ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«НАЦИОНАЛЬНЫЙ ЦЕНТР ПОДДЕРЖКИ И РАЗРАБОТКИ»

Операционная система «МСВСфера»

Описание применения

ЦАУВ.27001-01 31 01

(Листов - 31)

СОДЕРЖАНИЕ

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

ИИ – Извещение об изменении
НТД – Научно техническая документация
РД – Руководящий документ
РКД – Рабочая конструкторская документация
НТД – Научно-техническая документация
ОБ – Обновление безопасности
ПО – Программное обеспечение
ФО – Федеральный орган по сертификации

АННОТАЦИЯ

Настоящий документ является описанием применения операционной системы «МСВСфера» (далее по тексту — ОС, Изделие).

В документе описаны назначение ОС, условия ее применения, описание задачи, приведены входные и выходные данные. Также приведены сведения по получению обновлений ОС.

1. НАЗНАЧЕНИЕ ПРОГРАММЫ

1.1. Назначение

ОС является операционной системой типа «А», предназначенной для функционирования на средствах вычислительной техники общего назначения (автоматизированные рабочие места, серверы, планшетные компьютеры и иные) и предназначена для применения в составе информационных (автоматизированных) систем в целях обработки и защиты от НСД к информации любой категории доступа[1]: общедоступной информации, а также информации, доступ к которой ограничен федеральными законами (информации ограниченного доступа).

1.2. Основные характеристики

В состав ОС входят следующие программные компоненты:

- ядро ОС с поддержкой технологии виртуализации;
- средства установки и настройки ОС;
- системные и сервисные утилиты;
- базовые сетевые службы;
- средства управления конфигурациями и администрирования;
- средства управления программными пакетами;
- средства резервного копирования и восстановления данных;
- средства виртуализации;
- средства контейнеризации;
- средства управления доступом;
- средства управления информационными потоками;
- средства регистрации событий безопасности;
- средства обеспечения замкнутой программной среды;
- средства гарантированного стирания данных;
- средства контроля целостности;
- средства обеспечения надёжного функционирования;
- средства мониторинга функционирования;
- комплекс программ для функционирования веб-серверов;
- комплекс программ электронной почты;
- комплекс программ для серверов файлов и печати;
- комплекс мультимедийных программ;
- комплекс программ для идентификации, аутентификации и удаленного

1 В соответствии с Федеральным законом от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и о защите информации» (статья 5, пункт 2)

управления;

– пакет офисных программ.

1.3. Возможности

Изделие предоставляет следующие возможности:

– установку и функционирование на средствах вычислительной техники с процессорной архитектурой x86-64, а также поддержку периферийного оборудования;

– поддержку основных сетевых протоколов стека TCP/IP;

– работу с мультимедийными и гипертекстовыми данными;

– работу с реляционными базами данных;

– работу с электронной почтой;

– обработку текстовых документов и электронных таблиц различных форматов.

Дополнительно Изделие предоставляет следующие функциональные возможности по защите информации:

1. Обеспечивающие реализацию функциональных требований безопасности, предъявляемых к четвертому классу защиты операционных систем типа «А» в соответствии с требованиями документа ФСТЭК России «Профиль защиты операционных систем типа «А» четвертого класса защиты» ИТ.ОС.А4.ПЗ:

– возможность задания политики дискреционного и ролевого управления доступом для установленного множества операций, выполняемых субъектами доступа по отношению к объектам доступа;

– возможность реализации дискреционного и ролевого управления доступом на основе списков управления доступом (или матрицы управления доступом) и (или) ролей;

– возможность осуществления резервного копирования объектов файловой системы;

– возможность удаления объектов файловой системы;

– возможность восстановления объектов ОС из резервных копий, созданных с использованием ОС, и использования ассоциированных с ними атрибутов безопасности;

– возможность установки ПО (компонентов ПО) только администраторами;

– возможность задания правил автоматического запуска компонентов ПО при загрузке ОС;

– возможность контроля запуска компонентов ПО и реагирование на попытки запуска компонентов ПО, произведенные в нарушение установленных

правил запуска компонентов ПО;

– возможность осуществлять фильтрацию входящих и (или) исходящих сетевых потоков;

– возможность осуществлять фильтрацию сетевых потоков, основанную на следующих типах атрибутов безопасности субъектов доступа:

1) сетевой адрес узла отправителя;

2) сетевой адрес узла получателя;

3) и информации: сетевой протокол, который используется для взаимодействия.

– возможность явно разрешать сетевой поток, базируясь на устанавливаемом администратором наборе правил фильтрации сетевого трафика, основанном на идентифицированных атрибутах;

– возможность запрещать сетевой поток, базируясь на устанавливаемом администратором наборе правил фильтрации сетевого трафика, основанном на идентифицированных атрибутах;

– возможность удаления объектов файловой системы путем многократной перезаписи уничтожаемых (стираемых) объектов файловой системы специальными битовыми последовательностями;

– возможность обеспечения недоступности остаточной информации при распределении или освобождении ресурса памяти;

– возможность задания правил запуска компонентов ПО в процессе функционирования ОС;

– возможность контроля целостность компонентов ПО, разрешенного для запуска, и реагирования на попытки запуска компонентов ПО, целостность которых была нарушена;

– возможность осуществлять фильтрацию сетевого потока, основанную на атрибутах: разрешенные (запрещенные) протоколы прикладного уровня;

– возможность осуществлять фильтрацию сетевого потока, основанную на следующих типах атрибутов безопасности информации: транспортный протокол, который используется для взаимодействия, порты источника и получателя в рамках сеанса (сессии);

– возможность поддерживать для каждого пользователя ОС список атрибутов безопасности;

– возможность блокирования учетной записи пользователя ОС при превышении установленного администратором числа неуспешных попыток аутентификации;

– возможность обеспечения идентификации объектов доступа;

– возможность идентификации пользователя до выполнения действий по доступу в информационную систему или администратора до выполнения действий по управлению ОС;

– возможность исключения отображения действительного значения аутентификационной информации при ее вводе пользователем ОС в диалоговом интерфейсе;

– возможность ассоциировать атрибуты безопасности пользователя ОС с субъектами доступа (запускаемыми от его имени процессами);

– возможность проверки соответствия аутентификационной информации метрике качества, обеспечивающей адекватную защиту от нарушения безопасности нарушителем с потенциалом нападения, соответствующим классу защищенности;

– возможность идентификации и аутентификации пользователя до выполнения любых действий по доступу в информационную систему или администратора до выполнения действий по управлению ОС;

– возможность поддержки многофакторной или двухфакторной аутентификации;

– возможность со стороны администратора управлять атрибутами безопасности;

– возможность со стороны администратора управлять выполнением функций безопасности ОС;

– возможность со стороны администратора управлять параметрами функций безопасности ОС, данными аудита, правилами фильтрации сетевого потока;

– возможность поддержки определенных ролей для ОС и их ассоциации с пользователями ОС;

– возможность применения наборов базовых конфигураций ОС;

– возможность устанавливать пороговое значение количества неуспешных попыток аутентификации, предоставляемая администратору;

– возможность устанавливать срок действия паролей, предоставляемая администратору;

– возможность устанавливать срок действия идентификаторов для временных учетных записей, предоставляемая администратору;

– возможность обеспечения ограничительных значений по умолчанию для атрибутов безопасности, которые используются для осуществления политики дискреционного управления доступом;

– возможность обеспечения управления доступом к объектам ОС;

– возможность защиты хранимой аутентификационной информации от неправомерного доступа к ней и раскрытия;

– возможность обеспечения защиты от переполнения буфера;

– возможность постоянного контроля и проверки правомочности обращений субъектов доступа к объектам доступа;

– возможность предоставления надежных меток времени при проведении аудита, а также для ограничения срока действий атрибутов безопасности;

– возможность тестирования (самотестирования) функций безопасности ОС, проверки целостности ПО ОС и целостности данных (параметров) ОС;

– возможность возврата ОС при сбоях и отказах к безопасному состоянию в автоматизированном режиме;

– возможность работы экземпляров ОС на нескольких технических средствах в отказоустойчивом режиме, обеспечивающем доступность сервисов и информации при выходе из строя одного из технических средств (отказоустойчивый кластер);

– возможность работы экземпляров ОС на нескольких технических средствах в режиме балансировки нагрузки, обеспечивающем доступность сервисов и информации в условиях компьютерных атак, направленных на отказ в обслуживании, приводящих к полному исчерпанию вычислительных ресурсов одного из технических средств (кластер с балансировкой нагрузки);

– возможность предоставления приоритетов для использования субъектами доступа подмножества вычислительных ресурсов средства вычислительной техники под контролем функций безопасности ОС;

– возможность квотирования ОС вычислительных ресурсов средства вычислительной техники;

– возможность осуществлять блокирование сеанса доступа пользователя ОС по истечении заданного интервала времени бездействия;

– возможность осуществлять блокирование (разблокирование) собственного сеанса доступа в ОС пользователем ОС;

– возможность автоматически осуществлять блокирование интерактивного сеанса пользователя ОС после установленного периода бездействия;

– возможность осуществлять блокирование интерактивного сеанса по требованию уполномоченного привилегированного субъекта доступа;

– возможность в ОС отдельно осуществлять ограничение максимального числа одновременных (параллельных) интерактивных сеансов, предоставляемых уполномоченным привилегированным субъектам доступа и уполномоченным непривилегированным субъектам доступа;

– возможность обеспечения защиты от несогласованностей, возникающих на

уровне процессов при параллельной работе с ресурсами средства вычислительной техники и объектами доступа ОС;

– возможность блокирования попыток доступа к объектам доступа, если в момент обращения они используются другими процессами;

– возможность безопасного выделения областей оперативной памяти;

– возможность регистрации и учета выполнения проверок при фильтрации сетевого потока;

– возможность реагирования при обнаружении событий, указывающих на возможное нарушение безопасности;

– возможность включения и исключения событий в совокупность событий, подвергающихся аудиту, предоставляемая администратору;

– возможность предоставления администратору всей информации аудита в понятном для него виде;

– возможность защиты хранимых записей регистрации событий безопасности ОС (аудита) от несанкционированного удаления и предотвращения модификации записей аудита;

– возможность выполнения действий, направленных на сохранение данных журнала регистрации событий безопасности ОС и обеспечивающих непрерывность процесса аудита, если журнал регистрации событий безопасности ОС превысит определенный администратором размер;

– возможность регистрации (аудита) событий безопасности, которые в соответствии с национальным стандартом Российской Федерации ГОСТ Р ИСО/МЭК 15408-2-2013 «Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 2. Функциональные компоненты безопасности» включены в базовый уровень аудита;

– возможность просмотра записей аудита только администратором;

– возможность выборочного просмотра данных регистрации (аудита) событий безопасности ОС (поиск, сортировка, упорядочение данных аудита);

– возможность выполнения действий, направленные на предотвращение потери данных аудита при переполнении журнала регистрации событий безопасности ОС;

– возможность полнотекстовой регистрации привилегированных команд (команд, управляющих системными функциями);

– возможность передавать данные аудита для внешнего хранения.

2. *Обеспечивающие реализацию функциональных требований безопасности, предъявляемых к четвертому классу защиты средств виртуализации в*

соответствии с требованиями документа ФСТЭК России «Требования по безопасности информации к средствам виртуализации»:

– возможность блокировать запуск виртуальной машины при выявлении нарушения целостности конфигурации виртуального оборудования данной виртуальной машины;

– возможность блокировать запуск виртуальной машины при выявлении нарушения целостности файлов виртуальной базовой системы ввода-вывода (первичного загрузчика виртуальной машины) и (или) исполняемых файлов гостевой операционной системы;

– возможность контролировать целостность в процессе загрузки и динамически в процессе функционирования средства виртуализации объектов контроля самостоятельно или с применением сертифицированных хостовой операционной системы или средства доверенной загрузки;

– возможность информировать администратора безопасности средства виртуализации о нарушении целостности объектов контроля;

– возможность контролировать целостность конфигурации виртуального оборудования виртуальных машин;

– возможность контролировать целостность исполняемых файлов и параметров настройки средства виртуализации;

– возможность обеспечивать целостность сведений о событиях безопасности;

– возможность обеспечивать регистрацию событий безопасности, связанных с функционированием средства виртуализации;

– возможность оповещать администратора безопасности средства виртуализации о событиях безопасности;

– возможность выполнять действия, являющиеся реакцией на события безопасности самостоятельно или с применением сертифицированных средств защиты информации;

– возможность осуществлять сбор и хранение записей в журнале событий безопасности, которые позволяют определить, когда и какие действия происходили;

– возможность для регистрируемых событий безопасности в каждой записи журнала событий безопасности дополнительно должно регистрироваться описание события безопасности, включающее сведения о его важности;

– возможность реализовать ролевой метод управления доступом субъектов доступа среды виртуализации к объектам доступа среды виртуализации;

– возможность обеспечивать возможность определения полномочий для пользователей средства виртуализации в пределах назначенных им ролей;

– возможность обеспечивать резервное копирование образов виртуальных машин и конфигурации виртуального оборудования виртуальных машин самостоятельно или с применением хостовой операционной системы или сертифицированных средств резервного копирования;

– возможность обеспечивать резервное копирование параметров настройки средства виртуализации;

– возможность обеспечивать резервное копирование сведений о событиях безопасности;

– возможность осуществлять контроль за запуском компонентов программного обеспечения, обеспечивающий выявление и блокировку запуска компонентов программного обеспечения, не включенных в перечень (список) компонентов, разрешенных для запуска;

– возможность осуществлять контроль за запуском компонентов программного обеспечения, обеспечивающий:

– возможность выявление и блокировку запуска компонентов программного обеспечения, целостность которого нарушена;

– возможность блокировки запуска компонентов программного обеспечения, не прошедших аутентификацию с использованием свидетельств подлинности модулей (в том числе цифровых сигнатур производителя или иных свидетельств подлинности модулей);

– возможность осуществлять управление потоками информации между виртуальными машинами и информационными (автоматизированными) системами на канальном и сетевом уровнях самостоятельно или с применением сертифицированных средств управления потоками информации (коммутаторов, маршрутизаторов) и (или) межсетевых экранов, а также контроль взаимодействия виртуальных машин между собой;

– возможность очищать остаточную информацию в памяти средства вычислительной техники при ее освобождении (распределении) или блокирование доступа субъектов к остаточной информации;

– возможность удалять объекты файловой системы, используемые средством виртуализации, путем перезаписи уничтожаемых (стираемых) объектов файловой системы случайной битовой последовательностью;

– возможность размещать код средства виртуализации в области памяти, не доступной одновременно для записи и исполнения;

– возможность изолировать области памяти виртуальных машин;

– возможность обеспечивать возможность администратору средства

виртуализации осуществлять первичную идентификацию пользователей средства виртуализации;

– возможность блокировать доступ пользователям средства виртуализации при неуспешной идентификации и аутентификации последних в средстве виртуализации;

– возможность осуществлять аутентификацию пользователей при предъявлении идентификатора и пароля пользователя;

– возможность обеспечивать возможность администратору средства виртуализации устанавливать пароль пользователя для первичной аутентификации;

– возможность обеспечивать смену установленного администратором средства виртуализации пароля пользователя средства виртуализации после его первичной аутентификации;

– обеспечивать невозможность установления одинаковых идентификаторов и паролей для разных пользователей;

– возможность осуществлять вывод сообщения с приглашением ввести правильный идентификатор и пароль еще раз при попытке ввода неправильного значения идентификатора или пароля пользователем;

– возможность блокировать учетную запись пользователя средства виртуализации с возможностью разблокировки администратором средства виртуализации или с возможностью автоматической разблокировки по истечении временного интервала, устанавливаемого администратором средства виртуализации, при исчерпании установленного максимального количества неуспешных попыток ввода неправильного пароля;

– возможность обеспечивать защиту пароля пользователя средства виртуализации при его вводе за счет отображения вводимых символов условными знаками;

– обеспечивать возможность администратору средства виртуализации задавать и корректировать парольные политики (сложность пароля, количество неуспешных попыток аутентификации до блокировки) средства виртуализации;

– возможность обеспечивать хранение аутентификационной информации пользователя средства виртуализации в защищенном формате или в защищенном хранилище;

– возможность обеспечивать взаимную идентификацию и аутентификацию пользователей и средства виртуализации при удаленном доступе с использованием сетей связи общего пользования;

– возможность создавать, модифицировать, хранить, получать и удалять образы виртуальных машин в информационной (автоматизированной) системе;

– возможность обеспечивать чтение записей о событиях безопасности, формирование отчетов с учетом заданных критериев отбора, выгрузку (экспорт) данных из журнала событий безопасности средства виртуализации;

– возможность обеспечивать управление размещением и перемещением виртуальных машин и их образов с возможностью сохранения их конфигурации и настроек.

3. *Обеспечивающие реализацию функциональных требований безопасности, предъявляемых к четвертому классу защиты средств виртуализации в соответствии с требованиями документа ФСТЭК России «Требования по безопасности информации к средствам контейнеризации»:*

– возможность реализовывать механизмы изоляции контейнеров;

– возможность реализовывать следующие механизмы:

1) изоляция пространств идентификаторов процессов контейнеров;

2) изоляция пространств имен для межпроцессного взаимодействия контейнеров;

3) изоляция пространств имен для пользователей и групп контейнеров;

4) изоляция пространств имен хостов и доменов контейнеров;

5) изоляция сетевых пространств имен контейнеров;

6) изоляция пространств имен для иерархии каталогов контейнеров.

– возможность выявлять известные уязвимости при создании, установке образа контейнера в информационной (автоматизированной) системе и хранении образов контейнеров во взаимодействии с сертифицированным средством контроля и анализа защищенности на основе сведений, содержащихся в общедоступных банках данных угроз безопасности информации, содержащих сведения об известных уязвимостях программного обеспечения;

– возможность оповещать о выявленных уязвимостях в образах контейнеров разработчика образов контейнеров и администратора безопасности информационной (автоматизированной) системы;

– возможность осуществлять выявление известных уязвимостей образов контейнеров не реже одного раза в неделю;

– возможность запрещать создание образов контейнеров, содержащих известные уязвимости критического и высокого уровня опасности;

– возможность обеспечивать ограничение прав прикладного программного обеспечения, выполняемого внутри контейнера, на использование периферийных устройств, устройств хранения данных и съемных машинных носителей информации (блочных устройств), входящих в состав информационной (автоматизированной) системы;

– возможность обеспечивать ограничение прав прикладного программного обеспечения, выполняемого внутри контейнера, на использование вычислительных ресурсов (оперативной памяти, операций ввода-вывода за период времени) хостовой операционной системы;

– возможность обеспечивать монтирование корневой файловой системы хостовой операционной системы в режиме "только для чтения";

– возможность контролировать самостоятельно или с применением средств контроля целостности хостовой операционной системы и иных сертифицированных средств защиты информации целостность образов контейнеров и исполняемых файлов контейнеров;

– возможность информировать администратора информационной (автоматизированной) системы и администратора безопасности средства контейнеризации о нарушении целостности объектов контроля;

– возможность контролировать целостность параметров настройки средства контейнеризации;

– возможность контролировать целостность сведений о событиях безопасности самостоятельно или во взаимодействии с хостовой операционной системой и иными сертифицированными средствами защиты информации;

– возможность контролировать целостность образов контейнеров и параметров настройки средства контейнеризации при установке образа контейнера в информационной (автоматизированной) системе и далее периодически за счет применения цифровой подписи самостоятельно или во взаимодействии с хостовой операционной системой и иными сертифицированными средствами защиты информации;

– возможность блокировать запуск образа контейнера при нарушении его целостности;

– возможность регистрировать события, относящиеся к инцидентам безопасности средства контейнеризации, связанные с попытками осуществления несанкционированного доступа к средству контейнеризации;

– возможность оповещать администратора безопасности средства контейнеризации и администратора информационной (автоматизированной) системы об инцидентах безопасности;

– возможность выполнять действия, являющиеся реакцией на инциденты безопасности;

– возможность осуществлять сбор и хранение записей в журнале событий безопасности, которые позволяют определить, когда и какие действия происходили;

– возможность обеспечивать запись событий безопасности контейнеров в журнал событий безопасности информационной (автоматизированной) системы с указанием идентификатора пользователя хостовой операционной системы, от имени которого был запущен контейнер.

1.4. Функции безопасности

ОС реализует следующие функции безопасности для защиты информации:

1) на уровне операционной системы:

– идентификацию и аутентификацию;
– управление доступом;
– регистрацию событий безопасности;
– ограничение программной среды;
– изоляцию процессов;
– защиту памяти;
– контроль целостности;
– надежное функционирование;
– фильтрацию сетевого потока.

2) на уровне средства виртуализации:

– доверенную загрузку виртуальных машин;
– контроль целостности;
– регистрацию событий безопасности;
– управление доступом;
– резервное копирование;
– управление потоками информации;
– защиту памяти;
– ограничение программной среды;
– идентификацию и аутентификацию пользователей;
– централизованное управление образами виртуальных машин и виртуальными машинами.

3) на уровне средства контейнеризации:

– изоляцию контейнеров;
– выявление уязвимостей в образах контейнеров;
– проверку корректности конфигурации контейнеров;
– контроль целостности контейнеров и их образов;
– регистрацию событий безопасности.

2. УСЛОВИЯ ПРИМЕНЕНИЯ

2.1. Требования к техническим средствам

Для функционирования ОС необходима следующая минимальная конфигурация оборудования:

– аппаратная платформа — процессор с архитектурой не ниже x86-64-v2 (Intel Nehalem и более поздние, AMD Bulldozer и более поздние);

– оперативная память — от 2 ГБ;

– объем свободного дискового пространства — от 20 ГБ;

Для функционирования системы под управлением МСВСфера наличие следующих устройств не обязательно, при этом для ее установки они необходимы:

– устройство чтения DVD-дисков (либо USB-дисков)[2];

– устройство оперативной визуальной связи пользователя с управляющим устройством и отображением данных, передаваемых с клавиатуры, мыши или центрального процессора (например, стандартный монитор SVGA 15").

2.2. Порядок эксплуатации

При эксплуатации ОС на объектах информатизации, обрабатывающих информацию ограниченного доступа, необходимо выполнение следующих ограничений:

– запрет на использование ОС для обработки информации, содержащей сведения, составляющие государственную тайну;

– обеспечение физической сохранности средств вычислительной техники с установленной ОС и исключение возможности доступа к ним посторонних лиц;

– проведение периодического контроля целостности исполняемых файлов и библиотек с помощью вычисления контрольных сумм соответствующими средствами и сравнения с эталонными значениями, хранящимися в файле sha256sums.txt;

– проведение периодической проверки на наличие компьютерных вирусов с использованием средств антивирусной защиты;

– установка и настройка ОС на автоматизированные рабочие места и сервера должны проводиться в соответствии с положениями документа «Операционная система «МСВСфера». Руководство администратора» ЦАУВ.27001-01 32 01 (входит в комплектность документации на Изделие), расположенном на сайте www.msvsphere-os.ru;

– обновление программного обеспечения ОС должно проводиться в соответствии с положениями раздела 3 настоящего документа;

2 Для вариантов поставки в коробочном исполнении или OEM-поставки

– настройка, использование и контроль средств защиты информации ОС должны проводиться ответственными за эксплуатацию Изделия в соответствии с утвержденной политикой безопасности организации, организационно-методическими документами принятой системы защиты информации, положениями документа «Операционная система «МСВСфера». Руководство администратора» ЦАУВ.27001-01 32 01 и «Операционная система «МСВСфера». Формуляр» ЦАУВ.27001-01 30 01 (входят в комплектность документации на Изделие).

2.3. Требования к пользователям

Ко всем пользователям ОС предъявляется следующее требование: базовые навыки работы с ОС семейства Linux.

К администратору ОС предъявляются следующие требования:

– базовые навыки администрирования ОС семейства Linux;

– навыки конфигурирования и настройки программных продуктов и ОС;

– опыт работы со стандартными элементами графического интерфейса приложений;

– навыки поддержания в работоспособном состоянии технических средств ПК;

– навыки настройки средств защиты информации и средств электронной подписи.

3. ПОРЯДОК ОБНОВЛЕНИЯ ОС

Администратору необходимо периодически проверять сайт Изготовителя (www.msvsphere-os.ru) на наличие сведений об уязвимостях и обновлениях ПО, устраняющих уязвимости Изделия. Для реализации функций по защите информации в Изделии должны быть установлены все актуальные обновления безопасности.

Сертифицированные обновления Изделия должны быть загружены для последующей установки с сайта Изготовителя (www.msvsphere-os.ru) или с верифицированных инсталляционных комплектов наборов обновлений на материальных носителях.

Перед установкой дистрибутива изделия и наборов сертифицированных обновлений, загруженных с сайта Изготовителя (www.msvsphere-os.ru), необходимо убедиться в их подлинности и целостности (провести их верификацию) одним из следующих способов:

– Проверить контрольные суммы дистрибутива и (или) наборов обновлений Изделия с помощью средств контроля эффективности (целостности) средств защиты информации, реализующих алгоритм «Уровень-3» (например, с помощью программ ФИКС 2.0.1, ФИКС 2.0.2, ФИКС Unix 1.0), либо с помощью утилиты sha256sum из состава Изделия. Рассчитанная контрольная сумма дистрибутива (обновления) должна совпадать с контрольной суммой, приведенной в Формуляре или представленной в текстовом файле, приложенном к обновлению Изделия.

– Проверить электронную подпись (ЭП), которой подписаны дистрибутив Изделия или набор сертифицированных обновлений. ЭП должна принадлежать Изготовителю изделия (ООО «НЦПР», ИНН 7705776758), в дистрибутиве и наборах сертифицированных обновлений должны отсутствовать изменения, ЭП не должна быть отозванной на момент подписания. Инструкции по проверке ЭП располагаются на сайте Изготовителя (www.msvsphere-os.ru).

Руководство по загрузке сертифицированных обновлений и эксплуатационной документации Изделия и их верификации расположено на сайте Изготовителя.

Актуальные значения контрольных сумм исполняемых файлов, подлежащих периодическому контролю после установки Изделия или его сертифицированных обновлений, указываются в текстовом файле sha256sums.txt, размещенном на сайте Изготовителя (www.msvsphere-os.ru).

4. ОПИСАНИЕ ЗАДАЧИ

4.1. Определение задачи

ОС обеспечивает взаимодействие пользователя и приложений с аппаратным обеспечением компьютера. Основные задачи, которые решает операционная система можно разбить на несколько категорий:

– управление процессами;

– управление памятью;

– управление файловой системой;

– управление устройствами ввода и вывода;

– обеспечение пользовательского интерфейса;

– обеспечение безопасности;

– обеспечение сетевого взаимодействия;

– поддержка виртуализации;

– поддержка контейнеризации.

4.2. Методы решения

4.2.1. Управление процессами

Решение задачи управления процессами, которые представляют собой исполняемые программы и их текущие состояния, включает в себя:

– запуск и завершение процессов;

– планирование процессов (выбор того, какой процесс будет выполняться в данный момент);

– многозадачность (одновременное выполнение нескольких процессов с использованием планировщика задач);

– виртуализация процессов (каждый процесс получает свою виртуальную память и окружение);

– межпроцессное взаимодействие (IPC): ОС обеспечивает способы для процессов взаимодействовать друг с другом (например, через очереди сообщений, сокеты, каналы).

В ОС используется systemd в качестве основного менеджера процессов и сервисов. Запуск новых процессов в ОС происходит с помощью системных вызовов fork() и exec(). fork() создает новый процесс (потомок), являющийся копией родительского процесса, он клонирует текущее состояние процесса. Системный вызов exec() заменяет код и контекст текущего процесса на код и данные новой программы. Это позволяет запущенному процессу выполнять другую программу.. Эти вызовы используются как на уровне ядра, так и в пользовательских

приложениях.

Ядро ОС использует полностью справедливый планировщик (CFS) для управления очередью процессов и выделения процессорного времени. Этот планировщик работает по принципу справедливого распределения процессорных ресурсов между процессами. Процессы могут иметь разные приоритеты, что влияет на частоту их выполнения. Применяются механизмы nice и renice для изменения их приоритетов. Планировщик также отслеживает, сколько процессорного времени было выделено каждому процессу, и старается равномерно распределять ресурсы. Система поддерживает одновременное выполнение нескольких потоков и процессов с использованием многопроцессорных систем.

Межпроцессное взаимодействие (IPC) позволяет процессам в операционной системе обмениваться данными и координировать свои действия. IPC используется для создания сложных приложений, которые состоят из нескольких процессов, работающих совместно.

4.2.2. Управление памятью

ОС управляет оперативной памятью (RAM) компьютера, предоставляя процессам необходимые ресурсы, защищая их от несанкционированного доступа.

Физическая память (оперативная память, RAM) является конечным ресурсом, и ядро Linux отвечает за её эффективное распределение между процессами.

Основные задачи управления физической памятью: аллокация и деаллокация, кэширование, перемещение данных. ОС распределяет память процессам по их запросам и освобождает её, когда процесс завершается или больше не использует память. ОС активно использует свободную память для кэширования данных с дисков, что ускоряет доступ к файлам и программам. Когда оперативная память исчерпывается, ОС может переместить неактивные страницы в своп, освобождая ресурсы для более активных процессов.

ОС использует различные системы аллокации памяти (Slab и Slub Allocator) для выделения небольших блоков памяти. Система управления буферами памяти Slab Allocator оптимизирует выделение и освобождение памяти для небольших объектов. Она минимизирует фрагментацию памяти. Более современная версия Slub Allocator, которая используется по умолчанию, проще и быстрее по сравнению с slab и уменьшает накладные расходы на управление.

Для управления памятью в ОС также используется фоновый процесс kswapd, который отвечает за освобождение оперативной памяти, когда её объём становится критически малым. Он перемещает страницы, которые давно не использовались, в своп-файл на диске.

ОС поддерживает механизм прозрачных больших страниц (Transparent Huge

Pages), который позволяет использовать большие блоки памяти (например, 2 МБ вместо стандартных 4 КБ страниц). Это улучшает производительность при работе с большими массивами данных, так как уменьшает количество обращений к таблице страниц и снижает накладные расходы на управление памятью.

NUMA-архитектура позволяет управлять памятью на системах с несколькими процессорными узлами. В таких системах каждая группа процессоров имеет свою локальную память, доступ к которой быстрее, чем к памяти других узлов. ОС старается выделять память, которая ближе к процессору, выполняющему процесс, что ускоряет доступ к данным, а также распределяет память и задачи таким образом, чтобы минимизировать конкуренцию за ресурсы между процессами.

4.2.3. Управление файловой системой

Операционная система предоставляет абстракцию файловой системы для хранения данных на дисках и других носителях, обеспечивает управление операциями чтения, записи, создания и удаления файлов.

Конфигурационный файл /etc/fstab используется для автоматического монтирования файловых систем при загрузке системы. В этом файле указываются точки монтирования, типы файловых систем, параметры монтирования и другие важные настройки.

ОС использует LVM для гибкого управления дисковыми разделами. LVM позволяет объединять несколько физических дисков в логические тома, которые можно динамически изменять в размерах, создавать моментальные снимки и управлять ресурсами более гибко, чем при использовании традиционных статических разделов. LVM позволяет изменять размеры томов, перемещать данные между физическими дисками в реальном времени без остановки работы. LVM поддерживает создание снимков томов, что упрощает резервное копирование и восстановление данных.

ОС поддерживает несколько типов файловых систем, каждая из которых предназначена для различных сценариев использования. Основная — XFS, наиболее рекомендуемая файловая система по умолчанию в ОС. Это высокопроизводительная файловая система с поддержкой журналирования и масштабирования до экстремально больших объёмов данных (до 500 ТБ). XFS особенно эффективна при работе с большими файлами и многозадачностью.

XFS предпочитается для больших систем, поддержка файловой системы ext4, которая в основном работает с небольшими и средними объёмами данных, остаётся вариантом для совместимости и определённых задач.

Для совместимости с Windows и другими операционными системами, особенно для съёмных носителей (например, USB-накопителей) используются FAT

и NTFS.

4.2.4. Управление устройствами ввода-вывода

Драйверы устройств обеспечивают взаимодействие операционной системы с аппаратными устройствами. В ОС драйверы устройств являются частью ядра Linux и могут быть загружены динамически (в виде модулей ядра) или встроены в ядро.

Большинство драйверов в ОС загружаются как модули ядра и управляются с помощью modprobe и lsmod. Утилита modprobe используется для загрузки или удаления модулей ядра. lsmod показывает список загруженных модулей.

Модули ядра могут загружаться автоматически, когда подключается новое устройство, что делается через систему udev, которая отвечает за создание файлов устройств (device files) в каталоге /dev, которые используются для доступа к аппаратным устройствам. При подключении устройства udev автоматически создает соответствующие файлы в /dev (например, /dev/sda для жесткого диска). Система позволяет настраивать правила для различных типов устройств, например, изменять права доступа или назначать пользовательские имена устройствам. Правила хранятся в каталоге /etc/udev/rules.d/.

Для управления операциями ввода-вывода в ОС используются планировщики ввода-вывода (I/O schedulers), которые контролируют порядок выполнения операций чтения и записи на устройства хранения. Ядро ОС поддерживает несколько планировщиков:

– none: минимальная задержка и накладные расходы, предпочтителен для высокопроизводительных SSD;

– mq-deadline: предотвращает проблему "голодания" операций и обеспечивает равномерную задержку;

– bfq: планировщик, ориентированный на обработку данных, часто используется для систем с традиционными жесткими дисками и медленными устройствами;

– kyber: оптимизирован для систем с SSD и NVMe, минимизирует латентность.

4.2.5. Обеспечение интерфейса пользователя

ОС предоставляет командную строку (CLI) и графический интерфейс (GUI) для взаимодействия пользователя с системой. Командная строка — это основной интерфейс для администраторов и опытных пользователей в ОС. CLI обеспечивает гибкость и полный контроль над системой, предлагая тысячи команд для выполнения различных задач. По умолчанию ОС использует оболочку Bash для работы с командной строкой. Она предоставляет возможности для выполнения команд, написания скриптов и управления системой. Основные функции оболочки

Bash:

– ввод и исполнение команд системы, например, для управления файлами и процессами.

– возможность автоматизации задач с помощью скриптов.

– поддержка переменных среды для настройки рабочей среды пользователя.

Для пользователей, которым необходим графический интерфейс, ОС предоставляет рабочее окружение GNOME с набором приложений и инструментов для взаимодействия с системой. Это оболочка поддерживает возможность управления окнами приложений, виртуальными рабочими столами и виджетами, включает в себя меню приложений, панель задач для быстрого доступа к системным функциям и запущенным приложениям. Для работы с файлами и папками через графический интерфейс используется файловый менеджер.

Запуск графического интерфейса осуществляется через дисплейный менеджер GDM (GNOME Display Manager), который отвечает за вход пользователей в систему и запуск графической сессии.

ОС поддерживает два графических сервера: Wayland, X11. По умолчанию используется Wayland как более современный и безопасный графический сервер. X11 предоставляет обратную совместимость и поддержку для приложений, которые ещё не поддерживают Wayland.

Пользователь может переключаться между Wayland и X11 при входе в систему через экран входа GDM.

4.2.6. Обеспечение безопасности

ОС обеспечивает выполнение следующих функций безопасности:

– идентификацию и аутентификацию;

– управление доступом;

– регистрацию событий безопасности;

– ограничение программной среды;

– изоляцию процессов;

– защиту памяти;

– контроль целостности;

– надежное функционирование;

– фильтрацию сетевого потока.

4.2.6.1. Идентификация и аутентификация

Доступ к объектам ОС и ее ресурсам возможен только для зарегистрированных пользователей, прошедших идентификацию и аутентификацию, которая осуществляется через модули PAM и конфигурационные

файлы в каталоге /etc/pam.d. Для предупреждения пользователей о мерах безопасности используется файл /etc/issue, а учетные записи блокируются при определенном количестве неудачных попыток входа с помощью модуля pam_faillock. Информация об учетных записях хранится в каталоге /etc/, а за настройку параметров аутентификации отвечают файлы system-auth и password-auth.

Механизмы аутентификации предоставляют скрытую обратную связь и защищают пароли, хранящиеся в файле /etc/shadow. При помощи PAM настраивается контроль качества паролей, определяемый конфигурациями в /etc/security/pwquality.conf и других файлах PAM. Каждый объект ОС идентифицируется перед выполнением с ним любых действий, а доступ к файлам регулируется через системные вызовы. Политика управления доступом предотвращает хранение аутентификационных данных в открытом виде благодаря установленным правам доступа.

4.2.6.2. Управление доступом

Средства DAC и RBAC управляют доступом для субъектов и объектов, основываясь на их атрибутах безопасности и операциях, а также настраивают идентификацию, аутентификацию, права доступа и регистрацию событий безопасности. DAC использует биты разрешений и списки контроля доступа, применяя утилиты вроде useradd, usermod, и groupmod, а также графический интерфейс «Пользователи» для управления атрибутами пользователей. RBAC, основанный на SELinux, использует утилиты вроде semanage для настройки ролей и контекста безопасности, а также обеспечивает защиту на уровне системы через утилиты getenforce, setenforce, sestatus и другие.

Администраторы и владельцы файлов могут управлять атрибутами безопасности и правами доступа к объектам с помощью DAC и RBAC. DAC использует утилиты chown, chmod и setfacl, в то время как RBAC использует утилиты для работы с SELinux, такие как seinfo и sesearch, для задания разрешений, и audit2allow для генерации правил доступа. SELinux также контролирует доступ через ограничения по срокам действия учетных записей и паролей, настройку предельных значений по времени бездействия и параллельным сеансам, редактируя файлы конфигурации, такие как /etc/security/limits.conf и /etc/pam.d/system-auth.

Средства RBAC регулируют права записи в системные каталоги и модификацию системных компонентов для защиты ОС от сбоев. Утилиты lsof и top позволяют отслеживать используемые процессы и их ресурсы, а права доступа к ресурсам контролируются с помощью утилит renice, quota и repquota. Параметры бездействия и блокировки сессий можно настраивать через gsettings и

редактирование PAM-файлов, а защита данных обеспечивается системными вызовами, такими как fcntl, с помощью блокировок на уровне файловой системы.

4.2.6.3. Регистрация событий безопасности

Средства аудита безопасности в ОО позволяют регистрировать события безопасности, настраивать их типы и содержимое, выбирать параметры реагирования на сбои, а также управлять просмотром записей. Каждый системный вызов, связанный с безопасностью, прерывается на входе или выходе, что позволяет оценить действие и записать его в журнал аудита. Демон auditd управляет отправкой данных в журнал аудита через библиотеку libaudit, а его конфигурация задается в файле auditd.conf. Для управления правилами регистрации событий используется программа auditctl и файлы конфигурации, такие как audit.rules.

Администратор может использовать утилиты aureport и ausearch для анализа данных аудита, устанавливать правила через файлы /etc/rsyslog.conf и /etc/rsyslog.d, а также отправлять журналы в централизованное хранилище с помощью плагина audisp-remote. Доступ к записям аудита ограничен правилами контроля доступа, а записи защищены от несанкционированного изменения. Кроме того, служба journald протоколирует системные сообщения и обеспечивает удобный поиск событий, дополняя возможности rsyslogd.

4.2.6.4. Ограничение программной среды

В системе ОС предусмотрена возможность настройки базовых конфигураций и дополнений в зависимости от роли устройства и условий эксплуатации с помощью программы Anaconda. Уполномоченные субъекты устанавливают ПО с использованием утилит dnf и rpm, с ограничениями, налагаемыми средствами управления доступом. Служба systemd, управляемая через утилиту systemctl, отвечает за автоматический запуск компонентов при загрузке ОС, а средства управления доступом регулируют разрешение и запрет на запуск компонентов во время работы системы.

Механизм IMA (Integrity Measurement Architecture) обеспечивает контроль целостности файловой системы, собирая и сравнивая хеш-образы файлов для предотвращения запуска неавторизованных файлов. Защита от переполнения буфера осуществляется через ExecShield, поддерживающий технологию No eXecute для сегментации и защиты исполняемой памяти.

4.2.6.5. Изоляция процессов

Каждый процесс в ОС выполняется в изолированном адресном пространстве, разделение которого обеспечивает ядро с помощью регионов памяти, отслеживаемых через дескриптор mm_struct. Для создания виртуального адресного пространства применяются системные вызовы, такие как fork, vfork и clone,

которые предотвращают прямой доступ других процессов к выделенной виртуальной памяти.

Процессы получают случайное расположение стека и областей памяти, управляемое через /proc/sys/kernel/randomize_va_space и ExecShield, который рандомизирует адресное пространство для системного вызова mmap(). При попытке удаления занятого файла системные вызовы unlink, unlinkat и rmdir блокируют действие и возвращают ошибку EBUSY, что исключает удаление.

4.2.6.6. Защита памяти

Недоступность прежнего содержимого ресурсов при их распределении и освобождении в ядре обеспечивается с помощью функций get_zeroed_page(). Для управления несмежными областями памяти используется алгоритм Buddy System, схема Slab allocator и функция vmalloc(). Защита от выполнения произвольного кода благодаря переполнению буфера реализуется через бит NX, запрещающий выполнение кода в областях памяти, предназначенных только для данных. Управление очисткой кэшей осуществляется через /proc/sys/vm/drop_caches, а уничтожение файлов и каталогов — утилитами sdel, sdmem, sswap, sfill, dd, shred и scrub, которые многократно перезаписывают данные специальными битовыми последовательностями.

4.2.6.7. Контроль целостности

Контроль целостности компонентов ПО осуществляется через настройки системного журналирования syslogd и syslog.conf, а также утилиты coreutils. Автоматизация проверки целостности организована через cron и crontab, а при нарушениях запуска компонентов пользователю выдается сообщение о блокировке. Фиксация нарушений в журнале безопасности и уведомление администратора выполняются средствами регистрации событий. Самотестирование пакета программ производится утилитой rbac-self-test-helper при запуске и по запросу — через rbac-self-test, используя утилиты AIDE, его автоматизация также настраивается через cron и crontab.

4.2.6.8. Обеспечение надежного функционирования

Средства возврата операционной системы к безопасному состоянию при сбоях обеспечивают ручное и автоматизированное восстановление объектов с атрибутами безопасности из резервных копий, созданных утилитами tar, cpio и rsync, которые запускаются через cron и crontab. Для повышения отказоустойчивости и обеспечения доступности информации используются программы Corosync и Pacemaker, управляемые утилитой pcs. Проверка и исправление ошибок в файловой системе при переходе в аварийный режим выполняется утилитой fsck, а опции монтирования файловых систем описаны в

/etc/fstab. Настройка надежных временных меток выполняется с помощью hwclock, а данные о времени доступа и модификации файла можно определить командой stat.

Система управления приоритетами процессов на основе значения nice распределяет процессорное время между процессами в диапазоне от -20 до +19. Пользователи могут устанавливать приоритеты с помощью команд nice и renice, а значения по умолчанию для приоритетов настраиваются в /etc/security/limits.conf с помощью pam_limits. Ограничение использования дискового пространства настраивается через интерфейсы /etc/fstab и команды quota, edquota, quotacheck, quotaon и quotaoff.

4.2.6.9. Фильтрация сетевого потока

Фильтрация сетевых потоков к узлам системы осуществляется на канальном уровне с помощью ebtables, а на сетевом уровне — с использованием Netfilter, Firewalld и Iptables. Команды ping и ping6 проверяют связь и настраивают конфигурацию для IPv4 и IPv6. Администратор настраивает фильтрацию на основе атрибутов безопасности и цепочек правил для обработки сетевых пакетов, отклоняя запросы с некорректной адресацией, протокольной информацией и недопустимыми значениями, что запрещает нежелательный сетевой поток.

4.2.7. Обеспечение сетевого взаимодействия

Для управления сетевыми интерфейсами, их настройками и конфигурациями используется NetworkManager, который позволяет администрировать как проводные, так и беспроводные подключения, обеспечивая управление через командную строку и графический интерфейс. NetworkManager предоставляет команду nmcli для управления сетевыми подключениями через терминал, а также текстовый интерфейс nmtui для конфигурации сетевых подключений, который удобно использовать на серверах без графического окружения.

ОС поддерживает как статические, так и динамические (DHCP) IP-настройки. Конфигурационные файлы для сетевых интерфейсов обычно находятся в каталоге /etc/sysconfig/network-scripts/. Конфигурация сетевых интерфейсов может быть расширена для работы с протоколом IPv6. Это позволяет решать задачи по развертыванию сетей с поддержкой как IPv4, так и IPv6.

4.2.8. Поддержка виртуализации

ОС включает в свой состав ядро с поддержкой технологии виртуализации и предоставляет возможность создания и защиты среды виртуализации с обеспечением выполнения следующих функций безопасности:

– доверенная загрузка виртуальных машин;

– контроль целостности;

– регистрация событий безопасности;

– управление доступом;

– резервное копирование;

– управление потоками информации;

– защита памяти;

– ограничение программной среды;

– идентификация и аутентификация пользователей;

– централизованное управление образами виртуальных машин и виртуальными машинами.

Ядро ОС поддерживает технологию KVM (Kernel-based Virtual Machine), обеспечивающую создание и функционирование виртуальной инфраструктуры. Технология KVM включает в себя специальный модуль ядра KVM и средство создания виртуального программно-аппаратного окружения QEMU.

Управление средой виртуализации обеспечивается утилитой virsh с использованием программного интерфейса libvirt.

Libvirt — программное обеспечение сервера виртуализации, которое обеспечивает способ управления виртуальными машинами и другими функциями виртуализации, такими как управление хранилищем и сетевым интерфейсом, доступ к которому также ограничивается в соответствии с установленной в ОС политикой разграничения доступа.

Утилита virsh предназначена для управления гостевыми системами и гипервизором, использует программный интерфейс сервера виртуализации libvirt и служит альтернативой графическому менеджеру виртуальных машин. Непривилегированные пользователи могут выполнять доступ только в режиме чтения. С помощью virsh можно исполнять сценарии для виртуальных машин.

Поддержка функционирования виртуальной машины в режиме запрета модификации ее файлов-образов осуществляется специальными способами запуска виртуальной машины, реализованными в ОС, при которых основной файл-образ защищается от записи. В зависимости от выбранного режима используется создание физической копии или различные варианты создания снимков файл-образов с последующим их удалением после завершения работы виртуальной машины.

Управление доступом внутри гостевой операционной системы реализуется встроенными средствами защиты информации из состава операционной системы или сертифицированными наложенными средствами защиты информации (в случае использования в качестве гостевой операционной системы несертифицированную по требованиям безопасности информации операционную систему).

ОС обеспечивает функционирование виртуальных машин (виртуальной инфраструктуры) в условиях мандатного и дискреционного управления доступом при межпроцессном и сетевом взаимодействии, включая взаимодействие между виртуальными машинами по сетевому протоколу IPv4 (IPv6) в условиях мандатного управления доступом и доступ субъектов к файлам-образам и экземплярам функционирующих виртуальных машин.

4.2.9. Поддержка контейнеризации

ОС содержит программные средства (средства контейнеризации), реализующие создание и функционирование изолированных программных сред, с обеспечением выполнения следующих функций безопасности:

– изоляция контейнеров;
– выявление уязвимостей в образах контейнеров;
– проверка корректности конфигурации контейнеров;
– контроль целостности контейнеров и их образов;
– регистрация событий безопасности;
– идентификация и аутентификация пользователей.

Средства контейнеризации реализуют функциональные возможности по созданию образов контейнеров, формированию среды выполнения контейнеров и обеспечения выполнения их процессов, запуску контейнера и управление данным контейнером.

В состав ОС входит программное обеспечение Docker для автоматизации развертывания и управления приложениями в средах с поддержкой контейнеризации.

5. ВХОДНЫЕ И ВЫХОДНЫЕ ДАННЫЕ

5.1. Входные данные

Для операционной системы входными данными являются:

– пользовательский ввод - данные, вводимые пользователем с помощью клавиатуры, мыши, сенсорного экрана и других устройств ввода;

– файловая система - любые данные, поступающие из файлов, хранящихся на диске, включая конфигурационные файлы и другие документы;

– сетевые данные - информация, поступающая через сетевые интерфейсы, например, из интернета или локальной сети;

– устройства ввода-вывода - данные, получаемые от периферийных устройств, таких как сканеры, камеры и микрофоны;

– системные события - информация от системных таймеров, прерываний, сигналов и других событий на аппаратном уровне;

– процессы и службы - данные, поступающие от запущенных процессов и системных сервисов, которые могут взаимодействовать с ОС.

Эти входные данные обрабатываются операционной системой для обеспечения функционирования программного обеспечения и управлением оборудованием.

5.2. Выходные данные

Для операционной системы выходными данными являются результаты, которые она формирует и передает пользователю или другим системам после обработки входных данных. К выходным данным операционной относятся:

– пользовательский интерфейс - графические элементы (окна, меню, значки) или текстовые сообщения, отображающиеся на экране;

– отчеты и уведомления - системные сообщения, логи, уведомления об ошибках или успешном выполнении операций:

– контроль периферийных устройств - сигналы для управления устройствами, такими как принтеры, динамики или сетевые адаптеры;

– результаты вычислений и операции с файлами - данные, сохраняемые в файлы или передаваемые другим приложениям после выполнения процессов или программ;

– системные ресурсы и их состояние - информация о загруженности процессора, использовании памяти и другие данные о состоянии системы, которые могут быть отображены в диспетчере задач или аналогичных приложениях.

6. ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

Изм.	Номера страниц			Номера извещений	Подпись	Дата внесения изм.	Дата введения изм.