге ячейки W[11] в ячейку W[10] происходит также циклический сдвиг содержимого этой ячейки на 1 разряд в сторону старших разрядов.

Текст разбивается на блоки длины 16 байт. Эти блоки поступают по очереди на вход схемы и записываются в регистр W по байту в ячейку, начиная с W[0]. Если длина текста не кратна 16 (в байтах), то к концу текста дописываются один байт FF (в шестнадцатеричной записи), затем нулевые байты до длины кратной 16 (если они нужны). Последний блок, поступающий на вход схемы, это блок в 16 байт, в котором записана длина исходного текста в байтах.

Начальное состояние регистра H предлагается следующее:

6B 9D D4 57 CD F6 EA 58 E7 63 5B C5 27 FA 5F 9A D3

Состояние регистра H после обработки одного блока текста является начальным для обработки следующего. Состояние регистра H после обработки последнего блока объявляется сверткой текста.

При обработке одного блока схема работает 48 тактов. Первые 16 тактов для функции обратной связи регистра H выбирается f[0], следующие 16 тактов — f[1], следующие 16 тактов — f[2].

Приложение 3

Наименование и результат операций в системном журнале контроллера

Обозначение операции	Название операции
НС	Начало сеанса
ИА	Идентификация/аутентификация
КА	Контроль аппаратуры
КФ	Контроль файлов

Обозначение результата операции в журнале	Результат операции
OK	Успешное завершение
ULST	Создание списка пользователей
ITM	Незарегистрированный ТМ-идентификатор
IPSW	Неправильный пароль

Приложение 4

Рекомендации по организации службы информационной безопасности

Ответственными за защиту информации в АС (ПЭВМ) являются все руководители и отдельные пользователи (операторы) в пределах их служебной компетенции. Для непосредственной организации и обеспечения функционирования системы защиты информации, как компонента АС, в организации (на предприятии, фирме — далее по тексту организации) должны быть предусмотрены специальные органы или ответственные лица — служба безопасности информации (СБИ) или администратор безопасности информации. Сотрудники СБИ (администратор БИ) помимо безупречной репутации и полного доверия со стороны руководства организации должны обладать определенным уровнем знаний и навыков в области вычислительной техники, достаточным для ясного понимания всех видов угроз аппаратным и программно-информационным ресурсам АС (ПЭВМ) и необходимым для грамотного управления и эффективного применения средств защиты.

Организационно-правовой статус СБИ (администратора БИ).

— СБИ (администратор БИ) должны подчиняться тому лицу, которое в данной организации несет персональную ответственность за соблюдение правил обращения с защищаемой информацией;

— сотрудники службы (администратор БИ) должны иметь право доступа во все помещения, где установлена аппаратура АС и право прекращать автоматизированную обработку информации при наличии или угрозе утечки защищаемой информации;

— руководителю СБИ (администратору БИ) должно быть предоставлено право запрещать включение в число действующих новые элементы компонентов АС, если они не отвечают требованиям защиты информации;

— службе БИ (администатору БИ) должны обеспечиваться все условия, необходимые для выполнения своих функциональных обязанностей;

— численность службы должен быть достаточным для выполнения перечисленных выше функций, при этом штатный состав не должен иметь (по возможности) других обязанностей, связанных с функционированием АС.

Создаваемая структура защиты информации в ПЭВМ (АС) при применении программно-аппаратного комплекса защиты информации «Аккорд» должна поддерживаться механизмом установления полномочий пользователям ПЭВМ (АС) и управлением их доступом к информационным ресурсам. Для этого СБИ (администратор БИ) разрабатывает и вводит в действие установленным в организации порядком организационно-правовые документы по применению ПЭВМ (АС) с внедренными средствами защиты с учетом действующих нормативных и законодательных документов (см. Приложение 4).

Обязанности администратора БИ по применению СЗИ «Аккорд»:

1. На основе «Плана защиты», введенного в организации, разрабатывать таблицы разграничения доступа к защищаемым ресурсам, вводить (при установке комплекса) полномочия пользователей и корректировать их в ходе эксплуатации ПЭВМ (АС).

2. Устанавливать комплекс защиты в ПЭВМ и организовывать ее эксплуатацию с внедренными средствами защиты.

Внимание!

После установки комплекса в ПЭВМ должны быть приняты меры по обеспечению неизвлечения платы контроллера (опечатывание мастичной печатью, покрытой силикатным клеем (жидким стеклом) или др.

3. Тщательно анализировать процессы функционирования программ, которые будут закреплены за пользователями, в соответствии с этим создавать для каждого из них изолированную программную среду исполнения задачи, исходя из их функциональных обязанностей.

Внимание!

Нежелательно, чтобы программы, закрепленные за пользователями, имели возможность доступа к дискам по абсолютным секторам, возможность прямого редактирования памяти.

4. Обучать пользователей правилам обработки защищаемой информации, контролировать правильность применения ими средств защиты комп-

лекса и оказывать помощь в части организации работы на ПЭВМ с внедренным комплексом защиты.

5. Следить за целостностью, по крайней мере, следующих файлов: MS DOS.SYS (IBMDOS.SYS), IO.SYS (IBMIO.SYS), command.com (или другой интерпретатор командной строки, используемый в системе), autoexec.bat, config.sys (а также всех программ и драйверов, вызываемых из них), aced.exe, acrun.exe, aclog.ovl, accord.usr, accord.dat, tmdrv.exe и др.

6. Выявлять возможные каналы НСД к информации при применении комплекса, готовить предложения по их устранению.

7. Систематически анализировать состояние комплекса и его отдельных средств, периодически проводить их тестирование и проверку защитных функций комплекса, о чем делать отметку в формуляре.

8. Регулярно анализировать содержание системного журнала и разрабатывать меры по исключению неправильного применения комплекса пользователями.

Внимание!

Администратор должен довести до пользователей распоряжение о запрете снятия задач с выполнения при помощи выключения питания или нажатия на клавишу <RESET>.

9. Разрабатывать и вводить установленным порядком необходимую учетную и объектовую документацию (журнал учета идентификаторов, инструкции пользователям и др.).

10. Разрабатывать и утверждать в установленном порядке систему мер и действий на случай непредвиденных обстоятельств (заражение объекта ВТ новым типом вируса, фактов НСД к информации, нарушения правил функционирования системы защиты и т.д.).

11. В период профилактических работ на ПЭВМ снимать, при необходимости, комплекс с эксплуатации, о чем делать отметку в формуляре.

12. Принимать меры при попытках НСД к защищаемой информации и нарушении правил функционирования системы защиты.

Обязанности администратора БИ должны быть отражены в «Инструкции администратора безопасности информации», утвержденной соответствующим должностным лицом.

Приложение 5
Операции, регистрируемые подсистемой регистрации событий

№	*Код*	*Операция*
1	ACHE	Конец контроля целостности
2	ACHS	Начало контроля целостности
3	BIST	Буферизованный ввод строки
4	CFAT	Получить информацию о FAT текущего диска

5	CHEI	Завершение проверки целостности (вход)
6	CHEO	Завершение проверки целостности (выход)
7	CHKF	Контроль целостности файла
8	CHSI	Начало проверки целостности (вход)
9	CHSO	Начало проверки целостности (выход)
10	CI	Ввод с консоли без вывода
11	CINF	Ввод с консоли без вывода и фильтра
12	CNIO	Консольный I/O
13	CNTR	Получить/установить параметры страны
14	CPSP	Создать PSP
15	DFRE	Получить размер свободного места на диске
16	DGET	Получить текущий диск
17	DICH	Перейти в каталог
18	DIMK	Создать новый каталог
19	DINF	Получить информацию о диске
20	DIRM	Удалить каталог
21	DRES	Сброс диска
22	DSET	Установить текущий диск
23	DSPO	Вывод на дисплей
24	EMEM	Нарушение целостности ACRUN в памяти
25	EUED	ACED: Конец редактирования
26	EXCD	Получить код завершения программы
27	EXEC	Запустить программу
28	f1ST	Find1st через FCB
29	F1ST	Find1st
30	FACC	Запрет/разрешение файлового доступа
31	FATR	Установить/получить атрибуты файла
32	fCLO	Закрыть файл через FCB
33	FCLO	Закрыть файл
34	fCR	Создать файл через FCB
35	FCR	Создать файл
36	fDEL	Удалить файл через FCB
37	FDEL	Удалить файл
38	fGSZ	Получить размер файла через FCB
39	FNEW	Создать новый файл
40	fNXT	FindNext через FCB
41	FNXT	FindNext
42	FO	Открыть файл
43	FOC+	Открыть/создать файл 4.0+
44	fOP	Открыть файл через FCB
45	fRBR	Читать блок файла с произвольным доступом через FCB
46	FRD	Чтение из файла
47	fREN	Переименовать файл через FCB
48	FREN	Переименование/перемещение файла
49	fRRD	Чтение файла с произвольным доступом через FCB

50	fRSQ	Чтение последовательного файла через FCB
51	fSBA	Установить адрес блока файла с произвольным доступом через FCB
52	FSEK	Позиционирование в файле
53	FTIM	Запрос/установка даты/времени файла
54	FTMP	Создать уникальный временный файл
55	fWBR	Писать блок файла с произвольным доступом через FCB
56	FWR	Запись в файл
57	fWRD	Запись файла с произвольным доступом через FCB
58	fWSQ	Запись последовательного файла через FCB
59	GDIR	Получить текущий каталог
60	GDTA	Получить адрес DTA
61	GERR	Получить информацию об ошибке
62	GETD	Получить текущую дату
63	GETT	Получить текущее время
64	GFAT	Получить информацию о FAT
65	GPSP	Получить сегмент PSP
66	GVER	Получить версию ДОС
67	GVRF	Получить состояние флага ДОС Verify
68	HDUP	Дублировать Handle
69	HRED	Перенаправить Handle
70	IAUX	Ввод с AUX
71	ICHK	Проверка состояния ввода
72	ICLR	Ввод с очисткой
73	iDP	ИА Пароль получен
74	iDTM	ИА Дождались ТМ
75	iLOG	ИА Вход в систему
76	INL	Начало работы пользователя
77	IOCT	Функции IOCTL
78	iST	ИА Начало
79	iWP	ИА Ожидание пароля
80	iWTM	ИА Ожидание ТМ
81	KBDI	Ввод с клавиатуры
82	LOUT	Завершение работы пользователя
83	MEMA	Запросить блок памяти
84	MEMC	Изменить размер блока памяти
85	MEMF	Освободить блок памяти
86	NETM	Сеть: разное
87	NRDR	Перенаправление сетевого устройства
88	OAUX	Вывод на AUX
89	OPRI	Печать
90	PARS	Разбор имени файла
91	PRI	Печать
92	PRST	Печать строки
93	SBRK	Запросить/установить состояние флага Break

94	SETD	Установить текущую дату
95	SETT	Установить текущее время
96	SFDA	Получить адрес флага реентерабельности ДОС
97	STDA	Установить DTA
98	SUED	ACED: Начало редактирования
99	SVRF	Установка состояния флага Verify
100	SWTC	Set/Query Switchar (undocumented)
101	TERM	Завершение программы
102	TERM	Завершение программы
103	TSR	Завершить и остаться резидентом
104	UWRK	Продолжение работы
105	VGET	Получить адрес вектора прерывания
106	VSET	Установить вектор прерывания

Литература

1. Закон № 4524-1 от 19.2.93. «О федеральных органах правительственной связи и информации»
2. Закон № 5151-1 от 10.06.93. «О сертификации продукции и услуг»
3. Закон № 5485-1 от 21.7.93. «О государственной тайне»
4. Закон № 15-ФЗ от 16.02.95. «О связи»
5. Закон № 24-ФЗ от 20.02.95. «Об информации, информатизации и защите информации»
6. «Положение о государственной системе защиты информации в Российской Федерации от ИТР и от утечки по техническим каналам». Постановление Правительства РФ от 15.9.93. № 912-51.
7. «Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации». Руководящий документ. Москва. Гостехкомиссия России, 1992.
8. «Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от НСД к информации». Руководящий документ. Москва. Гостехкомиссия России, 1992.
9. «Концепция защиты СВТ и АС от несанкционированного доступа к информации». Руководящий документ. Москва. Гостехкомиссия России, 1992.
10. «Защита от несанкционированного доступа к информации. Термины и определения». Руководящий документ. Москва. Гостехкомиссия России, 1992.
11. «Временное положение по организации разработки, изготовления и эксплуатации программных и технических средств защиты информации от НСД в автоматизированных системах и средствах вычислительной техники». Руководящий документ. Москва. Гостехкомиссия России, 1992.
12. «Положение об обязательной сертификации продукции по требованиям безопасности информации». Москва. Гостехкомиссия России, 1994.
13. «Положение о лицензировании деятельности в области защиты информации». Москва. Гостехкомиссия России, ФАПСИ, 1994.
14. ГОСТ Р 50739-95 «Средства вычислительной техники. Защита от несанкционированного доступа к информации. Общие технические требования».
15. «Терминология в области защиты информации». Справочник. Москва. ВНИИ стандарт, 1993.

Глава 4.
ОПЫТ ПРИМЕНЕНИЯ КОДОВ АУТЕНТИФИКАЦИИ В АС РАЗЛИЧНОГО УРОВНЯ

Широкий опыт применения КА в АС, обрабатывающих ЭлД, еще не накоплен, но первые примеры уже есть. Ниже описаны три известных применения КА в АС различного уровня. Так, в контрольно-кассовых машинах (ККМ) КА используются как средства аутентификации чеков (как одного из видов ЭлД). В системе контроля целостности и подтверждения электронных документов КА используются практически по такой же схеме, как и другие трейлеры безопасности. Принципиальным отличием является здесь схема управления ключами. Эта система позволила обеспечить контроль каждого отдельного ЭлД, не увеличивая значительно трафик, т.к. КА значительно короче других трейлеров безопасности.

Еще в большей степени преимущества КА демонстрируются в подсистеме технологической защиты ЭлД. Основным методически значимым результатом здесь является то, что применение КА для этих целей продемонстрировано, а вот другие виды трейлеров применить в указанной схеме вообще вряд ли возможно. Действительно, кто будет хранить в тайне секретный ключ ЭЦП, если подписывающим субъектом является не пользователь, а операция? Естественно, можно сделать устройство, аналогичное «Аккорд СБ» для ЭЦП, а не для КА, но оно будет работать медленнее, трейлер будет перегружать трафик и архивы, усложнять управление ключами.

Возможно, изучение данного раздела позволит читателю сделать вывод о том, что применение ЭЦП целесообразно, когда субъектом является пользователь. В тех же случаях, когда субъект — это операция, процесс, объект информатизации и т.д., лучше использовать КА.

1. ПРИМЕНЕНИЕ КОДОВ АУТЕНТИФИКАЦИИ В ККМ

1.1. Низкая эффективность применения ККМ и возможность ее повышения

Единственным разумным средством повышения собираемости налогов и снижения значимости теневой части экономики является контроль финансовых потоков.

Контролировать финансовые потоки можно с двух сторон — со стороны плательщика и со стороны получателя. Здесь нужно отметить, что не существует и, видимо (к счастью), не будет существовать законных (конституционных) способов, не нарушающих прав человека, осуществлять тотальный контроль со стороны плательщика. Контроль же со стороны продавца (поставщика) товаров (услуг) организовать вполне реально и сделать это следует обязательно, тем более, что для этого необходимо всего-лишь установить контрольно-кассовые машины (ККМ) во всех точках, где оплата товаров (услуг) осуществляется за наличный расчет.

Нормативной базой применения ККМ является «Закон Российской Федерации от 18 июня 1993г. № 5215-1 «О применении контрольно-кассовых машин при осуществлении денежных расчетов с населением». Внедрению ККМ в последние годы уделялось много внимания, создавались различные структуры, вплоть до межведомственных, с целью создания системы сертификации, регистрации и аттестации ККМ, но до сегодняшнего момента положительных результатов это не принесло. Более того, появилась огромная армия чиновников, которых нужно содержать, органы сертификации, услуги которых отнюдь не бесплатны — следовательно увеличивается цена ККМ, и т.д. и т.п. Роста собираемости налогов от этих мероприятий не было.

Причины этого сейчас очевидны — в ККМ, которые использовались в тот период времени, вообще отсутствовали средства контроля итогов продаж. В этой связи при исчислении налогов можно было опереться только на отчетные материалы, предоставленные самим продавцом, методов инструментального контроля при этом не существовало. Сложилась ситуация, при которой могла подаваться отчетная документация, полностью не соответствующая действительности, либо предприниматель, желающий скрыть свои доходы приобретал два комплекта ККМ, один из которых применялся при расчете с покупателем, второй — для изготовления фиктивной контрольной ленты. Выявить махинации такого рода можно только оперативным путем, и только относительно некоторой конкретной покупки/продажи (контрольной закупки). Естественно, что примеры анализа такого рода неизвестны — у налоговых организаций нет ни необходимого количества личного состава для тотального анализа такого рода, ни достаточного количества профессионалов, умеющих выполнять такой анализ. Тем более, что по существующей нормативной базе контрольные ленты должны храниться весьма ограниченный период времени (15 дней после инвентаризации), а по истечении этого периода искать следы

вообще бесполезно. Такое положение вещей создавало широчайший простор для махинаций всякого рода, и в этой связи даже рассчитывать на рост уровня собираемости налогов не приходилось.

Учитывая сложившуюся ситуацию, а также мировой опыт, было принято решение о введении в состав ККМ блока фискальной памяти (ФП). Данный блок должен представлять собой некоторую защищенную энергонезависимую память, в которой в процессе работы ККМ должны фиксироваться данные об итогах продаж.

Переход на ККМ с блоком ФП потребовал много времени и финансовых затрат, но никак не повлиял на уровень собираемости налогов. В этом нет ничего удивительного — решение, как обычно, было принято некомплексное, и оставляло слишком много возможностей для злоупотреблений. Вот лишь некоторые из них:

1. Программные и технические решения ККМ и блока ФП не анализировались по критериям защиты от несанкционированного доступа (НСД), что создавало (и создает) предпосылки для несанкционированных модификаций программ и данных.

2. По-прежнему отсутствуют средства автоматизированного контроля отчетных данных и данных, фиксируемых в ФП, а сколько-нибудь полная проверка ручными методами невозможна, в том числе в силу причин, описанных выше.

3. До сих пор решение о том, анализировать фискальный отчет или нет принимает налоговый инспектор. Не говоря уже о том, что такой путь создает предпосылки для коррупции, анализ вообще очень проблематичен — ведь для того, чтобы получить фискальный отчет, налоговый инспектор должен прибыть непосредственно на место, где установлена ККМ, далее, в зависимости от типа ККМ, выполнить действия по изготовлению фискального отчета (а типов ККМ много, даже у профессионала работа с фискальными отчетами на некоторых типах ККМ может вызвать затруднения), затем вернуться на свое рабочее место, и лишь затем приступить к анализу. Естественно, единственной гарантией того, что на пути от торговой точки до инспекции фискальный отчет не будет подменен, является кристальная честность инспектора.

Сегодня ясно, что проблема должна решаться только комплексно (если мы вообще хотим её решить) — от накопления фискальных данных до их сбора и обработки.

Этот процесс может быть организован следующим образом. Каждая ККМ должна быть снабжена блоком интеллектуальной фискальной памяти (ИФП), которая кроме функций накопления данных об итогах продаж, выполняет еще ряд функций, а именно:

— обеспечивает защиту программного обеспечения ККМ и данных от НСД;

— вырабатывает коды аутентификации как ККМ, так и каждого чека — это позволит защитить в том числе торговую организацию от подделки чека;

— поддерживает типовой интерфейс взаимодействия с модулем налогового инспектора;

— обеспечивает съем фискальных данных и запись на носитель с энергонезависимой памятью для представления в налоговую инспекцию одновременно с балансом. Эта информация должна защищаться от несанкционированных модификаций специализированным кодом аутентификации, который может быть верифицирован непосредственно в налоговой инспекции.

Имея такой механизм, каждая организация при регистрации ККМ предъявляет в налоговую инспекцию, кроме паспорта ККМ, еще и электронный ключ, в котором зафиксирована информация для верификации отчетов. В дальнейшем, каждая сдача баланса сопровождается предъявлением ключей ККМ, которые одновременно служат энергонезависимой памятью для хранения фискальных отчетов.

Принимая баланс, сотрудник налоговой инспекции прикладывает предъявленный ему ключ к съемнику своей ККМ, и сверяет данные, отраженные в фискальном отчете с данными, приведенными в балансовом отчете. Одновременно данные будут проверяться на подлинность, и заноситься в базу данных для их возможного анализа в дальнейшем.

Такой механизм не требует от участников процесса никаких дополнительных действий и умений, но позволит очень эффективно и оперативно выявлять нарушения и предупреждать ошибки в отчетности.

В свою очередь, те данные, которые накапливаются в налоговой инспекции, могут обрабатываться в процессе полного анализа. При этом, если возникает необходимость, отчеты с фискальной памяти могут быть сняты уже «под акт» и использоваться в стандартных процедурах налоговых проверок.

Таким образом, необходимо, по крайней мере, решить следующие вопросы.

1. Разработать типовой интерфейс взаимодействия с блоком ФП, и на его основе, создать спектр блоков для всех типов ККМ. Интерфейс должен обеспечить возможность снятия фискальных отчетов с защитой от несанкционированных модификаций и передачи в электронной форме в налоговую инспекцию вместе с балансовым отчетом.

2. Разработать компьютерные средства для налоговой инспекции, позволяющие вводить в ПЭВМ фискальные отчеты в электронной форме при сдаче балансовых отчетов. Эти средства должны позволять:

— фиксировать целостность представленных отчетов;

— проводить экспресс-анализ на соответствие с данными, представленными в балансовом отчете (адекватность отчетов);

— накапливать данные для последующего подробного анализа (в случае необходимости).

3. Нормативно закрепить порядок предоставления и анализа фискальных отчетов в электронном виде, а также порядок разбора конфликтов в случаях нарушений (несоответствий).

Наличие в составе ККМ фискальной платы, с учетом сказанного выше,

позволяет говорить о возможности учета интересов двух участников процесса — продавца и государства. Однако существует еще и третий участник — покупатель, взаимодействующий с продавцом в процессе приобретения товара. Естественно необходимо учесть интересы сторон, возникающие при этом.

Обычно единственным документом, подтверждающим факт розничной покупки, является чек ККМ, который продавец обязан передать покупателю при оплате товара (услуг). Именно на основе чека должны разрешаться конфликты, возникающие в паре продавец — покупатель. Действительно, любой конфликт может быть разрешен, если подлинность чека установлена. Но вот это подчас становится весьма нетривиальной задачей. Широко распространенные сегодня технические средства позволяют без затруднений изготовить любой бумажный документ. Как быть, например, владельцу магазина, к которому с таким чеком и с бракованным товаром, якобы приобретенным в данном магазине, явится злоумышленник, и, ссылаясь на закон о правах потребителей, потребует вернуть деньги? Выход один — должен быть реализован механизм, позволяющий однозначно установить, был ли предъявляемый чек изготовлен на данной ККМ (является ли он подлинным). Это означает, что каждый чек должен быть снабжен кодом аутентификации, учитывающим параметры, важные для разбора вероятных конфликтов. По сути, КА позволяет установить подлинность тех параметров чека, которые являются и параметрами КА. В первую очередь, это:

— номер ФП;
— номер ККМ заводской;
— номер ККМ регистрационный;
— ИНН владельца ККМ;
— номер чека;
— дата чека;
— время чека;
— сумма продажи;
— нарастающий итог.

Такой состав параметров позволяет надежно защитить от подделки чеков как продавцов, так и покупателей.

1.2. Особенности применения контрольно-кассовых машин (ККМ) в системах массовых платежей

В соответствии с Законом Российской Федерации от 18 июня 1993г. № 5215-1 «О применении контрольно-кассовых машин при осуществлении денежных расчетов с населением», эти расчеты должны осуществляться с обязательным применением ККМ, внесенных в Государственный реестр ККМ и зарегистрированных в налоговых органах.

В целом, этот закон выполняется — различные типы контрольно-кассовых машин встречаются почти в каждой торговой точке. И, пожалуй, единственное место, где закон постоянно нарушается — это системы массовых

платежей, — то есть, как раз там, где соблюдение норм безопасности является наиболее важным.

Насколько нам известно, ни в одной из систем массовых платежей (крупнейшие из них — Электросвязь, Почта, Сбербанк РФ, железные дороги, гражданская авиация) широко не применяются контрольно-кассовые машины (ККМ). Вместо них зачастую используются АРМ на основе ПЭВМ, что связано с необходимостью интеграции кассовых терминалов в общую информационно-вычислительную сеть этих систем.

В этом случае ККМ используется не изолированно, а является, по существу, терминалом автоматизированной системы (АС) обработки информации. Поэтому потенциальный злоумышленник имеет возможность несанкционированного доступа к конфиденциальной информации (персональной информации), хранящейся на сервере АС.

В этой связи необходимыми являются меры по защите компьютерных ККМ, работающих в составе АС, от несанкционированного доступа (СЗИ НСД).

Важнейшим элементом ККМ является блок фискальной памяти (ФП), обеспечивающий контроль проведения расчетов. Задача блока ФП — регистрировать в своей энергонезависимой памяти информацию, на основании которой налоговая служба может проверить правильность уплаты налогов предприятием. В первую очередь — это итоги сменных продаж. Предполагается, что ФП выполнена так, что доступ к данным, хранящимся в ней, может получить только налоговый инспектор, предъявив специальный идентификатор, передаваемый ему при регистрации кассового аппарата.

Естественно, что ФП является программно- аппаратным комплексом — без специальной аппаратуры нельзя обеспечить сохранность и целостность данных, без программной части невозможна интеграция с программным обеспечением компьютерной ККМ. Информация, регистрируемая в фискальной памяти, приведена в таблице 4.1.

Применяемые в настоящее время в системах массовых платежей (СМП) программно-технические комплексы на базе персональных ЭВМ, выполняющие функции контрольно-кассовых машин, не содержат блока ФП, не удовлетворяют требованиям, предъявляемым к ККМ, и подлежат замене на ККМ, включенные в Государственный Реестр. Однако, использовать в интересах СМП можно далеко не любую из компьютерных ККМ.

Таблица 4.1

Вид записи	Реквизит	Разрядность реквизита (десятичных разрядов)	Кол-во записей
Номер	Заводской номер ККМ	7	1
Фискализация ККМ	Регистрационный номер ККМ Идентификационный код владельца ККМ Дата фискализации Пароль для проведения перерегистрации и получения фискального отчета	8 12 6 5	1
Перерегистрация ККМ	Регистрационный номер ККМ Идентификационный код владельца ККМ Дата перерегестрациии Номер последнего закрытия смены Пароль для проведения перерегистрации и получения фискального отчета	8 12 6 4 5	4
Закрытие смены	Дата отчета Номер закрытия смены Итог сменных продаж Итог сменных покупок	6 4 9 9	2000
Служебная информация (обязательная)	Контрольные суммы записей Место положения запятой в регистрируемых в ФП значениях итогов Служебные индексы, признаки, флаги, служебная информация		1

Применяемые в системах массовых платежей (в том числе в СБ РФ) ККМ должны быть:

1) компьютерным (на базе стандартных комплектующих IBM — совместимых ПЭВМ);

2) обеспечивать работу в сети;

3) снабжены блоком ФП;

4) снабжены СЗИ НСД.

Рассмотрим минимальные требования к системе защиты информации ККМ и возможную организацию блока фискальной памяти.

Как уже отмечалось, в целом ряде систем массовых платежей (в первую очередь тех, которые обеспечиваются СБ РФ) содержится информация о клиентах (т.е. каждом из нас), которая относится к категории персональных данных. Законом РФ «От информации, информатизации и защите информации» персональные данные отнесены защищаемой информации. В соответствии с РД АС, обрабатывающая такие данные, должна быть сертифицирована по классу не ниже 1В. В этой связи средства защиты информации от НСД также должны иметь сертификат Гостехкомиссии России по классу не ниже 1В и обеспечивать выполнение требований РД, в том числе процедуры идентификации/аутентификации, контроля целостности, контроля запуска задач, разграничения доступа.

Требования к блоку ФП определяются ГМЭК, они перечислены в нормативных документах этого ведомства. Основные из них приведены в таблице 1, другие определяют конструктивные особенности изготовления блока.

Имеется несколько вариантов выполнения требований Закона, в том числе:

1) замена всех ПЭВМ, используемых в системах массовых платежей, на компьютерные ККМ, внесенные в Государственный реестр ККМ;

2) доработка используемых ПЭВМ до уровня ККМ.

Обратим внимание на то, что в каждом из этих случаев обязательными является использование как СЗИ НСД, так и блока ФП.

Рассмотрим эти пути.

По имеющимся оценкам, для прямого выполнения Закона существующие платежные системы необходимо оснастить компьютерными ККМ в количестве около 150 000 штук, что связано с затратами в объеме около 700,0 млн. долларов США. При этом все приобретаемые ККМ необходимо будет дополнительно оснастить системой защиты от НСД, так как этого требует схема их применения в составе АС.

Однако есть и другой путь, значительно более экономически целесообразный. Он состоит в том, чтобы доработать ПЭВМ, используемые в системах массовых платежей, до уровня, отвечающего требованиям, предъявляемым к контрольно-кассовым машинам. Это означает в первую очередь, что ПЭВМ должна быть дополнена блоком фискальной памяти и системой защиты от НСД.

С целью сокращения объемов затрат была выполнена разработка блока фискальной памяти, обеспечивающего также защиту информации от несанкционированного доступа. Оснащение блоком ФП стандартного персонального компьютера позволит использовать его в качестве ККМ, выполняя все требования нормативных документов, а также предотвращая потенциальные потери от угроз информационных атак.

Данное техническое решение применено в ККМ «Аккорд КФ», включенной в Государственный реестр ККМ.

Стоимость блока ФП — около 220 долларов США, что в 15 раз эффективнее, чем приобретение импортных ККМ.

Следует иметь в виду, что инвестируемые в эту программу средства будут инвестированы в отечественную промышленность, а не будут вывезены за рубеж, что позволит поддержать конверсионные программы и развитие отечественных наукоемких производств.

Разработанный блок ФП «Аккорд ФП» выполнен на основе СЗИ «Аккорд», имеющей сертификат Гостехкомиссии России по классу 1В. Этот комплекс обеспечивает все необходимые защитные функции, а именно:

— защиту от несанкционированного доступа;

— идентификацию некопируемым уникальным идентификатором;

— аутентификацию с защитой от раскрытия пароля;

— защиту от несанкционированных модификаций программ и данных;

— контроль целостности программ и данных;

— функциональное замыкание информационных систем;

— исключение возможности несанкционированного выхода в ОС.

Фискальные функции блока «Аккорд ФП» выполнены в соответствии с требованиями ГМЭК. Для интеграции с программным обеспечением АС разработаны программные модули, реализующие необходимые функции.

Ниже кратко опишем основные особенности блока «Аккорд ФП».

Защищенная энергонезависимая память контроллера блока разделена на три зоны: в первой (32К) записано программное обеспечение СЗИ НСД, во второй (16К) — размещены энергонезависимые регистры ФП, в третьей части — регистрируются фискальные данные, и размещено программное обеспечение модуля налогового инспектора.

Особенностью здесь является следующее.

1. Функции СЗИ НСД интегрированы с функциями ФП. В этой связи при применении блока «Аккорд ФП» не требуется применять дополнительных средств обеспечения информационной безопасности, достаточно лишь настроить программное обеспечение СЗИ на выполнение защитных функций.

2. В составе блока ФП выполнены также энергонезависимые регистры ККМ. Это решение принято по следующим основным соображениям:

Первое — это соображение безопасности. Действительно, если регистры ККМ реализованы прикладной программой и хранятся в виде файлов на

диске ПЭВМ (как это делается практически во всех известных компьютерных ККМ), то без применения средств защиты от НСД предотвратить их несанкционированную модификацию (обеспечить целостность) практически невозможно. В этом случае в ФП могут записываться уже модифицированные данные, что, естественно, никак не укрепит финансовую дисциплину в стране.

Второе — это соображение интегрируемости. Так, раннее программист, разрабатывающий программу для ККМ, должен был сам заботиться об организации и взаимодействии всех регистров финансовых данных. При этом он должен был задумываться и о функциональности системы, и о безопасности (см. выше). Интеграция блока ФП, не обеспечивающего ведение регистров, потребовала бы значительной усилий от программистов и, скорее всего, была бы невозможной в масштабах СБ РФ. Механизм, применяемый в блоке «Аккорд ФП» делает такую интеграцию совершенно безболезненной.

3. Процедуры модуля налогового инспектора также должны быть интегрированы в состав блока «Аккорд ФП» как его неотъемлемая часть. Это обеспечивает следующие преимущества:

— целостность этих процедур обеспечивается процессом производства, и они не могут быть изменены при недобросовестной интеграции в ККМ или эксплуатации ККМ

— унифицируется работа налогового инспектора, так как порядок работы с ФП становится независимым от разработки прикладного программного обеспечения ККМ.

Нередко предлагается некоторый третий путь фискализации данных — применение так называемых «фискальных принтеров». При этом имеется ввиду, что к ПЭВМ подключается принтер, который обладает дополнительным процессором и энергонезависимой памятью, что и позволяет ему фискализировать все данные, печатаемые на чековой ленте.

При всей внешней привлекательности такого решения оно не свободно от недостатков. Во-первых — высокая цена, во-вторых — отсутствие возможностей защиты от НСД. Действительно, особые свойства принтера никак не могут усилить защищенность информации на ПЭВМ и сервере АС, а это означает, что средства защиты от НСД все равно нужно применять. Таким образом, применение фискальных принтеров нецелесообразно для систем массовых платежей.

На основании указанного выше можно сформулировать основные требования к защите информации в ККМ.

1.3. Защита информации в ККМ

1.3.1. Архитектура защиты

Блок фискальной памяти ККМ должен содержать защищенный от несанкционированных изменений и доступа программный модуль автоматического тестирования и контроля целостности, проводимых при перезагрузке ККМ. Данный программный модуль необходимо размещать в неизменяемой программным путем области памяти фискального блока и его работа должна осуществляться без загрузки исполняемого кода с других носителей информации. Программный модуль должен минимально обеспечивать:

— идентификацию и аутентификацию пользователя ККМ;

— проверку номера ФП;

— проверку заводского номера ККМ;

— проверку целостности объектов операционной среды ККМ и ППП и сравнение вычисленных значений с эталонными значениями, хранящимися в накопителе фискальных данных;

— проверку неизменности (целостности) фискальных данных, хранящихся в накопителе;

— проверку наличия печатающего устройства (устройств), входящих в состав ККМ и сигналов его (их) готовности (состояния);

— выработку сигналов блокировки работы ККМ и выдачу сообщений о результатах тестирования ККМ;

— формирование результатов тестирования;

— выработку и фиксацию КПД.

Средства защиты ФП должны вырабатывать и обеспечивать возможность печати на каждом фискальном документе кода аутентификации (КА) с возможностью его проверки, который бы однозначно определял, что для конкретного фискального документа произведена запись фискальных данных.

КА формируется для каждого фискального документа минимально из следующих параметров:

— информации, хранящейся в ФП (заводской номер ФП, заводской номер ККМ, регистрационный номер ККМ, идентификационный код владельца ККМ);

— номера документа;

— даты документа;

— времени документа;

— итога документа;

— нарастающего итога.

1.3.2. Гарантии свойств системы защиты и контроля целостности

Гарантии достоверной работы механизмов защиты реализуются на основе формирования функционально замкнутой среды ККМ, включающей только сертифицированное и проверенное на целостность ППП. Функционально замкнутая среда обеспечивается заданием только явных разрешений на запуск модулей ККМ. Механизмы защиты должны обеспечивать запрет загрузки операционной среды с внешних (по отношению к ККМ) устройств.

1.3.3. Регистрация

В модуле ФП должен быть организован минимальный набор энергонезависимых регистров и областей памяти с обеспечением их защиты от несанкционированного доступа программно-аппаратным способом:

1 — сменный денежный регистр продаж — обнуляется при закрытии смены (увеличивается при продаже);

2 — сменный денежный регистр покупок — обнуляется при закрытии смены (увеличивается при покупке);

3 — сменный счетчик продаж — обнуляется при закрытии смены (увеличивается при продаже);

4 — сменный счетчик покупок — обнуляется при закрытии смены (увеличивается при покупке);

5 — общий счетчик продаж закрытых смен — обнуляется при перерегистрации (увеличивается при закрытии смены);

6 — общий счетчик покупок закрытых смен — обнуляется при перерегистрации (увеличивается при закрытии смены);

7 — дата начала (конца) смены — изменяется при открытии (закрытии) смены;

8 — время начала (конца) смены — изменяется при открытии (закрытии) смены;

9 — дата последней продажи (покупки) — обнуляется при закрытии смены (изменяется при продаже/покупке);

10 — время последней продажи (покупки) — обнуляется при закрытии смены (изменяется при продаже/покупке);

11 — наименование предприятия;

12 — номер последнего фискального отчета — не обнуляется;

13 — регистр флагов (фискальный/нефискальный режим, выполнена перерегистрация, переключатель закрытия смены и т. п.).

1.3.4. Тестирование

Средства защиты должны обеспечивать тестирование правильности работы аппаратных и программных средств ККМ, а также собственно системы защиты с необходимыми количественными показателями.

1.3.5. Сигнализация

Средства ККМ должны предусматривать возможность реализации звуковой и визуальной сигнализации о НСД, неисправностях и нарушениях целостности ПО и данных.

Блок фискальной памяти ККМ должен содержать аппаратно защищенный от несанкционированных изменений и внешнего доступа программный модуль налогового инспектора. Данный программный модуль также необходимо размещать в ПЗУ фискального блока и его работа должна осуществляться без загрузки исполняемого кода с других носителей информации, что может гарантировать правильную выдачу фискальной информации при посещении налоговым инспектором мест расположения ККМ.

1.3.6. Блокирование

Средства защиты ФП должны обеспечивать следующие блокировки:

— блокировку всех операций кроме чтения ФП, при обнаружении испорченных фискальных данных в ФП;

— блокировку попыток изменения местоположения десятичной точки до операции перерегистрации;

— блокировку всех операций кроме чтения ФП, при переполнении ФП;

— блокировку всех операций при неисправности ФП;

— блокировку всех операций при отсутствии ФП;

— блокировку попыток подбора пароля доступа к фискальным данным;

— блокировку попыток выполнения любых действий с ФП до записи заводского номера;

— блокировку повторной записи заводского номера;

— блокировку попыток выполнения любых действий с ФП, кроме чтения и записи заводского номера, до проведения фискализации;

— блокировку попыток выполнения любых действий с использованием пароля доступа к ФП, до записи суточного (сменного) итога в ФП;

— блокировку записи в счетчики и регистры, организованные в отдельной энергонезависимой области фискальной памяти и блокировку выработки КПД при невыполнении следующих условий проверки даты и времени:

1) текущая системная дата не может быть более ранней, чем дата последней записи в ФП, включая и дату проведения операции фискализации или перерегистрации;

2) текущие системные дата/время не могут быть более ранними, чем дата/время последней продажи/покупки;

3) дата документа не может быть более ранней, чем дата последней записи в ФП, включая и дату проведения операции фискализации или перерегистрации;

4) дата/время документа не должны более чем на 10 сек отличаться от текущих системных даты/времени;

5) дата/время документа не могут быть более ранними, чем дата/время начала смены;

6) дата/время документа не могут быть более ранними, чем дата/время последней продажи/покупки;

7) дата/время документа не могут быть более чем на 24 ч поздними, чем дата/время начала смены.

1.3.7. Управление и сервис

Для автоматизации снятия налоговыми органами фискальных показаний блок фискальной памяти ККМ должен содержать защищенный от несанкционированных изменений и внешнего доступа модуль, обеспечивающий снятие фискальных показаний, а также дополнительной информации, необходимой налоговым органам (например: сумм налогов, подсчитанных ККМ, сведения о крупных покупках, электронный вариант контрольной ленты и т.п.); их запись на внешний носитель информации (например: электронный ключ, дискета, смарт-карта и т.п.) с защитой от несанкционированных модификаций и их прочтения продавцом (поставщиком) товаров (услуг); обязательную передачу этого внешнего носителя в налоговую инспекцию вместе с балансовой документацией. Из налоговой инспекции на том же внешнем носителе передается подтверждающий код, разрешающий дальнейшую работу ККМ. Данный программный модуль необходимо размещать в составе ПО фискального блока и его работа должна осуществляться без загрузки исполняемого кода с других носителей информации. Обязательность передачи носителя информации в налоговую инспекцию гарантируется тем, что без получения подтверждающего кода (что информация налоговыми органами получена) дальнейшая работа ККМ будет невозможна.

2. СИСТЕМА КОНТРОЛЯ ЦЕЛОСТНОСТИ И ПОДТВЕРЖДЕНИЯ ДОСТОВЕРНОСТИ ЭЛЕКТРОННЫХ ДОКУМЕНТОВ (СКЦПД)

Рассмотрим большую АС (федерального или регионального уровня), элементы которой размещены на значительном удалении один от другого. Как правило, в такой АС циркулирует значительное число ЭлД, обрабатываемых (используемых) большим числом участников (пользователей АС). Это означает, что любой пользователь в любой момент времени должен иметь возможность установить достоверность любого ЭлД, вне зависимости от того, где, когда и кем этот ЭлД был изготовлен (сквозной контроль). Более того, если в процессе проверки будет установлено нарушение достоверности ЭлД, должны быть механизмы, позволяющие провести анализ нарушения, несмотря на то, что за время от изготовления ЭлД до его проверки ключи вполне могли смениться.

Основой для создания такой системы может быть «Аккорд СБ/КА».

СКЦПД предназначена для обеспечения сквозного контроля целостности и подтверждения достоверности электронных документов, создаваемых ответственными сотрудниками в составе АС федерального или регионального уровня, хранимых в информационных системах, передаваемых по каналам связи и обрабатываемых в информационных системах в соответствии с используемой в АС технологией.

2.1. Требования к средствам СКЦПД

Список принятых сокращений, термины и определения.

ЭлД — электронный документ;

СКЦПД — система контроля целостности и подтверждения достоверности электронных документов;

КА — код аутентификации ЭлД;

РЦ — региональный центр СКЦПД;

N — количество операторов-участников СКЦПД;

i(НОП) — условный номер оператора-участника СКЦПД, $1 \leq i \leq N$;

А-СБ/КА — устройство «Аккорд СБ/КА»;

АРМ — автоматизированное рабочее место;

АРМ КА — АРМ оператора-участника, оснащенное А-СБ/КА и программными средствами СКЦПД;

АРМ-К — АРМ РЦ для изготовления ключей, используемых в СКЦПД для передачи проверочных данных;

АРМ-Р — АРМ РЦ для изготовления и подготовки рассылки проверочных данных;

M — количество задействованных в СКЦПД АРМ КА;

j(НА) — условный номер отдельного А-СБ/КА, $1 \leq j \leq M$;

КОП — персональный код оператора;

КОП(i) — персональный код оператора с условным номером i;

МОП(i) — метка (случайный идентификатор) КОП(i);

ТМ(i) — устройство DS199x — персональный идентификатор оператора с условным номером i;

INTM(i) — идентификационный (серийный) номер ТМ(i);

TMd(j) — устройство DS199x — память для хранения ключей передачи данных между РЦ СКЦПД и А-СБ/КА(j);

INTMd(j) — идентификационный (серийный) номер TMd(j);

Kb(j) — основной ключ доставки (передачи данных между РЦ СКЦПД и А-СБ/КА(j));

Kr(j) — резервный ключ доставки (передачи данных между РЦ СКЦПД и А- СБ/КА (j));

Ка — архивный ключ хранения данных на жестком диске АРМ-Р;

ТД — таблица достоверности, в которой хранятся КОП и МОП операторов, зарегистрированных к началу ее ввода в действие;

НТД — серийный (порядковый) номер таблицы достоверности (2 байта);

ДТД — дополнительная таблица достоверности, в которой хранятся КОП и МОП операторов, зарегистрированных после ввода ТД в действие;

Стоп-лист — таблица МОП операторов из действующих ТД и ДТД, заблокированных для работы в СКЦПД с указанием момента блокировки;

УПЗУ — недоступная компьютеру память устройства А-СБ/КА, предназначенная для хранения изменяемых регистрационных данных (условно-постоянное ЗУ).

Региональная СКЦПД ЭлД должна обеспечивать информационное взаимодействие удаленных АРМ КА в процессе создания и сквозной проверки КА ЭлД, а также для поддержания в актуальном состоянии размещенной во всех А-СБ/КА(j) проверочной базы данных, то есть ТД, ДТД и Стоп-листов.

РЦ должен обеспечивать управление деятельностью СКЦПД в целом, взаимодействуя при этом со всеми АРМ КА. В состав РЦ должно входить два автоматизированных рабочих места: АРМ-К и АРМ-Р.

АРМ-К выполняется в виде автономного IBM-совместимого компьютера с установленным программно-аппаратным комплексом (ПАК) «Аккорд» и специальным программным обеспечением. ПАК «Аккорд» содержит физический датчик случайных чисел, обеспечивает изолированность программной среды и поддерживает работу подсистемы идентификации/аутентификации (п/с ИА) лиц, допускаемых к работе на АРМ-К. Специальное программное обеспечение должно поддерживать выполнение в интерактивном режиме процедуры изготовления, регистрации и архивации основных и резервных ключей доставки {Kb(j), j=l...M} и {Kr(j), j=l...M}, а также процедуру изготовления архивного ключа Ка.

Ключи {Kb(j)}, j=l...M, записываются в архивную базу АРМ-К, в А-СБ/КА АРМ-Р вместе с INTMd(j), номером j и контрольной суммой (2 байта), а также в TMd(j) вместе с номером j и контрольной суммой (2 байта).

Ключи {Kr(j)}, j=l...M, записываются в TMd(j) вместе с контрольной суммой (2 байта) и хранятся в РЦ СКЦПД вместе INTMd(j), номером j и контрольной суммой (2 байта) на отдельном носителе для замены основных ключей в случае компрометации архивной базы АРМ-К или А-СБ/КА АРМ-Р.

В основном режиме для передачи данных между РЦ СКЦПД и А-СБ/КА всегда используются основные ключи доставки {Kb(j)}, j=l...M. Необходимость вывода их из действия определяется администратором РЦ СКЦПД, после чего производится одновременный переход всех участников системы на резервные ключи доставки {Kr(j)}, j=1..M. Процедура перехода на резервные ключи описывается ниже.

Изготовленные TMd(j), j=l...M, устанавливаются администратором региональной СКЦПД в устройства А-СБ/КА(j) под печать и затем эти устройства доставляются через надежные транспортные каналы на места эксплуатации.

АРМ-Р выполняется в виде автономного IBM-совместимого компьютера с установленным А-СБ/КА и специальным программным обеспечением. Реализованный в указанной конфигурации программно-аппаратный комплекс АРМ-Р должен поддерживать работу п/с ИА допускаемых к работе лиц, обеспечивать изолированность программной среды и выполнение в интерактивном режиме следующих процедур:

— загрузку и хранение в А-СБ/КА всех ключей Kb(j), j= 1...M или Kr(j), j= 1 ...M;

— прием и обработку запросов от АРМ КА;

— изготовление, хранение и подготовку рассылки на все АРМ КА конфиденциальной информации, необходимой для контроля достоверности ЭлД;

— преобразование конфиденциальной информации перед ее рассылкой в вид, доступный только определенным абонентам СКЦПД;

— осуществление совместно с АРМ КА процедуры «выравнивания таблиц кодов операторов (КОП)» - информации, используемой для контроля достоверности ЭлД.

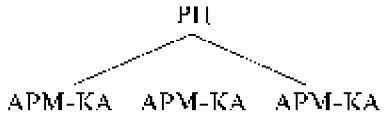
АРМ КА представляет собой компьютер, снабженный программно-аппаратным комплексом «Аккорд-СБ/КА» и программными средствами СКЦПД. Основные функции АРМ КА:

— поддержка работы п/с ИА допускаемых к работе лиц;

— обеспечение изолированности программной среды;

— изготовление и проверка кода подтверждения достоверности электронных документов.

Граф информационного взаимодействия РЦ и АРМ КА при «выравнивании КОП» описывается схемой «звезда», то есть информационный обмен ведется между РЦ СКЦПД и отдельными АРМ КА(j), $1 \leq j \leq M$. При этом в каждом направлении используются индивидуальные ключи Kb(j), $1 \leq j \leq M$.



Должна быть обеспечена возможность работы каждого оператора-участника СКЦПД (для создания или проверки КА ЭлД) на любом АРМ КА, где этот оператор зарегистрирован в п/с ИА. В этой связи граф информационного взаимодействия участников при проверке и подтверждении достоверности электронных документов описывается схемой полного графа, то есть разрабатываемые аппаратные и программные средства должны обеспечивать возможность проверки каждым участником системы КА ЭлД, созданный любым другим участником системы, при этом ключевая система должна поддерживать одновременную работу до 2000 операторов-участников СКЦПД.

КОП — код оператора — случайная последовательность длиной 14 байт + 2 байта контрольной суммы. КОП хранится в А-СБ/КА, вырабатывается на АРМ-Р по запросу оператора АРМ КА.

МОП — метка оператора — случайная последовательность длиной 8 байт. МОП хранится в А-СБ/КА, вырабатывается на АРМ-Р по запросу оператора АРМ КА. МОП записывается также в память персонального ТМ-идентификатора оператора.

ТД — таблица достоверности. Содержится в УПЗУ А-СБ/КА(j), j=1...M и на жестком диске АРМ-Р (в закодированном виде на ключе Ка).

На жестком диске АРМ-Р таблица достоверности имеет следующую структуру:

НТД (номер ТД)
дата изготовления ТД
INTMd(*) МОП(1) КОП(1)
NTMd(*) МОП(2) КОП(2)
:
:
INTMd(*) МОП(n) КОП(n)

Здесь INTMd(*) — индивидуальный регистрационный номер АРМ КА, на котором зарегистрирован данный участник в СКЦПД.

В УПЗУ А-СБ/КА(j), j=1...M таблица достоверности записывается в виде следующей структуры:

НТД (номер ТД)
дата изготовления ТД
дата загрузки ТД в А-СБ/КА(j) — вводится оператором по запросу А-СБ/КА(j)
МОП(1) КОП(1)
МОП(2) КОП(2)
:
:

МОП(n) КОП(n)

ТД формируется в РЦ на АРМ-Р и загружается в АРМ КА перед началом работы СКЦПД, а также при плановом переходе на новую ТД. Для обеспечения возможности проверки КА «задержавшихся» ЭлД в переходный период предыдущий комплект (ТД, ДТД, Стоп-лист) сохраняется в УПЗУ устройства А-СБ/КА до ввода в действие очередной новой ТД. Таким образом, в УПЗУ всегда находятся два комплекта (ТД, ДТД, Стоп-лист) — последний и предпоследний из вводившихся в действие.

ДТД — дополнительная таблица достоверности. На жестком диске АРМ-Р ДТД имеет следующую структуру:

НТД (номер соответствующей ТД)
дата изготовления ДТД
INTMd(*) МОП(n+1) КОП(n+1)
INTMd(*) МОП(n+2) КОП(n+2)
:
:
INTMd(*) МОП(N) КОП(N)

В УПЗУ А-СБ/КА(j), j=l...M дополнительная таблица достоверности записывается в виде следующей структуры:

НТД (номер соответствующей ТД)
дата изготовления ДТД
дата загрузки ДТД в А-СБ/КА(j) — вводится оператором по запросу А-СБ/КА(j)
МОП(n+1)КОП(n+1)
МОП(n+2) КОП(n+2)
:
:
МОП(N) КОП(N)

ДТД формируется РЦ при подключении новых участников СКЦПД, рассылается на все АРМ КА и загружается в А-СБ/КА, замещая ранее действовавшую ДТД;

Формирование новой ДТД производится путем последовательного добавления меток и кодов новых участников в конец действующей ДТД. При плановой смене ТД все зарегистрированные операторы, в том числе зарегистрированные по ДТД, учитываются по основной таблице, но с выработкой новых значений МОП и КОП.

НОП — номер оператора — порядковый номер КОП и МОП оператора в ТД или (с последовательным продолжением нумерации) в ДТД. Его длина 2 байта.

НА = j. Его длина 2 байта.

ХКА — значение хэш-функции h(), вычисленное от блока данных (информации) с участием КОП и НА. Длина ХКА 10 байт.

КА= (НТД,НОП,НА,ХКА). Длина КА 16 байт.

Создаваемые средства СКЦПД должны обеспечивать выполнение следующих основных процедур.

1. Персонализация А-СБ/КА.
2. Персонализация АРМ-Р.
3. Регистрация и блокировка оператора на АРМ КА в п/с ИА.
4. Назначение файлов и системных областей на контроль целостности.
5. Первоначальная регистрация оператора (операторов) в СКЦПД.
6. Регистрация оператора (операторов) в СКЦПД при плановой смене ТД.
7. Блокировка оператора в СКЦПД.
8. Изготовление КА.
9. Проверка КА.
10. Отработка файла конфигурации и формирование файла отчета.
11. Формирование файла загрузки.
12. Формирование файла сообщений.
13. Изготовление ТД.
14. Загрузка новой ТД на АРМ КА.
15. Подключение в СКЦПД нового АРМ КА.
16. Переход на резервные ключи доставки (передачи данных).
17. Тестирование А-СБ/КА.

В этом варианте система может выглядеть как на рисунке 4.1.

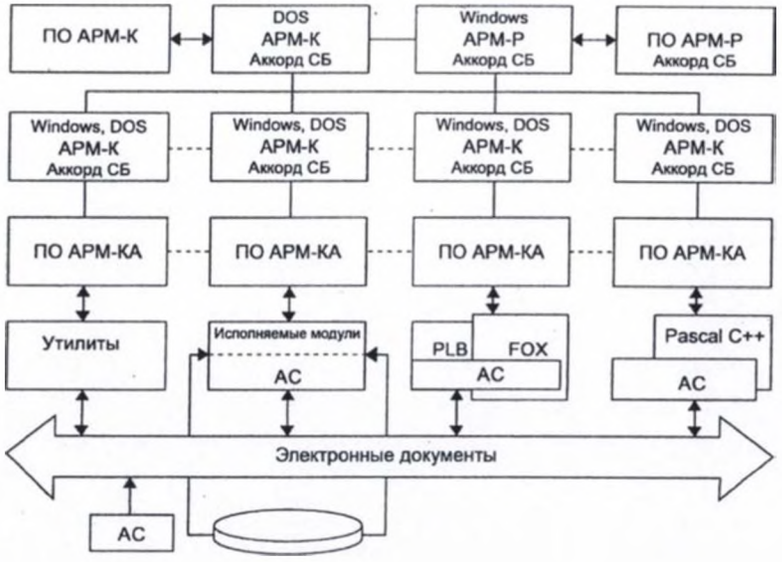


Рисунок 4.1. Схема полного контроля ЭлД

2.2. Краткое описание основных процедур

П1. Процедура персонализации А-СБ/КА (придание поступившему от изготовителя оборудованию специальных свойств)

1. Изготовить Kb(j) и Kr(j) на АРМ-К.

Ключи изготавливаются в формате

K	CRC
30 В	2В

2. Записать j, Kb(j) и Kr(j) на идентификатор TMd(j) А-СБ/КА.
3. Зарегистрировать TMd(j) на АРМ-Р.
4. Установить TMd(j) в А-СБ/КА и опечатать место установки.

П2. Процедура персонализации АРМ-Р

1. Изготовить Ка на АРМ-К.
2. Записать Ка в TMd, предназначенную к установке в А-СБ/КА для АРМ-Р.
3. Изготовить копию Ка и передать ее на хранение в установленном порядке.
4. Установить TMd в А-СБ/КА для АРМ-Р.

П3. Регистрация и блокировка оператора на АРМ КА в п/с ИА

Регистрация и впоследствии, при необходимости, блокировка оператора в подсистеме Идентификации/Аутентификации АРМ осуществляется по технологии, принятой в СЗИ «Аккорд». Данные для п/с ИА располагаются в УПЗУ. При регистрации персонального идентификатора оператора первоначально производится принудительная очистка памяти идентификатора.

П4. Назначение файлов и системных областей на контроль целостности

Назначение файлов на контроль целостности осуществляется по технологии, принятой в СЗИ «Аккорд». При этом обеспечиваются назначения на контроль целостности таких системных областей твердого диска, как таблица разделов (MBR) и загрузочных секторов разделов (BOOT).

П5. Первоначальная регистрация оператора (операторов) в СКЦПД

Эта процедура содержит три фазы:

— запрос на регистрацию нового оператора (операторов);
— регистрацию оператора (операторов) в РЦ;
— регистрацию оператора (операторов) на АРМ-КА;

П5-1. Запрос на регистрацию нового оператора (операторов)

Запрос генерируется на АРМ КА. Запрос по каждому оператору будет принят и обработан в РЦ только в том случае, если вводимая в действие ТМ(i) не зарегистрирована на данный момент в СКЦПД.

1. А-СБ/КА последовательно запрашивает ТМ(i) операторов.
2. Регистрирует ТМ(i) операторов в подсистеме ИА А-СБ/КА.
3. Вырабатывает запрос вида

< дата, НА, INTMd(j), INTM(il), INTM(i2),... >

4. Кодирует запрос с использованием действующего ключа доставки и отдает ПЭВМ.
5. ПО формирует файл запроса.

П5-2. Регистрация оператора (операторов) в РЦ

1. Из файла берется запрос, декодируется.
2. Проверяется легальность TMd(j) и ТМ(i) операторов.
3. С помощью ДСЧ АРМ-Р вырабатываются МОП(i) и КОП(i) для операторов.
4. Формируется ответ на запрос вида.

< имитоприставка на действующем ключе доставки,
дата регистрации,
INTM(i1), НОП(i1), МОП(i1),
INTM(i2), НОП(i2), МОП(i2),... >

5. Ответ кодируется на действующем ключе доставки и помещается в файл доставки.
6. Строки

INTMd(j) МОП(i1) КОП(i1)
INTMd(j) МОП(i2) КОП(i2)
.....

помещаются в ТД, если таблица еще не сформирована, или в ДТД, если ТД уже введена в действие.

7. По каждому зарегистрированному оператору информация вида

<НА, INTMd(j), INTM(i), МОП(i), КОП(i), дата регистрации> кодируется на ключе Ка и помещается в архив.

П5-3. Регистрация оператора (операторов) на АРМ-КА

1. Определяется, поступил ли файл загрузки.
2. Файл загружается в А-СБ/КА, декодируется с использованием ключа Kb(j).
3. Проверяется корректность имитоприставки блока #1 файла загрузки (см. ниже) на ключе Kb(j):

3.1. Если она корректна, то далее выполняется п.4

3.2. Если она некорректна, то файл вновь загружается в А-СБ/КА, декодируется с использованием ключа Kr(j)

3.3 Проверяется корректность имитоприставки блока #1 файла доставки на ключе Kr(j):

3.3.1. Если она корректна на ключе Kr(j), то этот ключ переносится в TMd(j) на место ключа Kb(j) и далее выполняется п.4

3.3.2. Если она некорректна на ключе Kr(j), то А-СБ/КА выдает сообщение об искажении файла доставки

4. Последовательно запрашиваются ТМ операторов.
5. Считанный INTM(i) сверяется по файлу доставки, если он там находится, то МОП(i) записывается в ТМ(i).
6. Начало работы оператора в СКЦПД возможно после загрузки новой ТД или ДТД.

П6. Регистрация оператора (операторов) в СКЦПД при плановой смене ТД

Содержит две фазы:

— выработку в РЦ новой ТД и подготовку ее рассылки;
— регистрацию оператора (операторов) на АРМ-КА.

П6-1. Выработка в РЦ новой ТД и подготовка ее рассылки

1. Путем совместного учета информации, содержащейся в текущих ТД, ДТД и Стоп-листе вырабатывается новая ТД, при этом МОП(*) и КОП(*) для каждого оператора вырабатываются заново с использованием физического ДСЧ на АРМ-Р.
2. Новая ТД кодируется с использованием ключа доставки и поступает в рассылку.

П6-2 . Регистрация оператора (операторов) на АРМ-КА

Производится аналогично процедуре П5-3

П7. Блокировка оператора в СКЦПДСодержит три фазы:

— извещение о блокировке;
— подготовка блокировки;
— блокировка.

П7-1 Извещение о блокировке

1. На АРМ КА отмечаются операторы, работу которых в СКЦПД необходимо заблокировать (например, по причине увольнения или утраты ТМ). На основании данных, хранимых в п/с ИА А-СБ/КА, формируется извещение вида:

< дата, НА, INTMd(j), INTM(i1), INTM(i2),... >

2. Извещение кодируется с использованием действующего ключа доставки и помещается в файл доставки.

П7-2 . Подготовка блокировки оператора выполняется на АРМ-Р

1. Получаемое сообщение (извещение о дискредитации, см. ниже блок С6 файла сообщений) загружается и декодируется с использованием действующего ключа доставки.

2. Проверяется легальность INTMd(j).
3. По INTM(i1), INTM(i2),... определяются МОП(i1),МОП(i2),...
4. Строки
МОП(i1),дата
МОП(i2), дата
.....
вносятся в Стоп-лист.
5. Формируется для рассылки новый файл загрузки (см. ниже).

П7-3. Блокировка

1. Определяется, поступил ли файл загрузки.
2. Файл загружается в А-СБ/КА, декодируется с использованием ключа Kb(j).
3. Проверяется корректность имитоприставки блока #2 файла загрузки (см. ниже) на ключе Kb(j):
3.1. Если она корректна, то далее выполняется п.4.
3.2. Если она некорректна, то файл вновь загружается в А-СБ/КА, декодируется с использованием ключа Kr(j)
3.3. Проверяется корректность имитоприставки блока #1 файла доставки на ключе Kr(j):
3.3.1. Если она корректна на ключе Kr(j), то этот ключ переносится в TMd(j) на место ключа Kb(j) и далее выполняется п.4.
3.3.2. Если она некорректна на ключе Kr(j), то А-СБ/КА выдает сообщение об искажении файла доставки.
4. Стоп-лист записывается в УПЗУ, замещая действовавший ранее.

П8. Изготовление КА

1. Подготовленная информация считывается в А-СБ/КА.
2. Из ТМ оператора считывается МОП.
3. Проверяется наличие данного МОП в Стоп-листе.
3. Путем поиска данного МОП в действующей ТД и ДТД, определяется КОП и НОП оператора:
НОП = i, если МОП(i) = МОП КОП=КОП(i).
4. Вычисляется ХКА = h (информация | НА, КОП).
5. Подготавливается КА = (НТД, НОП, НА, ХКА).
6. КА возвращается прикладной задаче.

П9. Проверка КА

1. Из проверяемого документа в А-СБ/КА считывается (информация) и КА.
2. По КА определяется НТД, НОП, НА, и ХКА.
3. Проверяется соответствие НТД номерам загруженных в А-СБ/КА таблиц достоверности.
4. По НОП и ТД с номером НТД определяются КОП(НОП) и МОП-(НОП).

5. Полученный МОП(НОП) проверяется по Стоп-листу.
6. Вычисляется ХКА* = h (информация | НА, КОП).
7. Если ХКА* = ХКА, то прикладной задаче возвращается TRUE + дата создания ТД.
8. Если ХКА* ≠ ХКА, то прикладной задаче возвращается FALSE + дата создания ТД.

Организация взаимодействия АРМ-Р и АРМ КА. АРМ-Р и АРМ КА взаимодействуют, обмениваясь данными, а именно:

АРМ-Р направляет на АРМ КА:
— файл запуска
— файл загрузки,
а АРМ КА направляет на АРМ-Р:
— файл сообщений.

П10. Отработка файла конфигурации

Файлы документов, обрабатываемых в СКЦПД, могут быть представлены в различных форматах, например: dbf-формате (базы данных), текстовом формате с фиксированной длиной документа (ограниченной кодами возврата каретки и перевода строки) и т.д.

Для обеспечения работы утилиты в соответствии с вышеизложенным необходимо иметь два файла текстового формата.

1. Файл конфигурации — имеет определенное имя (например KPD.CFG), находится в том же подкаталоге, что и утилита, или в каталоге, доступном через директиву PATH в файле AUTOEXEC.BAT и готовится администратором системы.

Возможные значения видов обработки, указываемые в файле конфигурации:

— вычислить и занести КА в файл;
— проверить КА без его удаления из файла;
— проверить КА с одновременным его удалением из файла (восстановлением первоначальной структуры файла);
— удалить поле КА без его проверки.

2. Файл отчета — имеет определенное имя (например , KPD.LOG) и располагается в том же каталоге, что и утилита. Формируется утилитой путем дозаписи в конец существующего файла при обработке файла документов в виде:

<ДАТА><ВРЕМЯ><ИМЯ_ФАЙЛА>
<номер строки/записи><вид нарушения контроля><код нарушения>
<номер строки/записи><вид нарушения контроля><код нарушения>
...
Например:
13 Несовпадение хэш-функции 03
167 Не найден код оператора 02
Код нарушения должен совпадать с кодом возврата утилиты.

П11. Формирование файла загрузки

Посылается из РЦ на АРМ-КА. Может содержать произвольный набор следующих блоков информации:

#1. Регистрация операторов:
< имитоприставка на действующем ключе доставки,
дата регистрации,
INTM(i1), НОП(i1), МОП(i1),
INTM(i2), НОП(i2), МОП(i2),... >
#2. Стоп-лист
< имитоприставка на действующем ключе доставки,
количество записей,
МОП дата блокировки,
:
:
МОП дата блокировки >
#3.ТД
< имитоприставка на действующем ключе доставки,
НТД,
дата изготовления ТД,
МОП(1), КОП(1),
МОП(2), КОП(2),
:
:
МОП(n), КОП(n) >
#4. ДТД
< имитоприставка на действующем ключе доставки,
НТД (номер соответствующей ТД),
дата изготовления ДТД,
МОП(n+1),КОП(n+1)
МОП(n+2), КОП(n+2)
:
:
МОП(N), КОП(N) >
#5. Новый резервный ключ доставки Kr(j)
< имитоприставка на ключе Kb(j),
Kr(j)>

П12. Формирование файла сообщений

Файл сообщений, передаваемых из АРМ КА на АРМ-Р, может содержать следующие типы сообщений:

С5. Регистрация оператора (операторов):
< дата, НА, INTMdQ), INTM(il), INTM(i2),... > .
С6. Извещение о блокировке:
< дата, НА, INTMd(j), INTM(i1), INTM(i2),... >
С11. Подключение в СКЦПД нового АРМ КА:
< дата, INTMd(j) >

П13. Изготовление ТД

ТД изготавливается РЦ на основании:
— прежнего актуального состояния ТД;
— прежнего актуального состояния ДТД;
— данных о блокировании операторов.
В режиме «Изготовить ТД» система:
— загружает актуальные ТД, ДТД и Стоп-лист;
— кодирует их на ключе Ка и переносит в архив;
— добавляет в ТД новые МОП, КОП из ДТД;
— исключает данные блокированных по Стоп-листу МОП;
— устанавливает дату (текущее время);
— устанавливает очередной номер ТД;
— с использованием ДСЧ заменяет в ТД все оставшиеся там МОП и КОП;
— кодирует новую ТД с использованием действующих ключей доставки и направляет ее в рассылку.

П14. Загрузка новой ТД на АРМ КА

1. Определяется, поступил ли файл загрузки.
2. Файл загружается в А-СБ/КА, декодируется с использованием ключа Kb(j).
3. Проверяется корректность имитоприставки блока #2 файла загрузки на ключе Kb(j):
3.1. Если она корректна, то далее выполняется п.4.
3.2. Если она некорректна, то файл вновь загружается в А-СБ/КА, декодируется с использованием ключа Kr(j)
3.3. Проверяется корректность имитоприставки блока #1 файла доставки на ключе Kr(j):
3.3.1. Если она корректна на ключе Kr(j), то этот ключ переносится в TMd(j) на место ключа Kb(j) и далее выполняется п.4.
3.3.2. Если она некорректна на ключе Kr(j), то А-СБ/КА выдает сообщение об искажении файла доставки.
4. Полученная ТД загружается в УПЗУ взамен предпоследней из действовавших ранее таблиц, одновременно стираются ДТД и Стоп-лист, относящиеся к замещаемой ТД.

П15. Подключение в СКЦПД нового АРМ КА

Содержит три фазы:
— подготовку заявки из АРМ КА на подключение;
— регистрацию АРМ КА на АРМ-Р;
— загрузку в АРМ КА актуальных контрольных данных.

П 15-1. Заявка из АРМ КА < дата, INTMd(j) >
П15-2. Регистрация на АРМ-Р
1. Номер INTMd(j) помечается как задействованный

2. Подготавливаются к отправке:
— файл конфигурации;
— файл загрузки для последнего комплекта (ТД, ДТД, Стоп-лист);
— файл загрузки для предпоследнего комплекта (ТД, ДТД, Стоп-лист).
3. Подготовленные файлы кодируются с использованием действующего ключа доставки и отправляются на АРМ КА.

П15-3. Загрузка АРМ КА

Загрузка актуальных контрольных данных должна производиться автоматически после занесения полученных файлов в определенные директории и перезагрузки компьютера.

П16. Переход на резервные ключи доставки (передачи данных)

Файлы запуска и файлы загрузки передаются на АРМ КА в закодированном виде вместе с имитоприставкой, вычисленной на используемом ключе доставки. После принятия решения о переходе всей системы на резервные ключи доставки на АРМ-Р производится загрузка в А-СБ/КА массива {Kr(j), j=1...M} вместо {Kb(j), j=1...M} с сохраненного внешнего носителя.

Вырабатываются и рассылаются на все А-СБ/КА (блок #5 файла загрузки) новые значения резервных ключей доставки.

3. ПРИМЕНЕНИЕ КОДОВ АУТЕНТИФИКАЦИИ В ПОДСИСТЕМАХ ТЕХНОЛОГИЧЕСКОЙ ЗАЩИТЫ ИНФОРМАЦИИ

3.1. Принципы построения подсистемы технологической защиты

Подсистема технологической защиты состоит из:
1) АРМ ключей;
2) АРМ персонализации;
3) АРМ администрирования и разбора конфликтов;
4) Серверов кода аутентификации;
5) Средств защиты от несанкционированного доступа к АРМ пользователя АС.

3.1.1. АРМ ключей

Предназначен для выработки и хранения и передачи АРМ-П ключей доставки. Ключи доставки используются при передаче таблиц КА серверам КА. АРМ-К является отправной точкой ключевой системы. Он вырабатывает собственный архивный ключ, на котором закрывается вся его конфиденциальная информация, хранимая на жестком диске.

АРМ-К представляет собой ПЭВМ с установленным контроллером «Аккорд СБ», который осуществляет защиту от несанкционированного доступа, и специальным ПО. Передача ключей на АРМ-П осуществляется при помощи устройств Touch Memory. «Аккорд СБ» используется и как СЗИ НСД «Аккорд 4++» (АМДЗ).

АРМ-К формирует архивный ключ АРМ-АР, таблицу ключей доставки СКА, и закрывает эту таблицу на архивном ключе АРМ-АР. Полученная информация передается АРМ-АР.

3.1.2. АРМ персонализации

Представляет собой ПЭВМ с установленным специальным ПО. Предназначен для инициализации контроллеров «Аккорд СБ» ключами доставки и номерами, употребляемыми в ПСТЗ как уникальные.

Персонализация контроллеров «Аккорд СБ» происходит в технологическом режиме. Данные для персонализации передаются с АРМ-К при помощи устройств Touch Memory. Никакой конфиденциальной информации на АРМ-П не хранится и на шине ПЭВМ не появляется. Загрузка данных производится самим контроллером «Аккорд СБ» под управлением программы, загружаемой из РС.

Контроллер «Аккорд СБ», устанавливаемый на АРМ-АР инициализируется архивным ключом АРМ-АР. Этот ключ используется для закрытия конфиденциальной информации АРМ-АР, а также для принятия ключей доставки от АРМ-К.

3.1.3. АРМ администрирования и разбора конфликтов

Представляет собой ПЭВМ с установленным контроллером «Аккорд СБ» и специальным программным обеспечением. «Аккорд СБ» служит для защиты от несанкционированного доступа и выполнения функций АРМ-АР.

АРМ-АР генерирует таблицы кодов аутентификации и вырабатывает на основании них файлы загрузки для каждого из СКА. Файлы загрузки закрываются с использованием ключей доставки. На АРМ-АР хранится полная таблица кодов аутентификации, а также архив таких таблиц. Вся информация закрывается с использованием архивного ключа АРМ-АР.

3.1.4. Сервер кода аутентификации

Представляет собой ПЭВМ с установленным контроллером «Аккорд СБ» и специальным программным обеспечением. «Аккорд СБ» служит для защиты от несанкционированного доступа и выполнения специализированных функций СКА.

Предназначен для выработки и проверки кодов аутентификации по запросам сервера информационной обработки. Обеспечивает выполнение этих функций через API, описанный ниже. Получает от АРМ-АР таблицы кодов аутентификации, закрытые на персональном ключе доставки этого сервера.

3.1.5. СЗИ НСД к АРМ пользователя АС

СЗИ НСД к АРМ пользователя АС представляют собой комплекс «Аккорд АМДЗ», обеспечивающий идентификацию/аутентификацию лиц, допущенных к работе с АРМ АС, контроль целостности аппаратных и программных ресурсов АРМ.

3.2. Функциональная схема подсистемы технологической защиты

3.2.1. Взаимодействие СКА и СИО

Работа подсистемы технологической защиты в части кода аутентификации основывается на взаимодействии сервера информационной обработки(-СИО) и сервера кода аутентификации (Рис. 4.2).

Основными функциями подсистемы являются процедуры выработки и проверки кодов аутентификации. При этом каждой технологической операции ставится в соответствие свой код. Документ, проходя обработку в АС, заверяется кодом аутентификации на каждом этапе (рис. 4.3).

1. Простановка КА1 на ЭлД.
2. Проверка и снятие ЭЦП.
3. Обработка на СИО.
4. Простановка КА2 на ЭлД.
5. Передача на ЭлД на СИО.
6. Простановка КА3 на ЭлД.
7. Проверка КА1 ЭлД.
8. Проверка КА2 и снятие КА1 и КА2.

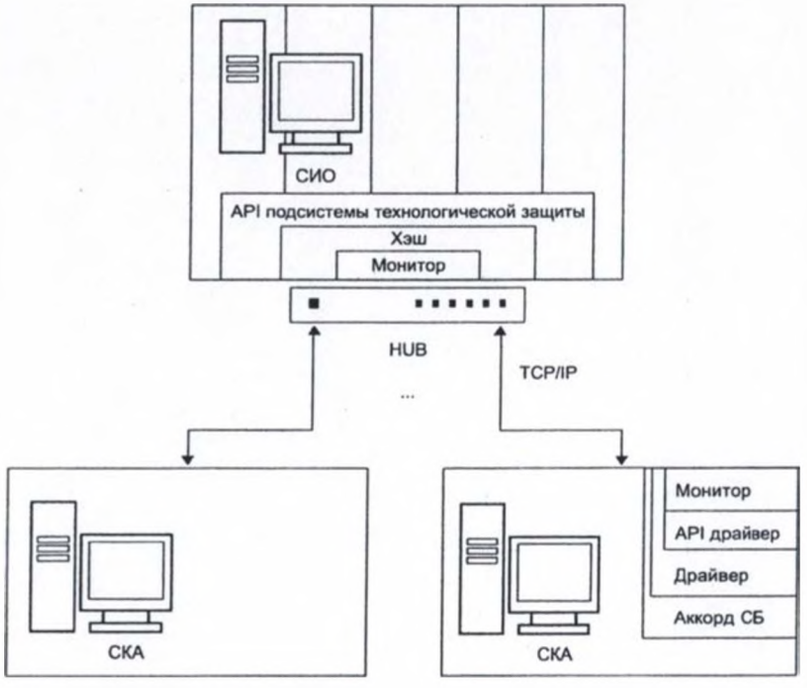


Рисунок 4.2. Схема взаимодействия подсистемы технологической защиты со смежными подсистемами

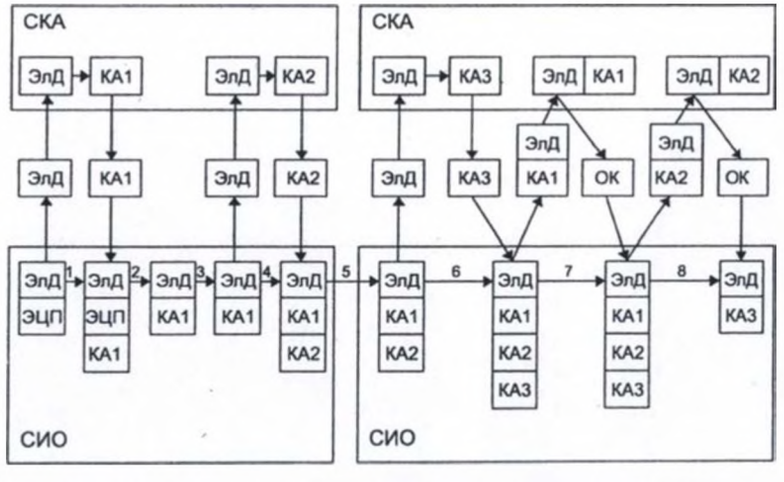


Рисунок 4.3. Схема прохождения электронного документа через СИО

Подобная схема обработки электронного документа может использоваться при прохождении любого ЭлД через сервера информационной обработки. На каждом из этапов архивной подсистеме должен передаваться ЭлД и его КА. При выполнении этого условия есть возможность произвести отложенный разбор ситуации, при которой информация была искажена, и выяснения на каком именно этапе (технологической операции) это произошло.

Система кодов аутентификации предполагает возможность проверки на СКА, установленном на СИО, любого КА, выработанного на СКА этого или другого СИО.

3.2.2. Схема вычисления кода аутентификации

Код аутентификации вырабатывается в два основных этапа. На первом этапе вычисляется хэш-функция документа. Этот этап может проводиться как на СИО, так и на СКА контроллером «Аккорд СБ».

Второй этап выполняется контроллером «Аккорд СБ» таким образом:

1) к результату хэш-функции данных добавляется конфиденциальный «код хэширования», выбираемый из специальной таблицы кодов аутентификации;

2) от полученной конкатенации вычисляется хэш-функция;

3) к результату хэш-функции добавляются номер таблицы кодов аутентификации, номер кода в таблице, который получают по формуле:

$$№КА = №СКА*16 + №Операции.$$

Хэш-функция вычисляется по ГОСТ Р 34.11-94. При вычислении хэш-функции документа и при вычислении хэш-функции для получения кода аутентификации начальный вектор должен быть нулевым. От полученного результата для получения кода аутентификации используются первые 12 байт (Рис. 4.4).

Таблица 4.2. Вид кода аутентификации

Смещение	Длина	Пояснение
0	2 байта	№ таблицы КА
2	2 байта	№ кода в таблице
4	12 байт	Хэш

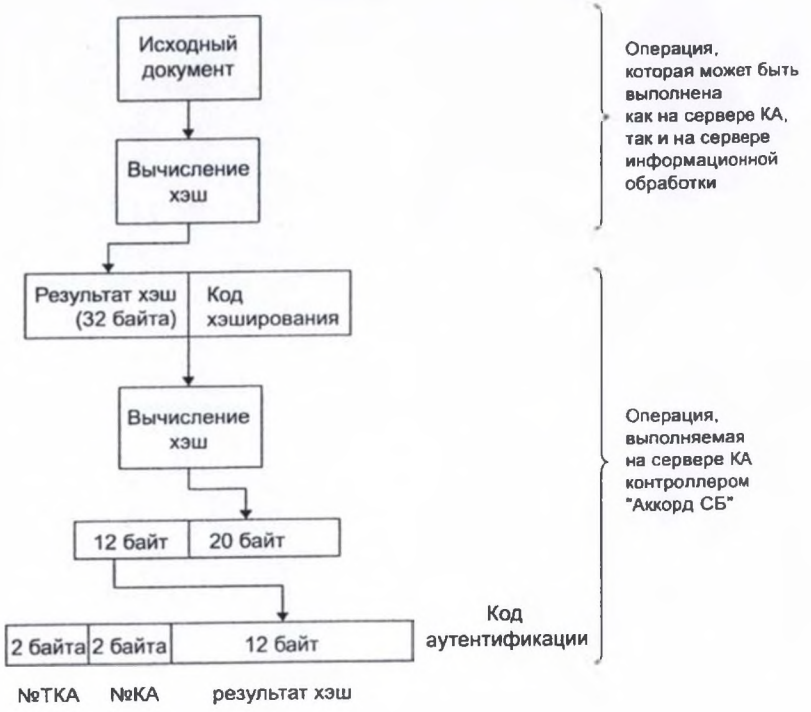


Рисунок 4.4. Схема вычисления кода аутентификации

3.3. Описание ключевой системы

Секретными элементами работающей подсистемы кода аутентификации являются хранящиеся в недоступной с шины ПК памяти контроллера «Аккорд СБ» коды хэширования, применяемые при вычислении КА. Для модифицирования таблиц этих кодов применяются ключи доставки. Принципиальная схема распределения ключей изображена на рисунке 4.5.

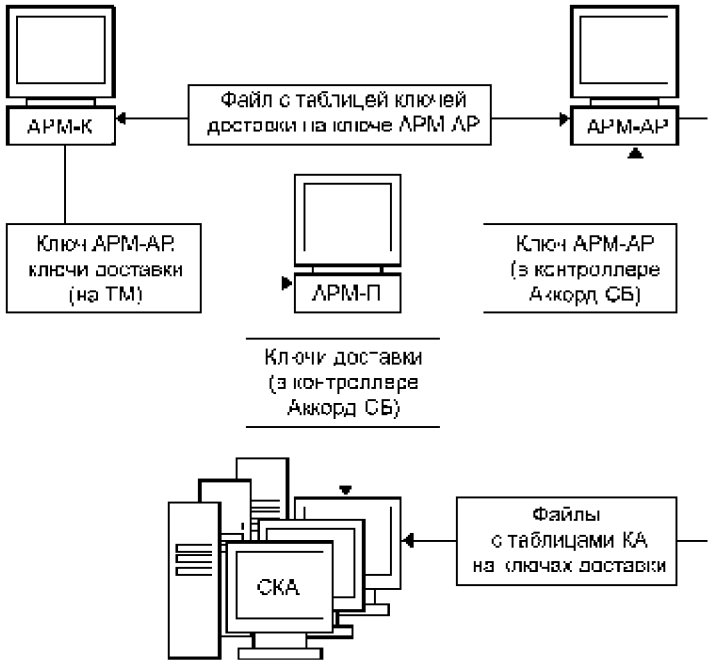


Рисунок 4.5. Принципиальная схема распределения ключей ПСТЗ

Ключи доставки применяются при обмене информацией между АРМ-АР и СКА. Схема обмена построена по принципу «звезда». АРМ-АР хранит полную таблицу ключей доставки всех СКА (внутри контроллера «Аккорд СБ»). Каждый СКА только пару ключей: базовый (Кб) и резервный (Кр). Базовые ключи применяются при обмене актуальными данными. При необходимости можно перейти на резервные. Тогда резервные становятся базовыми, а новые резервные присылаются по надежному каналу.

Хранение таблиц КА на АРМ-АР осуществляется на жестком диске, закрытыми на архивном ключе АРМ-АР.

3.4. Решения по взаимосвязи со смежными системами

Для взаимосвязи с внешними точки зрения подсистемы технологической защиты системами предоставляется API, обозначенный на рисунке 4.2

как «API подсистемы технологической защиты». Он представляет собой набор функций, к которым могут обращаться смежные подсистемы.

3.4.1. Взаимодействие с функциональной подсистемой

Вычисление кода аутентификации

Входными данными этой процедуры являются:

— электронный документ, КА для которого необходимо вычислить;

— номер операции информационной обработки (номер КА).

На выходе вызывающая подсистема получает КА или сообщение о невозможности его получения (код ошибки).

Проверка кода аутентификации

Код аутентификации содержит в себе номер сервера КА и номер информационной операции, для которой он был выработан. Исходя из этого, входными данными этой процедуры будут:

— электронный документ, КА которого необходимо проверить;

— КА соответствующий документу.

На выходе вызывающая подсистема получает код ошибки (нулевой при успешной проверке).

Получение уровня загруженности подсистемы КА

Для проведения мероприятий по тестированию подсистемы, а также для контроля за ее работой необходимой является функция получения уровня загруженности. Определение уровня загруженности основывается на оценке числа вызовов вычислительных функций с учетом их весовых коэффициентов. Результатом является процент загруженности от максимального.

В связи с возможностью прерванного вычисления/проверки КА аутентификации необходимо существование следующих функций:

Начать вычисление/проверку КА с сохранением контекста.

Входные данные:

— первая часть электронного документа.

На выходе:

— контекст, представляющий собой результат хэш-функции части ЭлД, длина которой кратна 32 байтам плюс оставшаяся часть ЭлД, длиной менее 32 байт. Для продолжения вычисления хэш-функции необходимо сохранить также текущую контрольную сумму хэширования (32 байта) и длину обработанных блоков (4 байта). Таким образом, общая длина контекста составит 100 байт.

Продолжить вычисление/проверку КА с сохранением контекста.
Входные данные:
— предыдущий контекст;
— очередная часть ЭлД.
На выходе:
— контекст, представляющий собой результат хэш-функции всех предыдущих частей.

Завершить вычисление КА по сохраненному контексту.
Входные данные:
— предыдущий контекст;
— последняя часть ЭлД;
— номер операции.
На выходе:
— код аутентификации или код ошибки.

Завершить проверку КА по сохраненному контексту.
Входные данные:
— предыдущий контекст;
— последняя часть ЭлД.
На выходе:
— код ошибки.

3.4.2. Взаимодействие с архивной подсистемой

Одной из важнейших задач подсистемы КА является отложенный разбор конфликтов. Выполнение ее выдвигает требования к архивной подсистеме. Каждая операция простановки КА должна фиксироваться. В архиве должен сохраняться документ и все его коды аутентификации. При возникновении ситуации, когда необходимо заблокировать СИО, факт блокировки должен регистрироваться архивной подсистемой. Совокупность этих данных должна при необходимости передаваться АРМ-АР подсистемы КА, для разбора конфликтных ситуаций.

3.5. Решения по режимам функционирования

3.5.1. Первоначальная установка подсистемы

На АРМ-К вырабатываются ключи доставки, которые используются впоследствии для обмена данными между АРМ-АР и серверами КА. Для каждого сервера КА ключ доставки записывается в Touch Memory. ТМ переносится на АРМ- П, где данные из нее записываются в контроллеры «Аккорд СБ» серверов КА и АРМ-АР.

На АРМ-АР вырабатываются коды хэширования Ki, j, где i — номер операции, j — номер платы сервера КА. Все Ki, j объединяются в таблицу КА, которая сохраняется в зашифрованном виде на жестком диске АРМ-КА на его архивном ключе.

Таблица кодов аутентификации, закрытая на ключах доставки, передается контроллерам «Аккорд СБ» серверов КА и записывается в энергонезависимую память контроллеров. После этого контроллеры устанавливаются в ПЭВМ, на ПЭВМ устанавливается ПО, обеспечивающее выполнение функций СКА.

3.5.2. Штатный режим

Сервера КА находятся в режиме ожидания запроса от серверов доступа (информационной обработки). Поступающие запросы через API (см. рис. 4.2) передаются на сервера КА, где происходит выработка и проверка кодов аутентификации. АРМ-К и АРМ-АР не участвуют в этом процессе. АРМ-П может быть демонтирован.

3.5.3. Плановая смена таблиц кодов аутентификации

Плановая смена таблиц кодов аутентификации проводится во время остановки функциональной части системы АС. Это связано со сложностью синхронизации изменений таких таблиц на ходу.

АРМ-АР заблаговременно вырабатывает новые списки кодов хэширования, закрывает их на ключах доставки серверов КА, формируя файлы загрузки «Аккорд СБ». Файлы загрузки передаются непосредственно на сервера КА, где загружаются администратором безопасности с помощью специального ПО в контроллеры «Аккорд СБ». После загрузки всех контроллеров система может быть вновь активизирована.

3.5.4. Блокировка серверов

Блокировка СИО в рамках подсистемы КА означает блокировку всех связанных с ним серверов КА. Блокировка проводится с АРМ-АР. АРМ-АР вырабатывает стоп-листы для всех оставшихся серверов КА.

Изменения в таблицу КА на ходу вносятся следующим образом. Работа всех серверов КА, кроме одного на данном СИО приостанавливается администратором безопасности. На приостановленных СКА производится загрузка

актуальных данных. После этого приостанавливается работа оставшегося СКА, с постепенным введением в работу ранее приостановленных. После загрузки стоп-листа последний СКА связки вводится в эксплуатацию.

3.5.5. Отложенный разбор конфликтов

Архивная подсистема передает на АРМ-АР данные об изменениях (блокировках) в составе СИО за все время жизни таблицы КА, при которой произошел конфликт, данные о документе и всех КА, которые вырабатывались для него. На основании этих данных и данных таблицы КА, хранимой на АРМ-АР можно провести анализ конфликта. Разбор конфликта производится администратором безопасности на АРМ-АР при помощи специализированного ПО.

3.6. Состав функций СКА

1. Функции вычисления и проверки КА:
— вычисление кода аутентификации документа;
— проверка кода аутентификации документа;
— начать вычисление/проверку КА с сохранением контекста;
— продолжить вычисление/проверку КА с сохранением контекста;
— завершить вычисление КА по сохраненному контексту;
— завершить проверку КА по сохраненному контексту;
2. Получение уровня загруженности СКА;
3. Функции тестирования.

На техническом уровне:

— контроллер КА должен обеспечивать выполнение внутренней процедуры тестирования на основе вычисления хэш-функции от стандартного примера. Системному ПО сервера КА выдается ответ о правильности вычислений.

На системном уровне (Рис. 4.6):

— выполнение процедуры тестирования на основании примера, получаемого от системного ПО сервера КА. Системное ПО сервера КА засылает контроллеру данные для вычисления хэш-функции, заранее зная правильный результат. На основании полученного значения хэш-функции делается вывод о корректности работы контроллера;

— получение и проверка контрольных данных о целостности внутреннего ПО и таблиц достоверности. Системное ПО дает котроллеру команду на вычисление хэш-функции от внутренней программы и таблицы, а так же на выдачу версии ПО и ТД. Данные о правильных значениях хэш-функции от этих данных получены при их загрузке в контроллер;

— при низком уровне загруженности сервер КА может тестировать правильность вычисления хэш-функции сервером доступа. После получения результата хэш-функции от сервера доступа, сервер КА может потребовать пересылки оригинала документа.

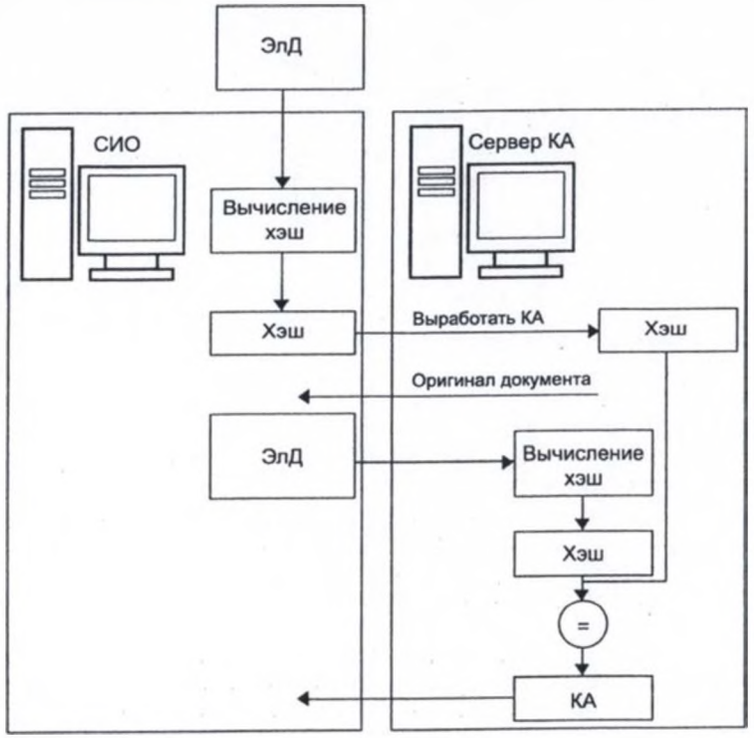


Рисунок 4.6. Функции тестирования на системном уровне

На прикладном уровне:

— интерфейс между сервером информационной обработки и сервером КА должен обеспечивать получение сервером доступа уровня загруженности сервера КА. При низком уровне загруженности проведение процедуры тестирования вида: засылка тестового документа для получения КА, с заранее известным результатом.

3.7. Работа СКА

Сервер кода аутентификации (СКА) — является одним из элементов подсистемы технологической защиты электронных документов, используется во взаимодействии с сервером информационной обработки (СИО).

СКА предназначен для выработки и проверки кодов аутентификации (КА) входящих, обрабатываемых и отправляемых в АС электронных документов.

В состав СКА входит:

— автономный IBM-совместимый компьютер под управлением Windows 95/98, Windows NT, соединенный каналом FastEthernet с СИО;

— программно-аппаратный комплекс «Аккорд СБ», устанавливаемый в этот автономный компьютер;

— специальное программное обеспечение СКА, устанавливаемое в этот автономный компьютер;

— специальное программное обеспечение СКА, устанавливаемое на СИО.

Вначале устанавливается программное обеспечение СКА на СИО и автономном компьютере, после чего на автономном компьютере устанавливается его аппаратное обеспечение — контроллер «Аккорд СБ», прошедший процедуру персонализации в Центральном органе управления подсистемой технологической защиты электронных банковских документов.

Установленный комплекс СКА после включения автономного компьютера выводит на экран монитора этого компьютера загрузочное меню, состоящее из двух пунктов:

1) *«Основной режим работы»*;

2) *«Загрузка новой таблицы достоверности»*.

При первоначальном запуске СКА требуется выбрать режим «Загрузка новой таблицы достоверности». Этот же режим потребуется выбрать в случае плановой замены таблицы достоверности или при установке на СКА новой таблицы достоверности при ликвидации последствий компрометации ранее установленного на СКА сопроцессора «Аккорд СБ» и замене его на нескомпрометированный. Работа в режиме «Загрузка новой таблицы достоверности» является исключительной обязанностью администратора СКА. В обязанности оператора СКА входят только контроль отсутствия НСД к СКА и повторное включение СКА без изменения таблиц достоверности (после технологических отключений СКА).

В нормальном рабочем режиме автономный компьютер СКА не подлежит выключению или отключению от СИО. Однако, в случае остановки работы СИО, с которым он взаимодействует, и последующем возобновлении работы этих хост-машин без компрометации автономного компьютера СКА он запускается в «Основном режиме работы».

Установка программного обеспечения СКА на СИО производится совместно с установкой программной системы на этих хост-машинах (входит в его состав).

Персонализация контроллеров «Аккорд СБ» происходит в технологическом режиме. Данные для персонализации передаются с АРМ-К при помощи устройств Touch Memory. Никакой конфиденциальной информации на АРМ-П не хранится и на шине РС не появляется. Загрузка данных производится самим контроллером «Аккорд СБ» под управлением программы, загружаемой из ПЭВМ.

3.7.1. Меры безопасности при проведении персонализации

Меры безопасности при проведении персонализации заключаются в осуществлении мероприятий по обеспечению организационной и физической защиты сопроцессоров «Аккорд СБ» (АРМ-П) и ТМ-идентификаторов с ключами.

Для эффективного применения комплекса и поддержания уровня защищенности ПЭВМ (АС) необходимы:

— физическая охрана ПЭВМ и ее средств, в т.ч. обеспечение мер по неизвлечению контроллера комплекса;

— периодическое тестирование средств защиты комплекса;

— разработка и ведение учетной и объектовой документации (инструкция администратора, инструкций пользователей, журнал учета идентификаторов и отчуждаемых носителей пользователей и др.). Все разработанные учетные и объектовые документы должны быть согласованы, утверждены у руководства и доведены до сотрудников (пользователей).

Коды аутентификации (КА) электронных документов (ЭлД) в подсистеме технологической защиты информации формируются и проверяются на серверах кода аутентификации (СКА) с помощью ключевых таблиц (таблиц достоверности), хранящихся во внутренней памяти установленных в СКА сопроцессоров «Аккорд СБ». Формирование и проверка КА выполняется серверами кода аутентификации по запросам сервера информационной обработки (СИО). Таблицы достоверности, закрытые на ключах доставки, доставляются на СКА и загружаются во внутреннюю память сопроцессоров, где и происходит их раскрытие. Ключи доставки загружаются в сопроцессоры на начальном этапе в процессе их персонализации.

Ключи доставки в подсистеме КА формируются и регистрируются на специализированном автоматизированном рабочем месте — АРМ-К.

Выработка ключей доставки осуществляется Центральным органом управления подсистемой КА при персонализации сопроцессоров «Аккорд СБ». Индивидуально подготовленные для каждого сопроцессора ключи доставки вырабатываются на АРМ-К и записываются на устройство Touch-memory

(TMd), с помощью которого они переносятся в сопроцессоры «Аккорд СБ» с помощью программного обеспечения АРМ-П. Затем эти сопроцессоры доставляются через надежные транспортные каналы на места эксплуатации. На TMd ключи пишутся *в открытом виде*.

В системе предусмотрено наличие основного и резервного ключей доставки. Вся совокупность выработанных основных ключей доставки хранится на АРМ-К *под защитой Главного ключа АРМ-К* (ТМ-устройство) и переносится на дискете на АРМ-АР *под защитой архивного ключа АРМ-АР* и устанавливается в его контроллер «Аккорд СБ».

Вся совокупность выработанных резервных ключей доставки хранится администратором АРМ-К отдельно на дискете в надежном месте и используется для замены основных ключей в случае компрометации АРМ-К или АРМ-Р. Необходимость ввода их в действие определяется Администратором Центрального органа подсистемы КА.

Архивные базы ключей — основных и резервных, хранятся на магнитных носителях в зашифрованном виде на архивном ключе АРМ-К. Архивный ключ содержится во внутренней памяти контроллера «Аккорд» АРМ-К. Архивный ключ зашифрован на Главном ключе АРМ-К.

После ввода в действие резервных ключей системой обеспечивается возможность выработки нового массива резервных ключей доставки для всех сопроцессоров «Аккорд СБ». Новые резервные ключи доставляются на места эксплуатации на ГМД в закрытом виде с использованием действующего ключа доставки.

При регистрации каждому сопроцессору «Аккорд СБ» присваивается уникальный номер и задается принадлежность (организация-владелец) ключей (на строку принадлежности также распространяется требование по уникальности). Эти данные: номер и принадлежность являются основными параметрами, однозначно идентифицирующими соответствующий сопроцессор «Аккорд СБ». Ключи взаимодействия записываются в архивную базу вместе с номером сопроцессора «Аккорд СБ» и его принадлежностью.

Используемая в подсистеме КА ключевая структура представлена в таблице 4.3.

Наиболее уязвимыми носителями ключевых устройств являются ТМ-устройство и резервная дискета с Главным ключом АРМ-К, а также ТМ-устройства, используемые при персонализации сопроцессоров «Аккорд СБ». Доступ к этим устройствам должен быть подчинен правилу «двух лиц», с фиксацией использования этих носителей в рукописном журнале за подписью двух ответственных лиц.

Таблица 4.3

Назначение ключа(ей)	Область хранения	Ключ защиты
Таблицы достоверности	Сопроцессоры "Аккорд СБ"	Ключи доставки
Таблицы достоверности	Жесткий диск АРМ-АР	Архивный ключ АРМ-АР
Основные ключи доставки	Жесткий диск АРМ-К	Архивный ключ АРМ-К
Основные ключи доставки	Сопроцессоры "Аккорд СБ"	Архивный ключ АРМ-К
Основные ключи доставки	Персонализирующее ТМ-устройство	Архивный ключ АРМ-К
Резервные ключи доставки	Дискета Администратора	Архивный ключ АРМ-К
Резервные ключи доставки	Сопроцессоры "Аккорд СБ"	Архивный ключ АРМ-К
Резервные ключи доставки	Персонализирующее ТМ-устройство	Архивный ключ АРМ-К
Архивный ключ АРМ-АР	Сопроцессор "Аккорд СБ" АРМ-АР	Архивный ключ АРМ-К
Архивный ключ АРМ-АР	Жесткий диск АРМ-К	Архивный ключ АРМ-К
Архивный ключ АРМ-К	Контроллер "Аккорд" АРМ-К	Главный ключ АРМ-К
Главный ключ АРМ-К	ТМ-устройство Администратора	Главный ключ АРМ-К
Главный ключ АРМ-К	Резервная дискета Администратора	Главный ключ АРМ-К

Утрата ТМ-устройства персонализации влечет необходимость удаления ключей доставки соответствующего сопроцессора из базы зарегистрированных ключей и выработку для него новых ключей доставки в соответствии «Инст-

рукцией по порядку заказа, смены и уничтожения ключей».

При подозрении на компрометацию ТМ-устройства его администратор докладывает о случившемся администратору подсистемы и по его указанию:

— осуществляет переход на резервные ключи доставки;

— вырабатывает новые таблицы достоверности с использованием этих резервных ключей;

— рассылает подготовленные таблицы достоверности на все СКА для замены ранее действовавших.

Утрата ТМ-устройства или резервной дискеты с Главным ключом влечет полную переинсталляцию ключевой системы, однако в этот переходный период остается возможность работы на резервных ключах доставки при сохранности базы резервных ключей от компрометации.

Остальные носители не создают опасности несанкционированного получения хранящейся на них ключевой информации, так как используется ее закрытие на внешних ключах.

Схема управления используемыми в подсистеме КА ключами представлена в таблице 4.4.

3.7.2. Описание прикладного программного интерфейса

Библиотечные функции для вычисления КА.

Потоковые КА

int InitHash(void *hashbuf);

void GetHash(void *hashbuf , char *databuf, long datalength)

int GetHashKA(short int *idclient, long idlength, void *hashbuf, void *kabuf, long *kalen);

int TestHashKA(void *idclient, long idlength, void *hashbuf, void *kabuf);

КА над единым буфером данных

int prostanovka_KA(short int *idclient, char *databuf, long datalen, char *kabuf, long *kalen);

int proverka_KA(short int *idclient, long idlength, char *databuf, long datalen, char *kabuf);

Максимальная длина КА равна значению переменной KA_SIZE. Точная длина КА возвращается функциями GetHashKA, prostanovka_KA в переменной kalen.

Таблица 4.4

Назначение ключа(ей)	Где вырабатываются	Куда переносятся
Таблицы достоверности	АРМ-АР	Сопроцессоры "Аккорд-СБ"
Таблицы достоверности	АРМ-АР	Жесткий диск АРМ-АР
Основные ключи доставки	АРМ-К	Жесткий диск АРМ-К
Основные ключи доставки	АРМ-К	Сопроцессоры "Аккорд-СБ"
Основные ключи доставки	АРМ-К	Персонализирующее ТМ-устройство
Резервные ключи доставки	АРМ-К	Дискета Администратора
Резервные ключи доставки	АРМ-К	Сопроцессоры "Аккорд-СБ"
Резервные ключи доставки	АРМ-К	Персонализирующее ТМ-устройство
Архивный ключ АРМ-АР	АРМ-К	Сопроцессор "Аккорд-СБ" АРМ-АР
Архивный ключ АРМ-АР	АРМ-К	Жесткий диск АРМ-К
Архивный ключ АРМ-К	АРМ-К	Контроллер "Аккорд" АРМ-К
Главный ключ АРМ-К	АРМ-К	Резервная дискета Администратора

Инициализация потока данных КА

Данная функция должна вызываться каждый раз, когда начинаются операции по вычислению КА над потоком данных. Для данного потока данных функция должна быть вызвана один раз.

int OpenHash(void *hashbuf).

Входные параметры:

hashbuf — указатель на неинициализированный хэшбуфер, размер бу-

фера равен 1024 байт.

Выходные параметры:

hashbuf — указатель на инициализированный хэшбуфер, имеет размер от 1 до 1024 байт.

Возвращаемое значение:

Функция возвращает длину инициализированного hashbuf.

Если возвращено отрицательное число, тогда возникла ошибка инициализации и дальнейшая работа невозможна.

Вычислить хэш

Данная функция вычисляет хэш над текущим потоком данных. Поток данных должен быть открыт функцией OpenHash. Функция может вызываться несколько раз до исчерпания потока данных.

void GetHash(void *hashbuf, char *databuf, long datalength)

Входные параметры:

hashbuf — инициализированный функцией InitHash() хэшбуфер;

databuf — указатель на буфер, где лежат хэшируемые данные клиента;

datalength — длина буфера databuf.

Выходные параметры:

hashbuf — указатель на инициализированный хэшбуфер, длина буфера не меняется.

Вычислить потоковый КА

int GetHashKA(short int *idclient, long idlength, void *hashbuf, void *kabuf, long *kalen);

Входные параметры:

idclient — идентификатор клиента. Это структура, включающая в себя:

1) номер клиента — 2 байта (short int);

2) номер рабочего места — 2 байта (short int);

3) ...(что-то еще)...

По этой структуре можно однозначно определить ключ клиента

idlength — длина буфера idclient

hashbuf — указатель на инициализированный хэшбуфер, хэшбуфер был вычислен в результате одно- или многократного вызова функции GetHash().

Выходные параметры:

kabuf — указатель на буфер КА, длина должна быть не менее KA_SIZE;

kalen — указатель на точную длину буфера КА.

Функция возвращает :

КА ОК

OK 0

Клиент не зарегистрирован

ERR_USER_NOT_FOUND 1

КА неверен

ERR_BAD_KPD 2

Ошибка связи с Сервером КА

ERR_CONNECT 3

Проверить потоковый КА

int TestHashKA(void *idclient, long idlength, void *hashbuf, void *kabuf);

Входные параметры:

idclient — идентификатор клиента. Это структура, включающая в себя:

1) номер клиента — 2 байта (short int);

2) номер рабочего места — 2 байта (short int);

3) другие параметры.

По этой структуре можно однозначно определить ключ клиента.

idlength — длина буфера idclient

hashbuf — указатель на инициализированный хэшбуфер. Хэшбуфер был вычислен в результате одно- или многократного вызова функции GetHash().

kabuf — указатель на буфер КА. Данный КА был вычислен ранее и теперь проверяется. Возвращаемое значение:

Функция возвращает :

КА ОК

OK 0

Клиент не зарегистрирован

ERR_USER_NOT_FOUND 1

КА неверен

ERR_BAD_KPD 2

Ошибка связи с Сервером КА

ERR_CONNECT 3

Вычислить КА над блоком данных

int prostanovka_KA(short int *idclient, char *databuf, long datalen, char *kabuf, long *kalen)

Входные параметры:

idclient — идентификатор клиента. Это структура, включающая в себя:

1) номер клиента — 2 байта (short int);

2) номер рабочего места — 2 байта (short int);

3) другие параметры.

По этой структуре можно однозначно определить ключ клиента.

databuf — указатель на буфер, где лежат данные клиента;

datalen — длина буфера databuf;

kabuf — указатель на буфер КА (длина буфера kabuf должна быть не менее KA_SIZE байт);

kalen — указатель на длину возвращаемого буфера КА.

Возвращаемое значение:

Функция возвращает :

КА ОК

OK 0

Клиент не зарегистрирован

ERR_USER_NOT_FOUND 1

КА неверен
ERR_BAD_KPD 2
Ошибка связи с Сервером КА
ERR_CONNECT 3

Проверить КА над блоком данных

int proverka_KA(short int *idclient, long idlength, char *databuf, long datalen, char *kabuf)

Входные параметры:

idclient — идентификатор клиента. Это структура, включающая в себя:

1) номер клиента — 2 байта (short int);

2) номер рабочего места — 2 байта (short int);

3) другая информация.

По этой структуре можно однозначно определить ключ клиента.

idlength — длина буфера idclient;

databuf — указатель на буфер, где лежат данные клиента;

datalen — длина буфера databuf;

kabuf — указатель на буфер КА. Данный КА был вычислен ранее и теперь проверяется.

Функция возвращает:

КА ОК
ОК 0
Клиент не зарегистрирован
ERR_USER_NOT_FOUND 1
КА неверен
ERR_BAD_KPD 2
Ошибка связи с Сервером КА
ERR_CONNECT 3

3.7.3. Отбраковка подсистемой кода аутентификации одного или группы электронных документов при их приеме или обработке

В случаях отбраковки подсистемой кода аутентификации одного или группы электронных документов при их приеме на каком-либо СИО, администратор СКА на этом СИО докладывает о случившихся фактах в Центральный орган управления подсистемой КА; извлекает из базы всю архивную информацию, связанную с обработкой отбракованных документов документов (электронные копии ЭлД вместе с КА), и направляет ее в Центральный орган.

Центральный орган запрашивает необходимую архивную информацию от администратора СИО, отправившего данный документ, и производит разбор конфликта.

В Центральном органе разбор каждого случая отбраковки производится путем проверки соответствия кода аутентификации документу по каждой архивной записи последовательно, по ходу продвижения документа при его обработке.

При этом обязательно должен быть выделен этап, на котором впервые проявилось несоответствие текущей версии документа его коду аутентификации.

По результатам разбора администратором АРМ-АР подготавливается заключение. На основании этого заключения и решения Администратора Центрального органа управления подсистемой КА принимаются меры по устранению вскрывшихся дефектов в работе данного сервера доступа.

3.8. Штатная структура и обязанности персонала Центрального узла

3.8.1. Штатная структура центрального узла СПТЗ

АРМ-К, АРМ-П и АРМ-АР размещаются в Центральном узле управления подсистемой кода аутентификации. Соответственно, штатная структура этого органа должна состоять из:

— администратора (руководителя) Центрального узла управления подсистемой кода аутентификации;

— заместителя администратора Центрального узла управления подсистемой кода аутентификации;

– администратора АРМ-К и АРМ-П;

— оператора (операторов) АРМ-К и АРМ-П;

— администратора АРМ-АР.

3.8.2. Обязанности должностных лиц центрального узла

Администратор узла

В функциональные обязанности Администратора Центрального узла управления подсистемой кода аутентификации входит:

— руководство и управление работой персонала подсистемы кода аутентификации;

— принятие решений о плановой смене ключевых элементов, используемых в подсистеме;

— принятие решений о внеплановой смене ключевых элементов, используемых в подсистеме при подозрении на их компрометацию;

— принятие решений о подключении новых СКА к подсистеме и орга-

низация необходимых для этого мероприятий;

— организация и принятие решений по результатам разбора нештатных ситуаций, возникающих в работе подсистемы.

Заместитель администратора узла

В функциональные обязанности Заместителя администратора Центрального узла управления подсистемой кода аутентификации входит:

— выполнение функций Администратора Центрального органа при его отсутствии;

— выполнение функций Администраторов АРМ-К, АРМ-П и АРМ-АР при их отсутствии, а также по специальным поручениям Администратора Центрального узла;

— организация рассылки в Региональные Центры актуальной эксплуатационной документации и ключевых элементов;

— организация разбора нештатных ситуаций, возникающих в работе подсистемы.

Администратор АРМ-К и АРМ-П

В функциональные обязанности Администратора АРМ-К и АРМ-П входит:

— периодический контроль защитных меток на корпусе АРМ-К;

— хранение Главного ключа АРМ-К на специально выделенном ТМ-устройстве;

— включение АРМ-К и АРМ-П;

— периодическая смена Главного ключа АРМ-К;

— работа с архивным ключом АРМ-К;

— хранение дискеты с базой резервных ключей взаимодействия;

— конфигурация комплекса АРМ-К;

— периодический контроль ведения оператором рукописного журнала работы;

— выполнение обязанностей оператора при его отсутствии.

Оператор АРМ-К и АРМ-П

В функциональные обязанности оператора АРМ-К и АРМ-П входят:

— периодический контроль защитных меток на корпусе АРМ-К;

— выполнение процедуры регистрации сопроцессоров «Аккорд СБ» и выработки ключей взаимодействия;

— периодический контроль целостности базы основных ключей взаимодействия;

— ведение ключей взаимодействия;

— отражение выполненных операций в рукописном журнале работы.

Администратор АРМ-АР

В функциональные обязанности Администратора АРМ-АР входят:

— периодический контроль защитных меток на корпусе АРМ-АР;

— конфигурация комплекса АРМ-АР;

— выработка и рассылка на СКА таблиц достоверности при их плановой и внеплановой смене;

— ввод в действие резервных ключей взаимодействия;

— регистрация в подсистеме новых СКА и рассылка для них актуальных ключевых данных;

— разбор и подготовка заключений по результатам разбора нештатных ситуаций, возникающих в работе подсистемы;

— периодический контроль ведения оператором рукописного журнала работы;

— ведение рукописного журнала работы.

3.9. Штатная структура регионального узла и обязанности должностных лиц

Сервера кода аутентификации (СКА) подсистемы технологической защиты в части кода аутентификации входят в состав Серверов информационной обработки (СИО), обслуживающих АС как центра, так и Региональных Узлов. Соответственно, штатная структура Региональных Узлов должна состоять из:

— администратора СКА;

— оператора (операторов) СКА.

Администратор и операторы СКА обслуживают все СКА, размещенные на одном СИО.

Функциональные обязанности администратора СКА

В функциональные обязанности Администратора СКА входит:

— периодический контроль защитных меток на корпусе СКА;

— загрузка новой таблицы достоверности;

— замена сопроцессора «Аккорд СБ» в случае его неисправности или при подозрении на его компрометацию;

— предоставление в Центральный орган управления подсистемой кода аутентификации всех запрашиваемых архивных данных по его запросу;

— периодический контроль ведения оператором рукописного журнала работы;

— выполнение обязанностей оператора при его отсутствии.

Функциональные обязанности оператора СКА

В функциональные обязанности оператора СКА входит:

— периодический контроль защитных меток на корпусе СКА;

— включение СКА в «Основном режиме работы»;

— ведение рукописного журнала работы.

3.10. Основные обязанности администратора безопасности АС

3.10.1. Общие обязанности администратора безопасности

Администратор безопасности АРМ АС обязан:

— принять ответственность за поддержание уровня защиты АРМ на себя;

— способствовать функционированию АРМ АС согласно принятому регламенту;

— содействовать внедрению унифицированных систем документооборота;

— координировать работу пользователей АРМ по обеспечению ЕДИНОЙ ТЕХНИЧЕСКОЙ ПОЛИТИКИ по защите информации;

— обладать правами на проведение всех проверок и контрольных испытаний на конкретном АРМ АС и умело применять их на практике;

— быть в вопросах защиты АРМ АС профессионально подготовленным и четко знать требования НД по защите;

— в совершенстве знать применяемые информационные технологии;

— участвовать в контрольных испытаниях и проверках АРМ АС;

— принимать участие в анализе защищенности АРМ АС;

— позаботиться о доведении до всех пользователей АРМ требований в части защиты данного АРМ от НСД к информации;

— позаботиться о том, чтобы каждый пользователь АС понимал свою ответственность за поддержание уровня защиты АС и правомерную работу на АРМ АС;

— предусмотреть средства резервирования и дублирования на случай нештатных ситуаций на АРМ АС;

— оценивать возможности внесения изменений в состав АРМ АС с учетом требований НД по защите;

— вносить предложения по повышению качества и уровня защиты АС и АРМ АС;

— анализировать учетные данные Журнала учета работы АРМ АС;

— исходить из того, что средства защиты АРМ АС входят в состав данного АРМ и являются его неотъемлемой частью;

— знать ЭД на средства защиты АРМ АС и руководствоваться этой документацией в практической работе;

— решать другие вопросы в области защиты информации и в области информатизации.

3.10.2. Обязанности администратора безопасности АС по установке средств защиты

Администратор безопасности на этапе установки и адаптации средств защиты АРМ АС от НСД к информации обязан:

— изучить общие организационно-распорядительные документы в части обеспечения и поддержания необходимо класса защищенности АС и ее подсистем;

— изучить эксплуатационную документацию (ЭД) на средства защиты АРМ АС;

— выполнить начальную установку и адаптацию средств защиты согласно ЭД на эти средства защиты;

— организовать и обеспечить ведение служебной информации СЗИ НСД (генерацию паролей, ключей, сопровождение правил разграничения доступа);

— организовать учет всех атрибутов пользователей АС;

— своевременно заблокировать все функции, которые не должны быть доступны данному пользователю АРМ АС;

— завести эталонные копии ПС АРМ АС и других резервируемых средств, предусмотренных на АС;

— сделать соответствующие отметки в формуляре на АРМ АС;

— уточнить правила ведения Журнала учета работы АРМ АС;

— провести инструктаж обслуживающего персонала и пользователей АРМ АС по правилам работы с используемыми средствами и системами защиты;

— при нарушении регламента применения АС (ее подсистемы) и других противоправных деяниях в отношении АС (ее подсистем) необходимо немедленно доложить руководству по подчиненности и принять соответствующие меры, указанные в Плане мероприятий на случай нештатных ситуаций в АС.

3.10.3. Обязанности администратора безопасности при применении защищенных АРМ АС

При применении АРМ АС в промышленной эксплуатации администратор безопасности обязан:

— обеспечить функционирование и поддержку работоспособности средств и систем защиты АРМ АС от НСД к информации в пределах возложенных обязанностей;

— обеспечить доступ к защищаемым ресурсам АРМ АС санкционированным пользователям согласно их прав;

— доводить до пользователей их полномочия на АРМ АС;

— проводить контроль за работой пользователей на АРМ АС, выявлять и

пресекать попытки противоправных деяний в отношении АРМ АС;

— поддерживать функционирование средств и систем защиты АРМ АС;

— формировать и распределять реквизиты (атрибуты) пользователей, определяемые их полномочиями;

— иметь утвержденные правила разграничения доступа (матрицу доступа) и вносить в них соответствующие корректировки;

— анализировать содержание Системных (регистрационных) журналов;

— уточнять порядок обработки защищаемой информации;

— уточнять условия эксплуатации АРМ АС;

— периодически убеждаться в неизменности общесистемной программной среды АРМ АС;

— контролировать ведение формуляра на АРМ АС;

— разработать и утвердить План мероприятий на случай нештатных ситуаций на АРМ АС;

— принимать участие в проведении служебных расследований фактов противоправных деяний в отношении АРМ АС или при применении АРМ АС, как средства противоправного деяния.

3.11. Права администратора безопасности АС

Администратор безопасности АРМ АС имеет право:

— обращаться к руководителям АС в части обеспечения и совершенствования защиты АРМ АС от НСД к информации;

— выдавать ЗАКЛЮЧЕНИЕ об устранении дефектов и недоработок, влияющих на УРОВЕНЬ защиты АРМ АС;

— вносить предложения о целесообразности приобретения новых типов ТС, ПС и средств защиты АРМ АС;

— контролировать хранение РЕЗЕРВНЫХ копий информационных ресурсов, ПС и баз данных АРМ АС;

— проводить проверки защищенности АРМ АС.

На администратора безопасности АРМ АС возлагается персональная ответственность за поддержание работоспособности защищенного АРМ АС, средств и систем защиты, за качество проводимых им работ по обеспечению защиты АРМ АС.

Администратор безопасности АРМ АС несет ответственность по действующему законодательству за разглашение защищаемой информации (сведений), ставших известными ему в соответствии с родом работы.

3.12. Действия администратора при компрометации СКА

Любой случай несанкционированного вскрытия корпуса автономного компьютера СКА следует считать компрометацией записанных в нем ключей доставки и таблиц достоверности.

В этом случае администратор СКА производит следующие действия:

— выключает компьютеры всех СКА;

— извещает Центральный орган управления подсистемой КА о случившемся происшествии и переходе на новые (резервные) таблицы достоверности;

— извлекает сопроцессор «Аккорд СБ» из скомпрометированных СКА и заменяет их на резервные;

— на всех СКА производит замену таблиц достоверности, пользуясь новыми (резервными) загрузочными дискетами, изготовленными в Центральном органе персонально для каждого сопроцессора «Аккорд СБ»;

— переводит все СКА в рабочий режим;

— делает соответствующую запись в рукописном журнале отражения работы;

— отправляет сопроцессоры «Аккорд СБ» из скомпрометированных СКА в Центральный орган управления подсистемой КА для повторной персонализации.

Администратор обязан надежно хранить выданный ему персональный ТМ-идентификатор, а в случае его утраты или временной потери контроля за его сохранностью немедленно произвести перерегистрацию своего нового ТМ-идентификатора в подсистеме «Идентификации\Аутентификации» всех СКА.

При поступлении сообщения от оператора о потере контроля за его персональным идентификатором администратор СКА аналогично производит перерегистрацию нового ТМ-идентификатора оператора СКА.

3.13. Другие типы подсистем технологической защиты

Описанная подсистема технологической защиты информации касается АС, обрабатывающей большой поток ЭлД, формируемых в своей основе во внешних подсистемах. Если же речь идет об АС, в которой ЭлД и порождаются, то схема применения КА может быть упрощена. Это возможно, если на АРМ АС ЭлД в основном порождаются, а проверка их осуществляется на выделенном СКА. Дело в том, что для выработки КА необходимо знать только свои ключи, а их немного. Эту функцию на себя может взять МСU СЗИ «Аккорд 4++». Проверка же требует знания всех ключей, и для этого требуется СКА на базе «Аккорд СБ». Если такое разделение операций возможно, то схема применения КА может быть такой, как на рис. 4.7. Это позволяет резко сократить затраты на создание подсистемы технологической защиты информации.

Рисунок 4.7

Уже понятно, что обработка ЭлД может проводиться только на основе защищенных АС. Понятно также, что далеко не все защищенные АС защищены СЗИ «Аккорд 4++». Как же быть тем, чьи АС защищены другими средствами? Ведь замена всех СЗИ — мероприятие весьма дорогостоящее.

Для этих целей на рабочей станции может использоваться описанный ранее блок установки кодов аутентификации («БУКА»). При этом средствами СЗИ должна обеспечиваться целостность драйвера «БУКА», а все контрольные операции будут выполняться аппаратно. Схема применения «БУКА» показана на рис. 4.8.

Рисунок 4.8

ЗАКЛЮЧЕНИЕ

Разработанная несколько лет назад концепция аппаратной защиты подтвердила адекватность и теперь является стандартом де-факто. В настоящее время начинается новый этап развития СЗИ — переход от защиты объектов информатизации к защите электронных документов. Уже разработаны концептуальные и теоретические основы, есть аппаратные и программные реализации этих механизмов, есть даже примеры реализации этих подходов на региональном и федеральном уровнях.

Как и любой новый механизм, предлагаемый подход открывает ряд новых направлений исследований. Среди них — классификация электронных документов и объектов информатизации, разработка основ теории защиты электронных документов, создание средств защиты электронных документов и защищенных объектов информатизации, создание защищенных технологий обработки электронных документов.

ОГЛАВЛЕНИЕ

Валерий Аркадьевич Конявский, чл.-корр. РАЕН
Управление защитой информации на базе СЗИ НСД «Аккорд»

Научные редакторы к.ф.м.н. Р.З. Хафизов, к.в.н. Ю.В. Гусаров
Макет и оформление: О.В. Корытов
Иллюстрации: О.С. Овдиенко

ИБ № 2919
Издательская лицензия № 010164 от 29.01.97

Подписано в печать 2.12.99
Формат 60X90/16
Усл. печ. л. 21
Печать офсетная
Тираж 1 000 экземпляров

Издательство «Радио и связь»
103473 Москва,
2-й Щемиловский пер., д. 4/5

Типография «Наука»
Москва, Шубинский пер., д.6

ОАО «Оптима» 107082, Москва, Рубцовская наб., д.3, стр.1
Тел. (095) 263 9946, 267 3347,
Факс: 267 5362
E-mail: SlavaVV@office.optima.ru

ОАО «ИнфоТеКС» 125315, Москва, Ленинградский пр-т, 80, а/я 35
Тел. (095) 913 2135, Факс (095) 913 2119,
E-mail: soft@infotecs.ru

ТОО «МАСКОМ» 117602, Москва, ул. Академика Анохина, дом 12, к.5
Тел./факс (095) 932 7006, 437 4194,
E-mail: mascom@aha.ru

ООО «Техинформконсалтинг» 129090, Москва, 2-й Троицкий пер., д. 6а, стр. 3
Тел./факс (095) 281 9801
E-mail: tic@mail.infotel.ru, WWW.protection.ru/about.htm1

АОЗТ НПО «Защита информации» Москва, Ломоносовский пр-т, д.31, к.2
Тел./факс (095) 143 1300, 147 9266, 147 9291

АО «ЛАРРО» 107066, Москва, ул.Ст. Басманная, д.21/4,
Тел./факс (095) 931 4385

ООО «Аквариус Дата» 107076, г.Москва, ГСП-6,а/я 1, ул.Стромынка, 20
Тел. (095)269 4621, 269 4587, факс: 269 5121
E-mail: api@asi.ru , Web: www.ag.ru

ЗАО "Радиус. Технические средства безопасности» 117334, Москва, Ленинский пр-т, 41/2
Тел./факс (095) 135 3594

Фирма «Комфакс» 127018, Москва, ул. Образцова, д. 38
Тел./факс (095) 289 2179
E-mail: comfax@rfnet.ru

В регионах и странах СНГ:

НПП «Икар» 420062, Казань, ул. Журналистов, 50/3

Центр ЗИ и режимно-секретных органов при Волгоградском ГТУ 400066, г. Волгоград, пр-т Ленина, 28
(8442) 337 383, 346 774
E-mail: czinform@vistu.ru, czgtk@vistu.ru

Тверское ГП ВТИ г. Тверь
Тел./Факс (0882) 314 816, 314 714

ТОО «Экран» 680000, г. Хабаровск, ул. Гоголя, д. 27
(4212) 393 420

ООО «Парус-Волгоград» г. Волгоград, 7-я Гвардейская,д.2,оф. 334а

Региональный учебно-научный центр БИ Поволжья г. Казань
Тел./Факс: (8432) 381 600
E-mail: cnirt@kai.ru

АЛСИ 480070, Алматы, мкр. Коктем-2,д.19-А
(3272) 476 305, 473 902, ф.476 466, 473 154
kizov@alsi.kz

Фирма «ГУДВИЛЛ» г. Тольятти
E-mail: guard@infopac.ru

ЗАО ПКФ «Севкавсвязьсервис» г. Ставрополь, ул. Ленина, 251

ТОО «Элинс» г. Самара
Тел./факс: (8 846 2) 357 356
E-mail: Golikovlv@samrosest.ru

ЗАО «Кордон» (Центр технических средств контроля) 344090, г. Ростов-на-Дону, ул. Стачки 194 НИИ Физики (РГУ к.715)
Тел. (8632) 285 200, Тел./факс (8632) 221 643
E-mail: kordon@ip.rsu.ru

Фирма «Новый Альянс» 350068, г. Краснодар,
Тел./факс (8612) 685 407, 684 164
E-mail: npsna02@online.ru

МАРФИ 220046, г. Минск, ул.Радиальная, 40, оф. 225
Тел. (017) 273 9954, Тел./факс (017 230 1756
E-mail: api97@user.unibel.by

ОАО Особое Конструкторское Бюро «Карат»	644065 г. Омск, пос. Первомайский-2, ул.Заводская, 2 Тел. (3812) 645 455, 645 500, факс (3812) 644 422 E-mail: karat93@dionis.omskelecom.ru
Краснодарский филиал государственного унитарного предприятия НТЦ	350068 г. Краснодар, ул. Красноармейская, 20 Тел./факс (8612) 686 767, 680 242
«Атлас»	г. Пенза Тел. (841) 2 523 037 E-mail: gvp.@crypto.ru
ПНИИП «Сталл»	191123, г. Санкт-Петербург, а/я 149 Тел. (812) 278 6738, 278 7317 E-mail: simonov@ntc.spb.ru
НТЦ «АТЛАС» Филиал в Санкт-Петербурге и Ленинградской области	630102,г. Новосибирск, ул. Нижегородская, 6 Тел. (3832) 101 917,101 275, факс 101 134 E-mail: siv@card.ru, http://www.card.ru
ЗАО «Компания «Кардинал»	г. Саратов, Привокзальная пл. 1, оф.3 Тел./факс (8452) 515 215 E-mail: zobs@tritec.ru
Учебный центр «Трайтек» «Котра»	г. Ростов-на-Дону, ул. Беговая, 10 Тел./факс (8632) 679 717, 620 384

Средства криптографической защиты информации

Московское отделение
Пензенского научно-исследовательского
электротехнического института

МО ПНИЭИ сегодня предоставляет своим заказчикам широкие возможности управления безопасностью и криптографической защиты информации в глобальных, корпоративных и локальных сетях, предлагая клиентам комплексную и многоуровневую криптографическую защиту информации, начиная от абонента и до межрегионального и межсетевого объединения, которая позволяет обезопасить информационный обмен и предотвратить несанкционированный доступ к базам данных.

Все предлагаемые МО ПНИЭИ криптографические средства защиты информации имеют сертификаты ФАПСИ, а криптографические алгоритмы, используемые в этих средствах, соответствуют требованиям отечественных стандартов по защите информации:

— Алгоритм шифрования выполнен в соответствии с требованиями ГОСТ 28147-89 «СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ. ЗАЩИ ТА КРИПТОГРАФИЧЕСКАЯ».
— Цифровая подпись выполнена в соответствии с требованиями ГОСТ Р34.10-94 «ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ. КРИПТОГ РАФИЧЕСКАЯ ЗАЩИТА ИНФОРМАЦИИ. ПРОЦЕДУРЫ ВЫРА БОТКИ И ПРОВЕРКИ ЭЛЕКТРОННОЙ ЦИФРОВОЙ ПОДПИСИ НА БАЗЕ АСИММЕТРИЧНОГО КРИПТОГРАФИЧЕСКОГО АЛГОРИТМА».

Среди продуктов МО ПНИЭИ можно выделить: центры формирования ключей и управления ключевой системой (АРМ АБ), базовые криптоядра (СКЗИ «Верба»/ «Верба-О») в различных вариантах комплектации, криптографический сервер, проходной шифратор IP потоков (ШИП), системы защиты электронного документооборота на базе защищенной электронной почты и некоторые другие.

Все перечисленные выше средства имеют встроенные средства контроля целостности программных компонент и защиты от НСД к СКЗИ, вместе с тем, для улучшения характеристик защиты от НСД ***рекомендуется совместное использование СКЗИ с аппаратно-программным комплексом защиты от НСД «Аккорд» производства ОКБ САПР, что подтверждается соответствующими сертификатами ФАПСИ.***

При этом:

1. На основании сертификатов ФАПСИ и документации на СКЗИ «Верба» («Верба-W»), совместное использование ПАК «Аккорд» и СКЗИ «Верба» («Верба-W») является обязательным. В документации на соответствующие СКЗИ приведены рекомендации по настройке ПАК «Аккорд».

2. Совместное использование ПАК «Аккорд» и СКЗИ «Верба-О» («Верба-OW») не является обязательным, однако носит рекомендательный характер, что отражено в документации на соответствующие СКЗИ.

3. Рекомендации по настройке ПАК «Аккорд» при совместной работе с СКЗИ «Верба» / «Верба-W» (СКЗИ «Верба-О» / «Верба-OW»).

Для защиты от НСД к ПЭВМ с установленной СКЗИ со стороны посторонних лиц, службой безопасности разрабатываются и контролируются соответствующие организационно-технические меры.

В частности, в качестве защиты от НСД, рекомендуется использовать аппаратно-программный комплекс «Аккорд», при этом:

— должна быть обеспечена целостность корпуса ПЭВМ с платой «Аккорд», доступ пользователей должен быть ограничен органами уп равления и средствами ввода/вывода информации в ПЭВМ, а также контактным разъемом устройства «Аккорд»;
— должны быть предприняты меры, препятствующие извлечению платы защиты от НСД «Аккорд» из ПЭВМ — системные блоки ПЭВМ должны быть опечатаны специально выделенной для этих целей печатью;
— должна быть обеспечена сохранность используемых ТМ идентифи каторов, паролей пользователей и администратора. Пароли пользова телей и администратора должны иметь длину не менее 8 символов, при этом не допускается повтор более двух подряд идущих символов пароля.
— настройка комплекса «Аккорд» на параметры конкретной программ ной среды должна исключать возможность вмешательства пользова теля в процессы загрузки операционной системы и ПО СКЗИ и про верки целостности программной среды.

При использовании ПАК «Аккорд», проверка целостности системного, сетевого и прикладного ПО, в среде которого работают СКЗИ, должна выполняться с использованием встроенного ПО комплекса «Аккорд» на этапе загрузки операционной системы не реже 1 раза в сутки.

4. Указанные выше рекомендации не препятствуют использованию всех возможностей ПО ПАК «Аккорд» по защите ПЭВМ с установленными СКЗИ от НСД и могут включаться пользователем СКЗИ на основании документации на соответствующий ПАК «Аккорд».

Москва, ул. Образцова, 38
Тел. (095) 289 4367, 289 1054, 289 7527
Факс (095)289 4142
E-mail: info@mo.msk.ru , WWW.security.ru

ЗАКРЫТОЕ АКЦИОНЕРНОЕ ОБЩЕСТВО «РФК»

107076 Москва
Преображенская площадь, д. 6/68, стр. 3
Телефон: (095) 964 0391
Факс: (095) 964 2519

НОВОЕ ВРЕМЯ —
НОВЫЙ УРОВЕНЬ ОБСЛУЖИВАНИЯ КЛИЕНТОВ!

ЗАО «РФК» — признанный разработчик банковских компьютерных коммуникационных защищенных систем предлагает семейство программ для ***ФРОНТ-ОФИСА*** *современного банка.*
Опыт работы на рынке автоматизированных банковских систем с 1993 года.

П2000: Сертификат соответствия ГОСТ Р РОСС RU. ME 20.H00255 №0090530

ЭЛЕМЕНТЫ ЕДИНОГО «ФРОНТ-ОФИСА»

КЛИЕНТ-БАНК™ — Система электронного документооборота с необходимой защитой информации для управления счетами из офиса клиента по любым каналам связи в Off-Line и Real-Time режимах, в том числе и через Internet.

НОВИНКА **«ORACLE» версия подсистемы «Банк»** — решение для крупных и средних Банков.

Для предоставления услуг различным группам клиентов предлагается несколько типов компонент:
— **«Клиент»** – подсистема для юридических лиц;
— **«Customer»** – подсистема с интерфейсом на **английском языке** для юридических лиц;
— **«Мульти-Клиент mini»** — подсистема для группы юридических лиц;

— **«Мульти-Клиент maxi»** — подсистема для удаленных офисов и представительств банка;
— **«Мульти-Клиент local»** — подсистема для автоматизации ***Фронт-офиса*** в самом банке;
— **«Клиент-Коммунальные платежи» —** подсистема для коллективов физических лиц;
— **корпоративные решения**.

Средства защиты информации поставляются с системой и входят в их стоимость. В качестве носителя ключа используется Touch Memory или дискеты. Бесплатно поставляются вместе с комплексом интерфейсы к типовым коммуникациям и глобальным сетям (Internet, Sprint).

«Клиент-Банк Lite» — Предназначена для ускорения обслуживания клиентов в операционном зале и обработки документов клиентов, доставленных на дискетах.
Internet-Банк — Технологии «тонкого клиента». Данные решения отличается возможностью охватить спектром услуг широчайший круг клиентов во всем мире.
VOICE-ИНФОРМАТОР — система автоматического информирования клиентов по телефону о состоянии счетов и карточных вкладов с парольным доступом. Автоматизированная выдача справок.
СЗИ — средства защиты информации от НСД: аппаратно-программный комплекс ЗИ **«Аккорд».**

УСЛУГИ

Установка и обновление программного обеспечения, гарантия, консультационная поддержка, VIP сопровождение, поставка необходимой техники, выбор и предоставление коммуникационных средств, страхование рисков.

НАША ПРОДУКЦИЯ - ФИНАНСОВАЯ БЕЗОПАСНОСТЬ!

Иллюстрация А

МУЛЬТИПЛИКАТИВНАЯ ПАРАДИГМА

Уровень защищенности АС не выше уровня защищенности самого слабого звена

Если в комплексе мер
(О, ужасный пример!)
Слабое есть звено,
То защита систем
Ненадежна совсем, —
Даже если оно одно!

Ведь защита —
Такой деликатный предмет:
Если что-то забыл,
Безопасности нет!

ШАГ ЗА ШАГОМ

Создание ИПС возможно
при корректной реализации
пошагового алгоритма контроля целостности.
Корректность процедуры
контроля целостности на очередном уровне
может основываться только
на корректных результатах контроля
на предыдущем уровне

Корректность первого этапа обеспечивается технологически

Результат — создание ИПС

Hard или Soft?

Основной вопрос
информационной безопасности —
что первично: hard или soft?

СИНДРОМ МЮНХГАУЗЕНА

Преодоление
«Синдрома Мюнхаузена»
состоит в последовательном
отказе от программных методов контроля, как
очевидно ненадежных

ПРИНЦИП АРХИМЕДА

«Принцип Архимеда»
состоит в переносе
наиболее критичных контрольных процедур
на аппаратный уровень — создание «точки опоры»

Иллюстрация Е

ОТЧУЖДАЙ И ВЛАСТВУЙ

Разделение программ и данных способствует повышению конфиденциальности

Отчуждая информацию, знаем и верим:
Чтоб хозяину вор помешать не мог,-
Запирая, ключ не клади под дверью,
А то не спасет никакой замок!

Чтобы все было в жизни правильно,
Надежно и безопасно,
Помни основополагающее правило:
Отчуждай и властвуй!

Иллюстрации Ж

АККОРД 4++

Иллюстрации З

АККОРД СБ

БУКА