ОПЕРАЦИОННАЯ СИСТЕМА СПЕЦИАЛЬНОГО НАЗНАЧЕНИЯ «ASTRA LINUX SPECIAL EDITION»

РУСБ.10152-02

Руководство по КСЗ. Часть 1

Оперативное обновление 4.7.3

Бюллетень № 2022-1121SE47

Листов 44

2022

АННОТАЦИЯ

В настоящем руководстве приводятся изменения в документ РУСБ.10152-02 97 01-1 «Операционная система специального назначения «Astra Linux Special Edition». Руководство по КСЗ. Часть 1» из комплектности изделия РУСБ.10152-02 «Операционная система специального назначения «Astra Linux Special Edition» (далее по тексту — ОС), которые необходимо учитывать при настройке и эксплуатации ОС с установленным оперативным обновлением согласно бюллетеню № 2022-1121SE47.

Руководство предназначено для администраторов безопасности.

СОДЕРЖАНИЕ

1. ОБЩИЕ СВЕДЕНИЯ

В настоящем руководстве приведены кумулятивные изменения в документ РУСБ.10152-02 97 01-1: измененные разделы, подразделы и пункты документа, а также добавленные разделы, подразделы и пункты.

При администрировании комплекса средств защиты ОС с установленным оперативным обновлением согласно бюллетеню № 2022-1121SE47 рекомендуется руководствоваться документом РУСБ.10152-02 97 01-1 совместно с настоящим руководством.

2. ПЕРЕЧЕНЬ ИЗМЕНЕНИЙ

2.1. Подраздел «4.1. Общие сведения»

Первый абзац подраздела 4.1 изложить в редакции:

4.1. Общие сведения

Механизмы мандатного управления доступом и мандатного контроля целостности реализованы в ядре ОС и затрагивают следующие подсистемы:

- механизмы IPC;
- стек TCP/IP (IPv4, IPv6);
- ФС ext2/ext3/ext4/xfs;
- сетевые ФС CIFS, OCFS2, Ceph;
- ФС proc, tmpfs.

2.2. Подраздел «4.3. Мандатный контроль целостности»

Подраздел 4.3 изложить в новой редакции и добавить пункт 4.3.2:

4.3. Мандатный контроль целостности

При реализации политики мандатного контроля целостности субъектам и сущностям задаются уровни целостности — совокупность (декартово произведение) неиерархических уровней (категорий) целостности и иерархических (линейных) уровней целостности, описание которых приведено в 4.3.1.

Также сущностям могут быть присвоены дополнительные атрибуты для мандатного контроля целостности, описание которых приведено в 4.3.2.

Для администрирования подсистемы мандатного контроля целостности множество Linux привилегий расширено специальными привилегиями, полное описание которых приведено в 4.7.

4.3.1. Уровень целостности

Уровень целостности сущности отражает степень уверенности в целостности содержащейся в ней информации. Уровень целостности субъекта соответствует его полномочиям по доступу к сущности в зависимости от их уровней целостности, а также отражает степень уверенности в корректности его функциональности.

При реализации политики мандатного контроля целостности субъектам и сущностям задаются уровни целостности — совокупность (декартово произведение) неиерархических уровней (категорий) целостности и иерархических (линейных) уровней целостности. Процесс при его непосредственном запуске наследует уровень целостности процесса-родителя.

В стандартной реализации иерархический (линейный) уровень целостности в ОС зарезервирован и не поддерживается его использование.

Неиерархический уровень целостности представляет собой 32-битную маску (технически реализован как беззнаковая величина `uint32_t`).

При установке ОС по умолчанию предлагается максимальным неиерархический уровень целостности `max_ilev`, равный 63 (битовая маска 00111111), а минимальный уровень всегда 0.

В ОС по умолчанию выделены нулевой, четыре ненулевых и несравнимых между собой (далее — изолированных) неиерархических уровня целостности и максимальный уровень целостности, который не меньше всех остальных в системе.

Непривилегированным пользователям по умолчанию присваивается нулевой уровень целостности, администратору присваивается максимальный уровень целостности 63, за системными службами, перечень и описание которых приведены в таблице 7, зарезервированы четыре изолированных уровня целостности.

Таблица 7

Уровень	Значение	Битовая маска	Описание
1	001	0000 0001	Уровень задействован для сетевых служб
2	002	0000 0010	Уровень задействован для виртуализации
3	004	0000 0100	Уровень задействован для специального ПО
4	008	0000 1000	Уровень задействован для графического сервера

Примечание. В текущей реализации, с учетом 32-битной маски, количество изолированных уровней целостности может быть увеличено до 32 при повышении максимального уровня целостности до 0xFFFF FFFF.

После установки ОС максимальный уровень целостности в системе может быть повышен. Максимальными уровнями целостности в системе могут быть числа, у которых битовая маска включает битовые маски всех остальных используемых уровней целостности в системе, например, 63 (0x3F, битовая маска 00111111), 127 (0x7F, битовая маска 01111111), 191 (0xBF, битовая маска 10111111) и т.д.

ВНИМАНИЕ! При повышении максимального уровня целостности в ОС выше значения 63, заданного при установке ОС, необходимо убедиться в повышении уровня целостности администратора ОС.

Дополнительно зарезервировано специальное наименование уровня целостности `Высокий` (`High`), которое используется для обозначения максимального уровня целостности в установленной ОС, а также `Низкий` (`Low`) — для обозначения нулевого уровня целостности.

Числовые значения уровня целостности сущности частично сравнимы между собой и определяются как суммы значений назначенных уровней целостности. Числовые

значения уровня целостности могут принимать значения от 0 до 2^{32}-1 или от 0x0 до 0xFFFF FFFF включительно и технически реализованы как 32-битная маска, беззнаковая величина (`uint32_t`).

В пользовательских интерфейсах представляется десятичным или шестнадцатеричным числом или наименованием.

Субъект с определенным уровнем целостности может получить доступ на запись к сущности, если его уровень целостности не ниже уровня целостности сущности.

4.3.2. Атрибуты сущностей для мандатного контроля целостности

Дополнительные атрибуты для МКЦ позволяют уточнять или изменять правила МКЦ для тех или иных сущностей:

- `silev` — присваивается файлам. Процессу, запускаемому из файла с атрибутом `silev`, присваивается метка целостности, равная наибольшему значению, которое одновременно меньше или равно значениям метки целостности файла и максимальной метки целостности в системе (значение параметра командной строки ядра `parsec.max_ilev`). Например, если метка целостности файла меньше или равна максимальной метке целостности в системе, то процессу назначается метка целостности, равная метке целостности файла. Атрибут `silev` необходим для корректного запуска процесса смены пароля из файла `/usr/bin/passwd`, имеющего высокую метку целостности, пользователем с низкой меткой целостности.

  ВНИМАНИЕ! Использовать атрибут рекомендуется в исключительных случаях в соответствии с принятой политикой безопасности;

- `irelax` — присваивается каталогам. Определяет, что в каталог может осуществлять запись процесс с любой меткой целостности. Создаваемые в данном каталоге сущности наследуют метку целостности от создающего их процесса, если его метка целостности меньше или равна метке целостности данного каталога. Если метка целостности процесса выше или несравнима с меткой целостности данного каталога, то метка целостности создаваемой в нем сущности устанавливается как наибольшее значение, которое одновременно меньше или равно значениям метки целостности данного каталога и метки целостности процесса. Атрибут доступен только при включенном расширенном режиме МКЦ.

Дополнительные атрибуты сущностей для МКЦ могут быть установлены только привилегированным процессом.

2.3. Подраздел «4.4. Расширенный режим мандатного контроля целостности»

После подраздела 4.3 ввести новый подраздел 4.4 с соответствующим изменением нумерации следующих подразделов:

4.4. Расширенный режим мандатного контроля целостности

В режиме МКЦ процесс при его непосредственном запуске наследует уровень целостности процесса-родителя. При этом в расширенном режиме МКЦ (strict mode) непосредственный запуск процесса запрещен в том случае, если исполняемый файл, из которого запускается процесс, имеет уровень целостности меньше или несравнимый с уровнем целостности процесса-родителя. В данном случае процесс возможно запустить только с использованием инструмента `sumic`, описание которого приведено в 4.15.9.

Создаваемому файлу (каталогу) назначается уровень целостности, равный уровню целостности того каталога, в котором он создается. При этом запрещено создавать файл (каталог) с уровнем целостности выше или несравнимым с уровнем целостности процесса, создающего данный файл (каталог).

ВНИМАНИЕ! Расширенный режим МКЦ предназначен для усиления защиты ОС. Вместе с тем его включение может привести к сбоям в работе некоторого прикладного ПО (особенно уже установленного). Поэтому перед включением расширенного режима МКЦ в ОС администратору рекомендуется провести тестирование используемого прикладного ПО с целью проверки его работоспособности при включенном расширенном режиме МКЦ и необходимости его дополнительной настройки.

Включение расширенного режима МКЦ осуществляется путем выполнения от имени администратора команды:

```
astra-strictmode-control enable
```

В результате будут выполнены необходимые настройки уровней целостности сущностей и для параметра командной строки ядра `parsec.strict_mode` установлено значение `1`. Для активации расширенного режима МКЦ необходимо перезагрузить ОС

Для проверки текущего состояния расширенного режима МКЦ (активен/неактивен) можно воспользоваться командой:

```
astra-strictmode-control status
```

В случае, если после выполнения команды включения расширенного режима МКЦ не выполнялась перезагрузка ОС, результат команды проверки состояния режима будет `НЕАКТИВНО`. Для получения информации о состоянии расширенного режима МКЦ, которое будет после перезагрузки ОС, выполнить команду:

```
astra-strictmode-control is-enabled
```

При включенном расширенном режиме МКЦ во время создания пользователя всему содержимому его домашнего каталога назначается уровень целостности, равный уровню целостность данного пользователя. В дальнейшем назначение уровней целостности содержимому домашних каталогов пользователей осуществляется в соответствии с общими правилам мандатного контроля целостности.

При включенном расширенном режиме МКЦ применяются следующие изменения в работе ОС:

1) вход в сессию возможен только на максимально доступном пользователю неиерархическом уровне целостности (выбирается автоматически). Т.е. администратор с высоким уровнем целостности, если при входе в сессию была выбрана нулевая метка конфиденциальности, не сможет выполнить вход на нулевом уровне целостности. И наоборот, если администратором с высоким уровнем целостности при входе в сессию была выбрана ненулевая метка конфиденциальности, то вход будет выполнен на нулевом уровне целостности;

2) не применяется привилегия `PARSEC_CAP_IGNMACINT` (см. таблицу 8);

3) не применяется параметр командной строки ядра `parsec.ccnr_relax` (см. таблицу 31);

4) возможно применение привилегии `PARSEC_CAP_CCNR_RELAX` (см. таблицу 8);

5) возможно применение атрибута irelax (см. 4.3.2).

2.4. Подраздел «4.4. Мандатный контекст безопасности»

Изменить нумерацию подраздела с 4.4 на 4.5 в связи с добавлением нового подраздела и изложить в редакции:

4.5. Мандатный контекст безопасности

Мандатные атрибуты субъекта (объекта) объединяются в мандатный контекст безопасности этого субъекта (объекта).

Мандатный контекст безопасности включает в себя:

- метку безопасности;
- дополнительные мандатные атрибуты управления доступом;
- дополнительные атрибуты для мандатного контроля целостности.

Метка безопасности состоит из:

1) классификационной метки, которая определяется:
   а) иерархическим уровнем конфиденциальности;
   б) неиерархическими категориями конфиденциальности;
2) метки целостности, которая определяется:
   а) иерархическим (линейным) уровнем целостности (зарезервирован);
   б) неиерархическим уровнем (категорией) целостности.

Классификационные метки вложенных сущностей не могут превышать значения классификационной метки контейнера, их содержащего.

Примечание. В информационных системах с мандатным управлением доступом как правило применяются классификационные метки, в которых используется только четыре

уровня конфиденциальности от 0 до 3 и 64-битовая маска с различными сочетаниями категорий.

Правила принятия решения о предоставлении доступа на основе метки безопасности описаны в 4.6. Управление мандатными атрибутами осуществляется с помощью инструмента `pdpl-file` в соответствии с описанием 4.15.1.

ВНИМАНИЕ! Устанавливать для пользователя одновременно высокий уровень конфиденциальности (классификационную метку) и высокий уровень целостности не рекомендуется.

ВНИМАНИЕ! Запрещен вход в сессию с выбранными одновременно ненулевой меткой конфиденциальности и ненулевым уровнем целостности.

ВНИМАНИЕ! При включенном в системе расширенном режиме МКЦ при входе в сессию уровень целостности назначается автоматически из максимально доступного данному пользователю.

2.5. Подраздел «4.6. PARSEC-привилегии»

Изменить нумерацию подраздела с 4.6 на 4.7 в связи с добавлением нового подраздела. В таблице 8 уточнить описание привилегий `PARSEC_CAP_IGNMACINT` и `PARSEC_CAP_CAP` и добавить новую привилегию `PARSEC_CAP_CCNR_RELAX`:

Таблица 8

Привилегия Атрибут Битовая маска	Описание
`PARSEC_CAP_CAP` `0x00400`	Позволяет устанавливать любой непротиворечивый набор привилегий для вызвавшего процесса и читать привилегии, присвоенные процессам
`PARSEC_CAP_IGNMACINT` `0x02000`	Для данного процесса отключает проверку правил доступа на основе уровней целостности. При включенном расширенном режиме МКЦ привилегия не применяется
`PARSEC_CAP_CCNR_RELAX` `0x100000`	Позволяет осуществлять в каталоге с установленным атрибутом `ccnr` действия (создание, удаление и др.) над вложенными файлами и каталогами с уровнями конфиденциальности, не выше уровня конфиденциальности данного каталога

2.6. Подраздел «4.8. Включение и выключение мандатного контроля целостности»

Изменить нумерацию подраздела с 4.8 на 4.9 в связи с добавлением нового подраздела, также изменить иерархию пунктов подраздела и изложить их в редакции:

4.9. Включение и выключение мандатного контроля целостности

Включение МКЦ может быть выполнено в процессе установки ОС путем выбора пункта «Мандатный контроль целостности» в программе установки ОС.

Включение и выключение МКЦ после установки ОС выполняется с помощью инструмента `astra-mic-control`, описанного в 16.4.27, или графической утилиты `fly-admin-smc` (см. электронную справку). При включении/выключении МКЦ автоматически включается/выключается МКЦ на файловой системе (см. 4.10).

При включении МКЦ для параметра командной строки ядра `parsec.max_ilev` в загрузчике ОС устанавливается значение максимального уровня целостности в системе. По умолчанию значение максимального уровня целостности в системе 63 (если при включении МКЦ не было указано другое значение, см. 16.4.27).

ВНИМАНИЕ! Графический сервер Xorg по умолчанию работает от имени учетной записи пользователя на выделенном уровне целостности 8.

При выключении МКЦ для параметра командной строки ядра `parsec.max_ilev` устанавливается значение `0`.

4.10. Мандатный контроль целостности на файловой системе

При включении МКЦ согласно 4.9 объектам файловой системы автоматически присваиваются атрибуты МКЦ.

Присвоение атрибутов МКЦ объектам файловой системы осуществляется в соответствии с конфигурационным файлом `/etc/parsec/fs-ilev.conf`. В данном конфигурационном файле перечислены объекты файловой системы и их уровень целостности в формате:

`<уровень_целостности> <путь>`

где `<уровень_целостности>` — уровень целостности для объекта файловой системы, указанного в `<путь>`;

`<путь>` — объект/объекты файловой системы или путь к ним.

Значения, указываемые в конфигурационном файле в качестве уровня целостности `<level>`, приведены в таблице 9.

Таблица 9

Значение	Описание
`<число>`	Определенный уровень целостности, заданный числовым значением. Может быть десятичным, восьмеричным, шестнадцатеричным или двоичным числом
`high`	Текущий `max_ilev` — максимальный уровень целостности в ОС, заданный в параметре командной строки ядра `parsec.max_ilev`
`max`	Текущий `max_ilev` — максимальный уровень целостности в ОС, заданный в параметре командной строки ядра `parsec.max_ilev`
`low`	То же что и нулевой уровень целостности

Окончание таблицы 9

Значение	Описание
`min`	То же что и нулевой уровень целостности
`exc`	Игнорировать файл при проверке целостности. В качестве символа подстановки в конце пути можно использовать символ «*»

Если в файле указаны несуществующие и неабсолютные пути, то они игнорируются. Корневому каталогу («/») уровень целостности не назначается.

Пример

Конфигурационный файл `/etc/parsec/fs-ilev.conf`

```
exc     /etc/xdg/autostart/vboxclient.desktop
exc     /etc/X11/Xsession.d/98vboxadd-xclient
exc     /etc/ld.so.*
exc     /etc/resolv.conf
exc     /root/.config/*
exc     /root/.gnupg/gpg-agent.conf
max     /etc
max     /lib
max     /lib64
max     /lib32
max     /bin
max     /sbin
max     /boot
max     /root
max     /opt
max     /srv
max     /usr
```

Для управления МКЦ на файловой системе используется инструмент командной строки `set-fs-ilev`.

Пример

После установки новых пакетов, а также в процессе работы ОС могут создаваться новые файлы в каталоге `/etc/`, которым атрибуты МКЦ автоматически не присваиваются. Чтобы привести МКЦ файловой системы в соответствие конфигурационном файлу `/etc/parsec/fs-ilev.conf`, необходимо выполнить команду:

```
sudo set-fs-ilev enable
```

Подробное описание инструмента `set-fs-ilev` приведено в `man set-fs-ilev`.

Также для управления МКЦ на файловой системе может использоваться графическая утилита `fly-admin-smc` (см. электронную справку).

Выключение МКЦ на файловой системе осуществляется автоматически при выключении МКЦ согласно 4.9.

Для обеспечения совместимости в ОС сохранен устаревший инструмент выключения МКЦ на файловой системе `unset-fs-ilev`.

4.11. Администрирование ОС при включенном МКЦ

Непривилегированный пользователь может выполнять вход в систему только на низком уровне целостности (соответствует минимальному уровню целостности). Привилегированный пользователь, при наличии соответствующего права, может входить в систему на высоком уровне целостности (соответствует максимальному уровню целостности ОС) и только для выполнения задач по конфигурированию ОС.

Администратор, созданный при установке ОС, может выполнять вход в систему с высоким уровнем целостности (по умолчанию 63) или с низким уровнем целостности. При графическом входе в систему для такого администратора по умолчанию выбран высокий уровень целостности. Графический рабочий стол на высоком уровне целостности имеет красный фон.

При консольном входе в систему администратор должен вручную выставлять уровень контроля целостности (для высокого уровня — 63, для низкого — 0 или пропустить данный шаг).

ВНИМАНИЕ! Вход в систему привилегированным пользователем (администратором) необходим только для выполнения настроек системы и только с высоким уровнем целостности. Для обычного (штатного) режима работы рекомендуется осуществлять вход в систему от имени непривилегированного пользователя на низком уровне целостности.

ВНИМАНИЕ! При включенном в системе расширенном режиме МКЦ при входе в сессию уровень целостности назначается автоматически из максимально доступного данному пользователю.

2.7. Подраздел «4.9. Запуск служб systemd с уровнем целостности и конфиденциальности»

Изменить нумерацию подраздела с 4.9 на 4.12 в связи добавлением нового подраздела и изменением иерархии пунктов.

2.8. Подраздел «4.10. Сетевое взаимодействие»

Изменить нумерацию подраздела и входящих в него пунктов с 4.10 на 4.13 в связи добавлением нового подраздела и изменением иерархии пунктов.

2.9. Подраздел «4.11. Шина межпроцессного взаимодействия D-Bus»

Изменить нумерацию подраздела и входящих в него пунктов с 4.11 на 4.14 в связи добавлением нового подраздела и изменением иерархии пунктов.

2.10. Подраздел «4.12. Средства управления мандатными ПРД»

Изменить нумерацию подраздела и входящих в него пунктов с 4.12 на 4.15 в связи добавлением нового подраздела и изменением иерархии пунктов. В третьем абзаце добавить пункты перечисления:

Для управления локальными мандатными ПРД в режиме командной строки используются следующие инструменты:

- `pdp-exec` — запуск процессов в заданном окружении, описание приведено в 4.15.7;
- `sumic` — запуск процесса на пониженном (заданном) уровне целостности, описание приведено в 4.15.9;

2.11. Пункт «4.12.1. pdpl-file»

Изменить нумерацию пункта с 4.12.1 на 4.15.1 в связи добавлением нового подраздела и изменением иерархии пунктов. Пункт 4.15.1 изложить в редакции:

4.15.1 pdpl-file

Инструмент командной строки `pdpl-file` предназначен для управления мандатными атрибутами (меткой безопасности, дополнительными мандатными атрибутами управления доступом и дополнительными атрибутами для МКЦ) сущностей ОС.

Синтаксис инструмента:

```
pdpl-file [<параметр>[...]]
    [<уровень_конфиденциальности>][:<уровень_целостности>
    [:<категория_конфиденциальности>[:<дополнительный_атрибут>]]] [<сущность>]
```

Уровень и категория конфиденциальности могут быть заданы именем или шестнадцатеричным значением.

Пример

Рекурсивно для всех файлов каталога `/tmp` изменить уровень на `Секретно` и категорию на `Категория_А` (уровень и категория должны быть определены в системе):

```
pdpl-file -Rv Секретно:0:Категория_А /tmp
```

Для присвоения сущности одновременно всех категорий, которые определены в системе, можно использовать значение `-1` для `<категория_конфиденциальности>`.

Пример

```
pdpl-file 1:0:-1 /tmp
```

Дополнительные мандатные атрибуты `ccnr`, `ehole`, `whole` и дополнительные атрибуты для МКЦ `silev` и `irelax` могут быть заданы значениями или именами через запятую.

Пример

```
pdpl-file 2:0:0:ccnr /tmp
```

Описание параметров инструмента `pdpl-file` приведено в таблице 12.

Таблица 12

Параметр	Описание
`-f`, `--silent`, `--quiet`	Не выводить сообщений об ошибках
`-v`, `--verbose`	Выводить диагностические сообщения для каждого файла
`-c`, `--changes`	То же, что и `--verbose`, но сообщать только об изменениях
`-u`, `--unite`	Объединить текущую метку безопасности файла с указанной в качестве аргумента
`-s`, `--subtract`	Вычесть из текущей метки безопасности сущности метку безопасности, указанную в качестве аргумента. При этом для итоговой метки безопасности значения задаются по следующим правилам: - уровень конфиденциальности — минимальное значение из текущей метки безопасности и указанной в качестве аргумента; - уровень целостности — минимальное значение из текущей метки безопасности и указанной в качестве аргумента (должно быть указано в десятичном виде); - категории конфиденциальности — из текущей метки безопасности вычитаются категории, указанные в качестве аргумента; - дополнительные атрибуты — из текущей метки безопасности вычитаются дополнительные атрибуты, указанные в качестве аргумента
`-R`, `--recursive`	Применить рекурсивно
`-r`, `--reverse`	Сначала файлы в каталоге, потом каталог
`-h`, `--help`	Вывести справку и выйти
`--version`	Вывести информацию о версии и выйти

2.12. Пункт «4.15.7. pdp-exec»

После пункта 4.15.6 (с учетом изменения нумерации) ввести новый пункт 4.15.7 с соответствующим изменением нумерации следующих пунктов:

4.15.7. pdp-exec

Инструмент `pdp-exec` позволяет администратору запускать процессы в заданном окружении:

- имя пользователя, от имени которого запускается процесс;
- метка безопасности процесса;
- PARSEC-привилегии.

При использовании инструмента pdp-exec следует учитывать, что возможен запуск процесса без применения мандатного контекста безопасности, поэтому использование pdp-exec должно быть регламентировано и ограничено.

Синтаксис инструмента:

```
pdp-exec [параметр[параметр...]] [--] [<команда>] [<параметры_запуска_команды>]
```

В случае если с командой заданы параметры ее запуска, то указание символов «--» перед командой обязательно.

Описание параметров инструмента pdp-exec приведено в таблице 17.

Таблица 17

Параметр	Описание
-c <привилегии>, --capability=<привилегии>	Установить процессу указанные привилегии в качестве эффективных (текущих), наследуемых и разрешенных
-u <имя_пользователя>, --user=<имя_пользователя>	Запустить процесс от имени указанного пользователя
-l <метка_безопасности>, --label=<метка_безопасности>	Установить процессу указанную метку безопасности
-v, --version	Вывести информацию о версии и выйти
-h, --help	Вывести справку и выйти

Привилегии задаются в виде битовой маски (как правило, в шестнадцатеричном виде). Соответствие отдельных битов полномочиям приведено в man parsec_capset, а также в таблице 8.

Метка безопасности задается в виде:

```
[<уровень_конфиденциальности>][:<уровень_целостности>
[:<категория_конфиденциальности>]]
```

Примеры:

1. Перезапустить службу dbus с PARSEC-привилегией PARSEC_CAP_PRIV_SOCK:

```
pdp-exec -c 0x100 -- /etc/init.d/dbus restart
```

2. Запустить оболочку bash от имени пользователя secretuser с PARSEC-привилегией PARSEC_CAP_SIG и с меткой безопасности 1:1:

```
pdp-exec -c 0x40 -u secretuser -l 1:1 -- bash
```

Более подробное описание pdp-exec приведено в man pdp-exec.

2.13. Пункт «4.15.8. sumac»

Пункт 4.15.8 (с учетом изменения нумерации) изложить в редакции:

Инструмент командной sumac используется для запуска процесса с заданными уровнем и категорией конфиденциальности в отдельной графической сессии с использованием

виртуального графического сервера Xephyr. Пользователь может запускать процесс только в пределах разрешенных ему уровней и категорий.

Синтаксис инструмента:

```
sumac [<параметр>] [<команда>]
```

ВНИМАНИЕ! Для запуска процесса на уровне, отличном от уровня текущей сессии, не должна стоять блокировка использования `sumac` в графической утилите `fly-admin-smc` (см. электронную справку) и пользователю должна быть назначена привилегия `PARSEC_CAP_SUMAC` (см. 4.7).

ВНИМАНИЕ! Для использования `sumac` при включенном МКЦ также должно быть включено МКЦ на файловой системе.

Если указанный уровень конфиденциальности выше текущего, т. е. происходит увеличение уровня, то переменные окружения наследуются от текущего процесса. Текущие переменные окружения сбрасываются, чтобы избежать утечки информации. Также при порождении нового процесса закрываются все файловые дескрипторы, метка безопасности которых не совпадает с указанной в командной строке. В том числе закрываются `stdin`, `stdout`, `stderr`. Перенаправить стандартный ввод и вывод для нового процесса можно с помощью параметров `-i`, `-o` и `-e` для `stdin`, `stdout` и `stderr`, соответственно.

ВНИМАНИЕ! Запуск процесса с понижением уровня конфиденциальности или с сокращением набора категорий конфиденциальности запрещен для предотвращения утечки информации на более низкие уровни конфиденциальности.

Далее по тексту...

2.14. Пункт «4.15.8. sumic»

После пункта 4.15.8 (с учетом изменения нумерации) ввести новый пункт 4.15.9 с соответствующим изменением нумерации следующих пунктов:

4.15.9. sumic

Инструмент `sumic` позволяет запускать процессы на уровне целостности ниже, чем уровень целостности процесса-родителя. При этом запускаемый с использованием `sumic` процесс будет иметь уровень целостности не выше уровня целостности исполняемого файла, из которого он запущен. При запуске процесса с помощью `sumic` запрещается наследование открытых в процессе-родителе ресурсов, имеющих уровень целостности, превышающий уровень целостности создаваемого процесса. Запуск графической утилиты с использованием `sumic` выполняется в изолированном X-сервере.

Синтаксис инструмента:

```
sumic [параметр] [--] [<команда>] [<параметры_запуска_команды>]
```

Описание параметров инструмента `sumic` приведено в таблице 18.

Таблица 18

Параметр	Описание
`-i <уровень_целостности>`	Уровень целостности, на котором будет запущен процесс указанной команды. В случае отсутствия параметра процесс будет запущен на нулевом уровне целостности
`-v, --version`	Вывести информацию о версии и выйти
`-h, --help`	Вывести справку и выйти

2.15. Пункт «4.13.2. execaps»

Изменить нумерацию пункта с 4.13.2 на 4.16.2 в связи добавлением нового подраздела и изменением иерархии пунктов. В пункте пример и абзац после него изложить в редакции:

Пример

```
echo 1 | sudo tee /parsecfs/unsecure_setxattr
sudo execaps -c 0x1000 -- tar --xattrs
     --xattrs-include=security.{PDPL,AUDIT,DEF_AUDIT} --acls -xzf
     backup.tar.gz -C /
echo 0 | sudo tee /parsecfs/unsecure_setxattr

echo 1 | sudo tee /parsecfs/unsecure_setxattr
sudo execaps -c 0x1000 -- sudo rsync -a --xattrs --acls /backup/ /
echo 0 | sudo tee /parsecfs/unsecure_setxattr
```

Будет запущен процесс восстановления из резервной копии с установленной привилегией `PARSEC_CAP_UNSAFE_SETXATTR`.

2.16. Пункт «4.13.3. pscaps»

Изменить нумерацию пункта с 4.13.3 на 4.16.3 в связи добавлением нового подраздела и изменением иерархии пунктов. Пункт изложить в редакции:

4.16.3. pscaps

Синтаксис:

```
pscaps <pid> [-v, --version] [-h, --help] [<действующие_полномочия>
     [<разрешенные_полномочия> [<наследуемые_полномочия>]]]
```

Если в качестве аргумента указан только идентификатор процесса `pid`, то команда `pscaps` показывает набор PARSEC-привилегий (указанных в виде битовых масок привилегий) процесса.

При указании с командой битовых масок привилегий (в десятичном или шестнадцатеричном виде) будут изменены привилегии процесса `pscaps`. В этом случае в качестве значения `pid` должно быть указано «0» или идентификатор процесса `pscaps`.

Описание параметров приведено в таблице 29.

Таблица 29

Параметр	Описание
`-h, --help`	Вывести справку и выйти
`-v, --version`	Вывести информацию о версии и выйти

2.17. Пункт «4.16. Настройка загрузчика GRUB 2»

Изменить нумерацию подраздела с 4.16 на 4.19 в связи добавлением новых подразделов. В таблице описания параметров добавить новую строку с параметром `parsec.enable_exec_on_fuse`:

В загрузчике GRUB 2 возможно задать параметры командной строки ядра PARSEC, приведенные в таблице 31.

Таблица 31

Параметр	Описание
`parsec.enable_exec_on_fuse`	Разрешение запуска сценариев и исполняемых файлов с файловых систем, смонтированных с помощью файловой системы FUSE. Допустимые значения `0/1`. Значение по умолчанию `0` (запуск запрещен)

2.18. Раздел «6. Регистрация событий безопасности»

В разделе 6 изложить в новой редакции текст между заголовком раздела 6 и заголовком подраздела 6.1, а также подраздел 6.1.

Добавить новый подраздел после подраздела 6.2 с соответствующим изменением нумерации следующих подразделов.

Изменить порядок и иерархию подразделов и пунктов, отдельные пункты изложить в новой редакции, изменить наименование заголовков.

6. Регистрация событий безопасности

В ОС регистрация событий безопасности реализуется использованием службы `auditd`.

Служба `auditd` выполняет регистрацию событий объектов файловой системы (аудит файлов) и пользователей (аудит процессов) согласно заданным правилам. Работа с

правилами аудита описана в 6.1 и 6.2. Регистрация событий осуществляется в соответствии с 6.3). Описание настройки параметров аудита приведено в 6.4.

Применение настроенных параметров аудита процессов осуществляется PAM-модулем `pam_parsec_aud`. По умолчанию регистрация событий аудита процессов включена в PAM-сценарии: `fly-dm`, `fly-dm-np`, `login`, `su`, `sshd`, `sumac.xauth`. Для регистрации событий аудита процессов пользователя, проходящего аутентификацию через другие PAM-сценарии, необходимо включить в соответствующие сценарии строку следующего вида:

```
session required pam_parsec_aud.so
```

В библиотеках подсистемы безопасности PARSEC реализован программный интерфейс для регистрации событий с использованием службы регистрации событий безопасности ОС, применяемый для регистрации событий в СУБД PostgreSQL (описано в 6.5) и комплексе программ электронной почты.

Информация о событиях от службы `auditd` также обрабатывается подсистемой регистрации событий, описание подсистемы регистрации событий приведено в РУСБ.10152-02 95 01-1.

6.1. Правила регистрации событий

Регистрация событий осуществляется в соответствии с правилами аудита, правила делятся на два типа:

1) временные — действуют до перезагрузки системы. Такие правила задаются посредством инструмента `auditctl`;

2) постоянные — действуют всегда, даже после перезагрузки системы. Такие правила задаются в файлах формата `*.rules`, располагающихся в каталоге `/etc/audit/rules.d/`.

Подробное описание правил аудита, а также синтаксис использования инструмента `auditctl` приведены в справочной странице `man auditctl`.

Примеры:

1. Записывать все системные вызовы от процесса с идентификатором (PID) `1005`

```
auditctl -a exit,always -S all -F pid=1005
```

2. Записывать все файлы, открытые пользователем с идентификатором `auid 510`

```
auditctl -a exit,always -S open -F auid=510
```

При добавлении постоянных правил аудита в файлах используется синтаксис инструмента `auditctl` без указания имени инструмента.

Пример

Записывать все системные вызовы от процесса с идентификатором (PID) `1005`

```
-a exit,always -S all -F pid=1005
```

Правила, необходимые для работы встроенного аудита, заданы в файле /etc/audit/rules.d/10-parsec.rules:

- аудит процессов (данное правило необходимо для работы утилит useraud и psaud):

```
-a always,exit -F subj_type=psaud -S all -k parsec-p
```

где -a always,exit — записывать события в журнал, добавить правило в список exit (события, происходящие при выходе из системного вызова);

-F subj_type=psaud — обрабатывать правила, заданные утилитами useraud и psaud;

-S all — перехватывать события при любых системных вызовах;

-k parsec-p — присвоить ключ фильтрации parsec-p событиям по данному правилу;

- аудит файлов (данное правило необходимо для работы команд getfaud и setfaud):

```
-a always,exit -F obj_type=faud -S all -k parsec-f
```

где -a always,exit — записывать события в журнал, добавить правило в список exit (события, происходящие при выходе из системного вызова);

-F obj_type=faud — обрабатывать правила, заданные командой setfaud;

-S all — перехватывать события при любых системных вызовах;

-k parsec-f — присвоить ключ фильтрации parsec-f событиям по данному правилу.

В дополнение к этим правилам можно задавать собственные правила аудита.

Постоянные правила рекомендуется добавлять в файл audit.rules, расположенный в каталоге /etc/audit/rules.d/. При желании можно создать в данном каталоге новый файл формата *.rules с произвольным именем и задать в нем нужные правила. Файл audit.rules можно редактировать вручную или с помощью графической утилиты system-config-audit (см. 6.4). Другие файлы правил можно редактировать только вручную.

6.2. Регистрация событий на основе меток безопасности

Без изменений.

6.3. Журнал аудита и журнал событий

Служба auditd регистрирует события безопасности в журнале аудита /var/log/audit.

Для просмотра журнала аудита используется графическая утилита ksystemlog («Системный журнал»), описание утилиты приведено в электронной справке.

Информация о событиях аудита, обработанная подсистемой регистрации событий (см. РУСБ.10152-02 95 01-1), записывается в журнал событий `/parsec/log/astra/events`.

Журнал событий `/parsec/log/astra/events` доступен только администратору. На файле журнала событий установлены атрибуты, разрешающие только доступ на чтение и на добавление в конец файла новых строк.

При включенном МКЦ файл журнала событий имеет высокий уровень целостности и процесс, который осуществляет регистрацию событий в файле журнала, также имеет высокий уровень целостности. Таким образом, изменение файла журнала событий доступно только администратору на высоком уровне целостности.

Для просмотра журнала событий используется графическая утилита `fly-event-viewer` («Журнал системных событий»), описание утилиты приведено в электронной справке.

Действия с журналом аудита службы `auditd` (удаление, переименование, перемещение файла журнала аудита) регистрируются подсистемой регистрации событий и указываются в журнале событий.

Действия с журналом событий (удаление, переименование, перемещение, ротация файла журнала событий) регистрируются подсистемой регистрации событий и указываются первой записью в журнале событий, а также регистрируются службой `auditd` и указываются в журнале аудита.

Дополнительно подсистемой регистрации событий регистрируются события записи (изменения и удаления) в файл журнала событий недоверенными процессами (всеми процессами, кроме `syslog-ng` и `logrotate`).

6.4. Средства управления регистрацией событий

6.4.1. Графические утилиты

Для управления регистрацией событий и просмотра журналов могут использоваться следующие графические утилиты (описание утилит доступно в электронной справке):

- `fly-admin-smc` («Управление политикой безопасности») — управление аудитом, привилегиями и мандатными атрибутами пользователей, работа с пользователями и группами;
- `system-config-audit` («Конфигурация аудита») — включение и выключение регистрации событий, настройка службы `auditd`, настройка журнала аудита, а также добавление, удаление и редактирование правил аудита;
- `fly-admin-events` («Настройка регистрации системных событий») — утилита из состава подсистемы регистрации событий (см. РУСБ.10152-02 95 01-1), в которой доступно управление регистрацией событий запуска и остановки службы `auditd`,

добавления и удаления правил auditd, регистрация действий с журналом аудита, а также возможно добавление правил аудита.

6.4.2. getfaud

Без изменений.

6.4.3. setfaud

Без изменений.

6.4.4. useraud

Без изменений.

6.4.5. psaud

Без изменений.

6.4.6. ausearch

Без изменений.

6.4.7. Дополнительные параметры регистрации событий

Без изменений.

6.5. Регистрация событий в СУБД PostgreSQL

6.5.1. Режимы регистрации событий

В СУБД PostgreSQL для настройки режима регистрации событий используется конфигурационный параметр ac_audit_mode файла postgresql.conf. Этот параметр может быть изменен только перезапуском сервера. Параметр может принимать следующие значения:

- internal — для настройки регистрации событий используются соответствующие команды SQL, а настройки хранятся в таблице pg_db_role_settings;
- external — для настройки регистрации событий используется внешний файл pg_audit.conf;
- external,internal — смешанный режим. Настройки регистрации событий берутся сначала из внешнего файла pg_audit.conf, после чего дополняются настройками из таблицы pg_db_role_settings;
- internal,external — смешанный режим. Настройки регистрации событий берутся сначала из таблицы pg_db_role_settings, после чего дополняются настройками из внешнего файла pg_audit.conf;
- none — регистрация событий отключена.

6.5.2. Настройка маски регистрации событий

Без изменений.

6.5.3. Назначение списков регистрации событий в режиме internal

Без изменений.

6.5.4. Назначение списков регистрации событий в режиме external

Для назначения маски событий в режиме external используется конфигурационный файл pg_audit.conf конкретного кластера данных, который имеет следующий формат:

```
success events mask = значение failure events mask = значение user =
имя_пользователя database = имя_базы_данных
success events mask = значение failure events mask = значение user =
имя_пользователя
success events mask = значение failure events mask = значение
```

Пример

Файл pg_audit.conf

- аудит действий администратора СУБД:

```
success events mask = F00E7 failure events mask = 0 user = postgres
```

- для пользователя any выполнять регистрацию только неуспешных действий:

```
success events mask = 0 failure events mask = FFFFF user = any
```

- для всех остальных пользователей выполнять регистрацию всех неуспешных действий и всех успешных действий, кроме доступа к данным:

```
success events mask = F0707 failure events mask = FFFFF
```

В конфигурационном файле задаются списки успешных (success events mask) и неуспешных (failure events mask) типов запросов на доступ, которые будут регистрироваться в журнале СУБД и журнале аудита ОС для отдельных пользователей и по умолчанию. Списки типов запросов на доступ задаются в виде шестнадцатеричных чисел, в которых каждому типу запроса соответствует установленный (для регистрируемых запросов) или сброшенный (для не регистрируемых запросов) бит. Типы запросов и их описание приведены в таблице 38.

Таблица 38

Тип запроса	Описание	Бит	Шестнадцатеричное значение
SUBJECT	Добавление/изменение/удаление пользователей и групп	0	1
CONFIGURATION	Изменение конфигурации, влияющей на доступ к данным (запрос на изменение значения переменной ac_session_maclabel)	1	2
RIGHTS	Изменение прав доступа к объектам БД	2	4
CHECK_RIGHTS	Модификация прав доступа к объектам БД	3	8
SELECT	Выборка информации из БД	4	10
INSERT	Добавление информации в БД	5	20
UPDATE	Изменение информации в БД	6	40
DELETE	Удаление информации из БД	7	80

Окончание таблицы 38

Тип запроса	Описание	Бит	Шестнадцатеричное значение
TRUNCATE	Очистка данных	8	100
REFERENCES	Задание столбца таблицы в качестве внешнего ключа	10	400
TRIGGER	Добавление триггера к таблице	11	800
EXECUTE	Запуск хранимой процедуры или триггера	12	1000
USAGE	Использование объекта БД	13	2000
CREATE	Создание объектов в БД	16	10000
CREATE_TEMP	Создание временных объектов в БД	17	20000
DROP	Удаление объектов БД	18	40000
ALTER	Изменение объекта БД	19	80000
CONNECT	Соединение пользователя с БД	30	40000000
DISCONNECT	Разъединение пользователя с БД	31	80000000

Информация о соединении пользователей с БД (CONNECT) и разъединении с ней (DISCONNECT) регистрируется всегда, при условии, что список событий не установлен в 0.

Примечание. Любые изменения этого файла будут применены только при перезапуске сервера.

6.5.5. Назначение списков регистрации событий в режимах external, internal и internal,external

Без изменений.

6.5.6. Назначение списков регистрации событий в режиме none

Без изменений.

6.6. Средства централизованного аудита и протоколирования

Без изменений.

2.19. Подраздел «7.3. Работа с Docker в непривилегированном режиме с ненулевыми метками безопасности»

Ввести новый подраздел 7.3:

7.3. Работа Docker в непривилегированном режиме с ненулевыми метками безопасности

7.3.1. Принцип функционирования

Контейнеры Docker в непривилегированном (rootless) режиме (описание режима приведено в РУСБ.10152-02 95 01-1) могут быть запущены от имени любого непривилегированного пользователя с ненулевой меткой безопасности контейнера.

Метка безопасности контейнера всегда задается четырьмя десятичными неотрицательными числами, разделенными двоеточием:

```
0:63:0:0
```

где первое число — иерархический уровень конфиденциальности;

второе число — иерархический уровень целостности;

третье число — неиерархические категории конфиденциальности;

четвертое число — зарезервировано для задания флагов МРД для файловых объектов, и в работе с контейнерами не применяется (следует всегда использовать значение `0`).

Для контейнера в непривилегированном режиме с ненулевой меткой безопасности применяются следующие ограничения:

1) запуск процессов внутри контейнера возможен только с одинаковой для всех процессов меткой безопасности, равной метке безопасности контейнера (задается при запуске контейнера);

2) в метке безопасности контейнера ненулевой может быть либо только классификационная метка, либо только метка целостности (при этом отрицательная метка целостности также является ненулевой меткой);

3) для системных файлов внутри контейнера (исполняемых, конфигурационных и т.д.) всегда используется ненулевой уровень целостности (например, `0:2:0:0` или `0:63:0:0`), и, соответственно, всегда используется нулевая классификационная метка;

4) внутри контейнера, запущенного в непривилегированном режиме, API PARSEC не работает;

5) если внутри контейнера необходима работа с файлами с ненулевой классификационной меткой, то данные файлы должны группироваться в специально созданных каталогах:

```
/home/.pdp/<имя_пользователя>/<уровень_конфиденциальности>:
    <иерархический_уровень_целостности>:<категории_конфиденциальности>:0
```

При этом:

а) метки безопасности таких каталогов могут сочетать ненулевые классификационные метки и ненулевые метки целостности;

б) при работе в расширенном режиме МКЦ (strict mode) метки целостности создаваемых внутри этих каталогов файловых объектов наследуют значение метки целостности родительского каталога.

Для хостовой ОС контейнер, запущенный в непривилегированном режиме, является группой процессов:

1) имеющих метку безопасности субъекта, запустившего контейнер;

2) не имеющих прав администратора;

3) работающих с набором файловых объектов, расположенных в файловой системе контейнера и имеющих собственные метки безопасности.

При этом процессы внутри контейнера:

1) запущены от имени администратора;

2) имеют неограниченный доступ к объектам файловой системы контейнера (за исключением возможности изменять метки безопасности).

Файловые объекты в файловой системе контейнера (rootFS) создаются с меткой безопасности, в которой:

1) классификационная метка равна классификационной метке субъекта, запустившего контейнер;

2) метка целостности всегда нулевая.

При этом к процессам (субъектам) внутри контейнера, запущенного в непривилегированном режиме, будут применяться общие правила МРД и МКЦ хостовой ОС. Например, процессы контейнера в непривилегированном режиме, имеющего метку безопасности `1:0:0:0`, смогут изменять файлы с меткой безопасности `1:0:0:0`, читать содержимое файлов с меткой безопасности `1:0:0:0` или `0:0:0:0`).

Процессы (субъекты), работающие внутри контейнера в непривилегированном режиме, не могут изменять метки безопасности файловых объектов в файловой системе своего контейнера, т.к. для контейнеров в непривилегированном режиме нет возможности запуска с привилегиями (`docker run --privileged`) и, соответственно, нет доступа к программному интерфейсу Parsec (Parsec API) через ParsecFS.

Описание работы с контейнерами Docker в непривилегированном режиме с ненулевыми метками безопасности также приведено в `man rootless-helper-astra` `man rootlessenv`.

7.3.2. Управление запуском контейнера с ненулевой меткой безопасности

Для предоставления возможности работать с контейнерами в непривилегированном режиме от имени пользователя с ненулевой меткой безопасности следует запустить для этого пользователя непривилегированную службу Docker с указанием соответствующей метки безопасности.

Запуск службы для работы с ненулевой классификационной меткой выполняется командой:

```
sudo systemctl start rootless-docker@$(systemd-escape <имя_пользователя>@
    <метка_безопасности>)
```

Запуск службы для работы с различными классификационными метками выполняется командой:

```
sudo systemctl start rootless-docker@$(systemd-escape <имя_пользователя>@
    0:0:0:0@privsock)
```

при этом использование флага `privsock` запускает непривилегированную службу Docker с привилегией `PARSEC_CAP_PRIV_SOCK`, позволяющей выполнять команды для сетевых подключений (например, `docker pull`) игнорируя мандатные ограничения (см. 4.7).

Для настройки автоматического запуска службы после перезагрузки можно выполнить команду:

```
sudo systemctl enable rootless-docker@$(systemd-escape <имя_пользователя>@
    <метка_безопасности>@privsock)
```

После запуска для пользователя непривилегированной службы Docker возможно выполнение команд Docker от имени данного пользователя (например, копирование образов в пользовательские репозитории образов или запуск контейнеров).

7.3.3. Копирование образа в репозиторий пользователя

Для запуска контейнера пользователем в сессии с ненулевой классификационной меткой требуется создать пользовательский репозиторий с соответствующей меткой, содержащий образы.

Для создания пользовательских копий образов Docker необходимо:

1) экспортировать образ:

```
rootlessenv docker save -o /tmp/<имя_архива>.tar <имя_образа>
```

Экспорт рекомендуется выполнять в каталог `/tmp/`, доступный для чтения всем пользователям;

2) разрешить чтение экспортированного образа:

а) всем пользователям:

```
chmod o+r /tmp/<имя_архива>.tar
```

б) только указанным пользователям:

```
setfacl -m u:<имя_пользователя>:r /tmp/<имя_архива>.tar
```

3) импортировать образ с помощью инструмента `pdp-exec`, указав в команде имя пользователя с нужными метками безопасности (при этом должны быть запущены непривилегированные службы Docker для указанного пользователя с соответствующими метками безопасности):

```
sudo pdp-exec -u <имя_пользователя> -l <метка_безопасности>
     -- rootlessenv docker load -i /tmp/<имя_архива>.tar
```

7.3.4. Выполнение команд и запуск контейнеров в непривилегированном режиме от имени пользователя

Команды Docker, выполняемые пользователем (или администратором от имени пользователя) выполняются с меткой безопасности сессии пользователя (или указанной

администратором):

1) для запуска контейнера пользователем из своей сессии (контейнер будет запущен из репозитория образов пользователя, имеющего метку безопасности, равную метке безопасности сессии пользователя, и унаследует данную метку безопасности):

```
rootlessenv docker run <имя_образа>
```

2) для запуск администратором оболочки командой строки контейнера, из которой можно выполнять команды Docker от имени указанного пользователя, выполнить команду:

```
sudo pdp-exec -u <имя_пользователя> -l <метка_безопасности>
     -- rootlessenv
```

3) для запуска контейнера администратором от имени указанного пользователя выполнить команду:

```
sudo pdp-exec -u <имя_пользователя> -l <метка_безопасности>
     -- rootlessenv docker run --rm -ti <имя_образа>
```

Пользователь (администратор) может со стороны хостовой ОС присваивать метки безопасности файловым объектам в файловой системе контейнера (например, `1:0:0:0` для конфиденциальных пользовательских файлов и `0:2:0:0` для системных файлов контейнера с высокой целостностью).

Местонахождение файловой системы контейнера на хостовой ОС можно узнать командой:

```
docker inspect <имя_контейнера> | egrep "(Lower|Upper)Dir"
```

При работе администратора с контейнером пользователя данная команда должна выполняться от имени пользователя с указанием нужной метки безопасности:

```
sudo pdp-exec -u <имя_пользователя> -l <метка_безопасности>
     -- rootlessenv docker inspect <имя_контейнера> | egrep "(Lower|Upper)Dir"
```

2.20. Подраздел «9.5. Сервис электронной подписи»

В подразделе 9.5 исключить пункты «9.5.4. Использование на ненулевом уровне конфиденциальности» и «9.5.7. Условия применения СКЗИ».

В подразделе 9.5 изменить нумерацию пунктов в связи с исключением пункта.

Подраздел 9.5 и его пункты изложить в редакции:

9.5. Сервис электронной подписи

Комплекс программ из состава ОС предоставляет сервис электронной подписи (СЭП), обеспечивающий интеграцию с дополнительно устанавливаемыми сертифицированными ФСБ России средствами криптографической защиты информации[1)] (СКЗИ) в целях

[1)] Не допускается применение для защиты информации, содержащей сведения, составляющие государственную тайну.

создания и проверки усиленной квалифицированной электронной подписи.

ВНИМАНИЕ! СЭП предоставляется программами, функционирующими в условиях политики разграничения доступа, не допускающей их применение совместно с СКЗИ в режиме обработки сведений, составляющих государственную тайну.

ВНИМАНИЕ! Эксплуатация СКЗИ в составе информационных систем должна осуществляться в соответствии с правилами пользования СКЗИ и указаниями, определенными в формуляре (или иных эксплуатационных документах) на СКЗИ.

СЭП обеспечивает создание и проверку ЭП[1)] электронных документов в соответствии с ГОСТ Р 34.10-2012 средствами сертифицированных СКЗИ.

Дополнительная информация о настройке ОС и СКЗИ для совместного использования, а также варианты реализации отдельных решений приведены на официальном информационном ресурсе разработчика ОС `https://wiki.astralinux.ru`.

9.5.1. Принцип функционирования

СЭП обеспечивается совокупностью совместно функционирующих программных компонентов ОС:

- `astra-csp-cryptopro` — программный модуль, обеспечивающий предоставление СЭП из командной строки и программного интерфейса. Реализует функции создания и проверки ЭП без взаимодействия в графическом интерфейсе[2)] ;
- `fly-csp-cryptopro` — программный модуль, обеспечивающий предоставление СЭП из основного меню, графического интерфейса менеджера файлов `fly-fm` и командной строки. Предоставляет функции создания и проверки ЭП в графическом интерфейсе пользователя;
- `libreoffice-astracsp-plugin` — расширение, обеспечивающее предоставление СЭП из графического интерфейса пакета офисных программ LibreOffice в процессе работы с электронными документами;
- `fly-csp-preview` — расширение, обеспечивающее графическое представление на электронном документе отметки об ЭП;
- менеджер файлов `fly-fm`;
- пакет офисных программ LibreOffice.

Непосредственное взаимодействие с СКЗИ (формирование управляющих команд, получение и отображение результата работы СКЗИ с использованием программы `cryptcp`) осуществляет программный модуль `astra-csp-cryptopro`.

[1)] Электронная подпись (в соответствии с № 63-ФЗ от 06.04.2011) — информация в электронной форме, которая присоединена к другой информации в электронной форме (подписываемой информации) или иным образом связана с такой информацией и которая используется для определения лица, подписывающего информацию (является юридически значимой).

[2)] Здесь и далее в качестве примера приведено описание взаимодействия с СКЗИ «КриптоПро CSP».

Программный модуль fly-csp-cryptopro в целях решения функциональных задач СЭП взаимодействует с программным модулем astra-csp-cryptopro и не взаимодействует непосредственно с СКЗИ.

Менеджер файлов fly-fm, расширение пакета офисных программ LibreOffice libreoffice-astracsp-plugin и расширение для добавления отметки об ЭП fly-csp-preview в целях решения функциональных задач СЭП взаимодействуют с программным модулем fly-csp-cryptopro и не взаимодействуют непосредственно с СКЗИ.

Пакет офисных программ LibreOffice взаимодействует с расширением пакета офисных программ LibreOffice libreoffice-astracsp-plugin и не взаимодействует непосредственно с СКЗИ.

В процессе функционирования СЭП осуществляется вызов перечисленных в **??** программ из состава ОС для предварительного просмотра электронных документов, формируются управляющие команды для СКЗИ с целью вызова функций проверки или создания ЭП, осуществляется отображение получаемого в после выполнения вызываемых функций результата.

Обращение к программному модулю fly-csp-cryptopro осуществляется:

- менеджером файлов fly-fm (с использованием соответствующего пункта контекстного меню);
- пользователем путем интерактивного запуска приложения с использованием основного меню ОС («Утилиты — Электронная подпись КриптоПро»);
- расширением пакета офисных программ LibreOffice, запуск которого инициируется путем вызова пользователем соответствующего пункта меню графического интерфейса пакета офисных программ LibreOffice.

Обращение к программному модулю astra-csp-cryptopro осуществляется пользователем посредством интерфейса командной строки и прикладным программным обеспечением путем запуска на выполнение программного модуля с соответствующими параметрами.

9.5.2. Условия применения

В качестве СКЗИ разрешается использовать только КриптоПро CSP версии 5.0 R2 (исполнения 1-Base и 2-Base).

СКЗИ КриптоПро CSP в исполнении 2-Base должно использоваться с аппаратно-программным модулем доверенной загрузки (АПМДЗ).

При эксплуатации СКЗИ необходимо соблюдать требования и рекомендации эксплуатационной документации на СКЗИ, в частности требования по защите от несанкционированного доступа и по криптографической защите, а также требования по поддерживаемым СКЗИ аппаратно-программным платформам. В частности, при использовании СЭП со встро-

енным СКЗИ необходимо проведение проверки программного обеспечения загрузчика ЭВМ, на которых предполагается функционирование СКЗИ и СЭП, на соответствие методическим документам ФСБ России в области исследований программного обеспечения загрузчика.

Контроль целостности СКЗИ и СЭП должен выполняться с использованием механизма замкнутой программной среды ОС или с использованием стандартных средств контроля целостности КриптоПро CSP.

Из числа компонентов СЭП контролю целостности подлежит программный модуль `astra-csp-cryptopro`. Контрольная сумма программного модуля `astra-csp-cryptopro`, рассчитанная по ГОСТ Р 34.11-2012 с использованием программы подсчета контрольных сумм `gostsum` из состава изделия:

`c38ce123cee3aeaf38793b907e763573`

`55b78e7407766da35892fd224c555f19`

Для подсчета контрольной суммы с использованием `gostsum` необходимо на рабочей станции под управлением изделия выполнить в командной строке:

`gostsum <путь_к_файлу>`

Перечень компонентов СКЗИ, подлежащих контролю целостности, приведен в эксплуатационной документации на СКЗИ.

Подробный порядок настройки механизма контроля целостности описан в эксплуатационной документации на ОС и используемое СКЗИ.

9.5.3. Установка

Менеджер файлов `fly-fm` и пакет офисных программ LibreOffice из состава ОС устанавливаются в соответствии с эксплуатационной документацией ОС.

Для использования СЭП дополнительно в системе должны быть установлены следующие программные компоненты (описание компонентов приведено в 9.5.1) из состава ОС:

- программный модуль `astra-csp-cryptopro`;
- программный модуль `fly-csp-cryptopro`;
- расширение `libreoffice-astracsp-plugin`.

Для установки программного модуля и расширения необходимо в терминале последовательно выполнить следующие команды от имени администратора:

```
apt update
apt install astra-csp-cryptopro
apt install fly-csp-cryptopro
apt install libreoffice-astracsp-plugin
```

При установке расширения libreoffice-astracsp-plugin проводится проверка наличия в информационной системе установленного программного модуля fly-csp-cryptopro и пакета офисных программ LibreOffice.

При запуске программных модулей fly-csp-cryptopro и astra-csp-cryptopro проводится проверка наличия в информационной системе установленного программного обеспечения СКЗИ и пакета офисных программ LibreOffice.

Дополнительно возможно установить расширение fly-csp-preview (описание расширения приведено в 9.5.1), выполнив в терминале команду от имени администратора: apt install fly-csp-preview

Программные модули astra-csp-cryptopro и fly-csp-cryptopro разработаны на языке программирования Python версии 3 и включают в свой состав:

1) основной файл программы /usr/bin/astra-csp-cryptopro;
2) файл графической программы /usr/bin/fly-csp-cryptopro;
3) дополнительные файлы и библиотеки в каталогах /usr/lib/python3/dist-packages/astra_csp_cryptopro и /usr/lib/python3/dist-packages/astra_csp-0.1.0.egg-info;
4) файлы с указанием значений параметров СЭП, задаваемых по умолчанию:
   а) /usr/share/astra-csp/astra-csp.yaml — общие параметры программных модулей;
   б) /usr/share/astra-csp/fly-csp.yaml — параметры программного модуля fly-csp-cryptopro;
   в) /usr/share/astra-csp/cryptopro/astra-csp-cryptopro.yaml — порядок взаимодействия программного модуля astra-csp-cryptopro с СКЗИ;
   г) /usr/share/astra-csp/cryptopro/fly-csp-cryptopro.yaml — порядок взаимодействия программного модуля fly-csp-cryptopro с СКЗИ;
5) ярлык для запуска /usr/share/flyfm/actions/fly-csp-cryptopro.desktop.

Файл /usr/share/astra-csp/astra-csp.yaml содержит следующие параметры и их значения по умолчанию:

1) максимально допустимый для предоставления СЭП уровень конфиденциальности:

```
max_mac_level: 1
```

2) максимальное время ожидания выполнения запросов к СКЗИ:

```
max_csp_time_waiting: 30000
```

3) перечень расширений файлов электронных документов, для которых разрешено предоставление СЭП:

```
file_extensions_filter:
...
```

```
extensions: ['.pdf', '.png', '.jpg', '.html', '.xml', '.txt', '.odt',
'.ods', '.odp', '.odg', '.doc', '.xls', '.ppt', '.docx', '.xlsx', '.pptx']
```

4) отключение (`false`) или включение (`true`) действия перечня, заданного в пункте 3) перечисления (по умолчанию значение `false`):

```
file_extensions_filter:
enable: false
```

Установка значения `false` допускается только для целей тестирования и апробации технических решений с применением СЭП в информационной системе.

В файле `/usr/share/astra-csp/fly-csp.yaml` приведен перечень mime-типов файлов электронных документов, для просмотра которых будет проведена предварительная конвертация во временный файл формата PDF:

```
pdf_conversion_mime_types:
- application/vnd.oasis.opendocument.text
- application/vnd.oasis.opendocument.spreadsheet
- application/vnd.oasis.opendocument.presentation
- application/vnd.oasis.opendocument.graphics
- application/msword
- application/vnd.ms-word
- application/vnd.ms-excel
- application/vnd.ms-powerpoint
- application/vnd.openxmlformats-officedocument.wordprocessingml./
  document
- application/vnd.openxmlformats-officedocument.spreadsheetml.sheet
- application/vnd.openxmlformats-officedocument.presentationml./
  presentation
- text/plain
```

В случае, если mime-тип выбранного файла электронного документа включен в список, определенный в файле `/usr/share/astra-csp/fly-csp.yaml`, будет проведена предварительная конвертация файла электронного документа в файл формата PDF.

Файл `/usr/share/astra-csp/cryptopro/astra-csp-cryptopro.yaml` содержит параметр, задающий адрес источника доверенного времени:

```
time_address: 'http://testca2012.cryptopro.ru/tsp/tsp.srf'
```

Данный источник допустимо использовать только для выполнения тестирования. При эксплуатации СЭП источник доверенного времени должен быть задан в файле `/etc/astra-csp/astra-csp-cryptopro.yaml` (если файл отсутствует, то необходимо создать его) в качестве значения параметра `time_address`:

```
time_address: <адрес_источника_доверенного_времени>
```

ВНИМАНИЕ! Внесение изменений в любые файлы из состава СЭП, расположенные в каталоге `/usr/`, в т.ч. в `/usr/share/astra-csp/`, запрещено. При необходимо-

сти переопределить значения параметров, заданных по умолчанию в файлах в каталоге /usr/share/astra-csp/, требуется изменить значения параметров в файлах с аналогичными именами, расположенных в каталоге /etc/astra-csp/ (если файл с таким же именем отсутствует, то его следует создать). Переопределять значения параметров разрешено только администратору.

ВНИМАНИЕ! Добавление в конфигурационный файл mime-типов, отличных от перечисленных, без проведения дополнительных исследований запрещено.

Для изменения параметров, указанных в файлах /usr/share/astra-csp/astra-csp.yaml и /usr/share/astra-csp/fly-csp.yaml, необходимо создать файлы /etc/astra-csp/astra-csp.yaml и /etc/astra-csp/fly-csp.yaml, в которых требуется указать только изменяемые параметры с новыми значениями. Для остальных параметров будут использоваться значения, заданные по умолчанию в файлах /usr/share/astra-csp/astra-csp.yaml и /usr/share/astra-csp/fly-csp.yaml.

Также возможно изменить значения некоторых параметров, указанных в файлах /usr/share/astra-csp/astra-csp.yaml и /usr/share/astra-csp/fly-csp.yaml, с использованием графической утилиты fly-csp-cryptopro (см. электронную справку). При этом файлы /etc/astra-csp/astra-csp.yaml и /etc/astra-csp/fly-csp.yaml с измененными значениями параметров будут созданы автоматически.

9.5.4. Просмотр электронного документа

Просмотр документа при создании и проверке электронной подписи при работе СЭП осуществляется с использованием программных средств, включенных в состав ОС. Доступен предварительный просмотр электронных документов, хранящихся в файлах следующих форматов:

1) PDF — для просмотра используется программа Okular;
2) PNG, JPG — для просмотра используется программа gwenview;
3) HTML, XML — для просмотра используется браузер Firefox.

Для предварительного просмотра электронных документов, хранящихся в файлах форматов TXT, ODT, ODS, ODP, ODG, DOC, XLS, PPT, DOCX, XLSX и PPTX, осуществляется их автоматическая предварительная конвертация в формат PDF и отображение пользователю.

Перечень mime-типов файлов, предполагающих для просмотра предварительную конвертацию в формат PDF, задается в конфигурационном файле в соответствии с описание в 9.5.3 и приведен в таблице 41.

Таблица 41

Mime-тип файла	Описание
text/plain	Текстовый документ TXT
application/vnd.oasis.opendocument.text	Текстовый документ ODT
application/vnd.oasis.opendocument.spreadsheet	Электронная таблица ODS
application/vnd.oasis.opendocument.presentation	Презентация ODP
application/vnd.oasis.opendocument.graphics	Графический документ ODG
application/msword	Текстовый документ DOC
application/vnd.ms-word	Текстовый документ DOC
application/vnd.ms-excel	Электронная таблица XLS
application/vnd.ms-powerpoint	Презентация PPT
application/vnd.openxmlformats-officedocument.wordprocessingml.document	Текстовый документ DOCX
application/vnd.openxmlformats-officedocument.spreadsheetml.sheet	Электронная таблица XLSX
application/vnd.openxmlformats-officedocument.presentationml.presentation	Презентация PPTX

9.5.5. Проверка уровня конфиденциальности сеанса

В целях исключения использования СКЗИ при обработке информации, содержащей сведения, составляющие государственную тайну, перед началом работы выполняется проверка допустимого для предоставления СЭП уровня конфиденциальности путем сравнения уровня конфиденциальности сеанса текущего пользователя с максимально допустимым значением, заданным в конфигурационном файле.

Максимально допустимый для предоставления СЭП уровень конфиденциальности задается в конфигурационном файле `/usr/share/astra-csp/astra-csp.yaml` параметром `max_mac_level`, а также в графической утилите `fly-csp-cryptopro` (см. электронную справку). По умолчанию установлено значение `max_mac_level: 1` и может быть изменено администратором безопасности. Политикой управления доступом и организационно-распорядительными документами по защите информации объекта информатизации (объекта вычислительной техники) должен быть предусмотрен явный запрет внесения изменений в конфигурационный файл любыми пользователями, за исключением ответственного за защиту информации администратора безопасности, зарегистрированного в ОС как привилегированный пользователь.

В случае, если уровень конфиденциальности текущего сеанса пользователя превышает максимально допустимый, программы `fly-csp-cryptopro` и `astra-csp-cryptopro` выдают сообщение об ошибке и завершают свою работу.

2.21. Подраздел «10.1. Восстановление ОС после сбоев и отказов»

Подраздел 10.1 начиная с абзаца «После серьезного повреждения ФС, когда компьютер невозможно перезагрузить...» и до конца изложить в редакции:

10.1. Восстановление ОС после сбоев и отказов

По тексту...

После серьезного повреждения ФС, когда компьютер невозможно перезагрузить, существует возможность восстановления без переустановки ОС. Для этого необходимо:

1) установить DVD-диск с дистрибутивом ОС в устройство чтения DVD-дисков;

2) загрузить программу установки ОС с DVD-диска;

3) в окне приветствия программы установки выбрать язык установки (русский или английский);

4) в окне приветствия программы установки выбрать «Режим восстановления»;

5) в окне «[!] Лицензия» подтвердить согласие с лицензионным соглашением;

6) в окне «[!] Настройка клавиатуры» выбрать настройки переключения раскладки клавиатуры, после чего программой установки будет выполнена проверка оборудования и первичная загрузка программ;

7) в окне «[!] Настройка сети» задать имя компьютера (можно указать произвольное имя компьютера, настройки восстанавливаемой ОС не изменятся);

8) в окне «[!] Настройка времени» выбрать часовой пояс;

9) в окне «[!] Войти в режим восстановления» последовательно выполнить следующие шаги:

   а) выбрать пункт «Не использовать корневую файловую систему»;

   б) выбрать следующую операцию режима восстановления: «Запуск оболочки в рабочей среде программы установки»;

   в) нажать на кнопку **[<Продолжить>]**.

Будет выполнен переход в режим командной строки под управлением ядра, загруженного с DVD-диска;

10) определить имя раздела, в который была установлена ОС, для этого выполнить команду:

```
blkid
```

На экране монитора должна появиться информация о разделах жесткого диска (если в результате ввода команды на экране монитора нет информации о разделах диска, то повреждения слишком серьезны и необходима полная переустановка системы).

П р и м е р

Вывод выполнения команды `blkid`

```
/dev/sda1: UUID="bc485787-ef37-431c-8c8b-401055066c99" TYPE="ext4"
     PARTUUID="9492e90e-01"
/dev/sda5: UUID="e8987cad-ee16-427a-a768-a9aa896b048c" TYPE="swap"
     PARTUUID="9492e90e-05"
/dev/sr0: UUID="2021-06-11-12-41-04-00" LABEL="Astra 4.7_arm"
     TYPE="iso9660" PTUUID="66c613b0" PTTYPE="dos"
```

В приведенном примере ОС была установлена в раздел `/dev/sda1`;

11) запустить автоматическую проверку и восстановление ФС, выполнив команду:

```
fsck.ext4 -p -f -c /dev/<имя раздела>
```

Пример

Вывод выполнения команды `fsck`

```
/dev/sda1:Updating bad block inode.
/dev/sda1:318177/2297456 files (0.2% non-contiguous), 4157309/9186816
     blocks
```

12) после проверки нажать комбинацию клавиш **<Ctrl+D>** и извлечь DVD-диск с дистрибутивом ОС из устройства чтения DVD-дисков;

13) в окне «[! !] Войти в режим восстановления» выбрать пункт «Перезагрузка системы».

2.22. Пункт «10.2.2. pg_dump»

В таблице 42 добавить описание параметра `--disable-macs`.

Таблица 42

Опция	Описание
`--disable-macs`	Исключить из выгрузки вывод команд `MAC LABEL` и `MAC CCR`

2.23. Пункт «16.1.1. Режимы функционирования»

Первый абзац пункта 16.1.1 изложить в редакции:

Инструменты замкнутой программной среды (ЗПС) предоставляют возможность внедрения ЭЦП[1)] в исполняемые файлы формата ELF, входящие в состав устанавливаемого СПО, и в расширенные атрибуты файловой системы, обеспечивая таким образом динамический контроль целостности.

[1)] Электронная цифровая подпись — строка бит, полученная в результате процесса формирования подписи (применяется для подписи средствами ОС исполняемых файлов с использованием функции хэширования на базе асимметричного криптографического алгоритма (в соответствии с ГОСТ Р 34.11-2012).

2.24. Пункт «16.4. Функции безопасности системы»

Пункт 16.4 изложить в редакции:

16.4. Функции безопасности системы

Пакет `astra-safepolicy` содержит инструменты управления функциями безопасности системы. Описание инструментов приведено в 16.4.3-16.4.32.

Все инструменты из состава пакета `astra-safepolicy` поддерживают стандартный набор параметров вызова, приведенный в 16.4.1. Некоторые инструменты дополнительно к стандартному набору параметров вызова поддерживают дополнительные параметры. Описание дополнительных параметров приведено в описании соответствующего инструмента.

Инструменты пакета `astra-safepolicy` выступают в роли команд-переключателей, осуществляющих включение и выключение соответствующих функций безопасности. Для просмотра состояния некоторых команд-переключателей можно использовать инструмент `astra-security-monitor` из состава пакета `astra-safepolicy`, описание инструмента приведено в 16.4.2.

2.25. Пункт «16.4.2. Монитор безопасности»

Пункт 16.4.2 изложить в редакции:

16.4.2. Монитор безопасности

Утилита `astra-security-monitor` отображает информацию о состоянии некоторых функций безопасности, а также выводит информацию о состоянии функции безопасности по ее идентификатору.

Примечание. Отображение состояния функции безопасности также зависит от наличия установленных в системе соответствующих пакетов или служб.

Для функций безопасности, информацию о которых выводит утилита, возможны следующие состояния:

- `ВКЛЮЧЕНО` — функция безопасности активна;
- `ВЫКЛЮЧЕНО` — функция безопасности неактивна;
- `ВКЛЮЧАЕТСЯ` — функция безопасности находится в процессе включения или будет включена после перезагрузки, но в настоящий момент неактивнаа;
- `ВЫКЛЮЧАЕТСЯ` — функция безопасности находится в процессе выключения или будет выключена после перезагрузки, но в настоящий момент активна;
- `ЧАСТИЧНО` — функция безопасности включена, но не все параметры, контролируемые этой функцией, соответствуют заданным по умолчанию.

При отображении состояний отдельных функций безопасности учитывается следующее:

1) при выводе информации о состоянии МКЦ на файловой системе проверяется соответствие уровней целостности объектов ФС уровням целостности, указанным в конфигурационном файле `/etc/parsec/fs-ilev.conf`. При несоответствии уровней целостности выводится сообщение о количестве файловых объектов, уровень целостности которых не соответствует заданному в конфигурационном файле:

а) «ниже» — уровень целостности файлового объекта ниже заданного;

б) «выше» — уровень целостности файлового объекта выше заданного;

в) «норма» — уровень целостности файлового объекта соответствует заданному.

Список файловых объектов, уровень целостности которых не соответствует заданному в файле `/etc/parsec/fs-ilev.conf`, можно вывести командой:

```
sudo set-fs-ilev status -v
```

2) для функции проверки подписи в расширенных атрибутах `xattr` — включена ли проверка подписи не только в исполняемых файлах, но и в других файлах, которые подписаны в `xattr` соответствующей утилитой.

Дополнительно монитор безопасности выводит следующую информацию:

1) запрет входа `root` по ssh (если установлены средства удаленного подключения ssh) — запрет удаленного входа в систему пользователю `root`. По умолчанию `root` не может войти через ssh, функция запрета имеет состояние `ВКЛЮЧЕНО`;

2) безопасный вход в домен (если компьютер введен в домен) — значение `ВКЛЮЧЕНО`, если в файле `/etc/parsec/parsec.conf` параметр `login_local` имеет значение `admin` (вход для локального пользователя разрешен, если пользователь входит в группу `astra-admin`) или значение `no` (вход для локальных пользователей запрещен);

3) системный киоск — значение `ВКЛЮЧЕНО`, если системный киоск включен и настроен хотя бы для одного пользователя;

4) графический киоск — значение `ВКЛЮЧЕНО`, если графический киоск настроен хотя бы для одного пользователя.

Подробное описание инструмента приведено в `man astra-security-monitor`.

При работе с `astra-security-monitor` также может использоваться графическая утилита `fly-admin-smc` («Монитор безопасности»).

2.26. Пункт «16.4.6. Блокировка интерпретаторов»

В пункте 16.4.6 перечень блокируемых интерпретаторов дополнить следующими наименованиями:

- nodejs;
- php.

2.27. Пункт «16.4.9. Блокировка клавиши <SysRq>

В пункте 16.4.9 добавить предупреждение:

ВНИМАНИЕ! Если в командной строке ядра указан параметр `sysrq_always_enabled`, то при включении функции блокировки клавиши **<SysRq>** вызов `status` будет выводить `НЕАКТИВНО`, а вызов `is-enabled` будет выводить `ВКЛЮЧЕНО`.

2.28. Пункт «16.4.15. Управление загрузкой модуля ядра lkrg»

Пункт 16.4.15 дополнить примечанием в редакции:

Примечание. Модуль ядра lkrg работает с использованием механизма kprobes, предназначенного для отладки и трассирования ядра ОС. Этот механизм отключен в ядре hardened, в связи с этим функционирование модуля lkrg не поддерживается в ядре hardened.

2.29. Пункт «16.4.22. Блокировка выключения компьютера пользователями»

Пункт 16.4.22 изложить в редакции:

16.4.22. Блокировка выключения компьютера пользователями

Инструмент `astra-shutdown-lock` блокирует выключение компьютера пользователями. Параметры вызова, используемые данным инструментом, приведены в таблице 63.

При включении данной функции добавляется политика `policykit`, которая запрещает выключение компьютера без ввода пароля администратора. Также при включении инструмента блокируется возможность перезагрузки компьютера путем нажатия комбинации клавиш **<Ctrl+Alt+Delete>**.

Изменение режима блокировки вступает в действие немедленно.

Описание инструмента приведено в `man astra-shutdown-lock`.

Управление блокировкой выключения компьютера пользователями также может осуществляться с помощью графической утилиты `fly-admin-smc`.

2.30. Пункт «16.4.27. Управление режимом мандатного контроля целостности»

Изменить заголовок пункта 16.4.27, первые два абзаца изложить в редакции:

16.4.27. Управление мандатным контролем целостности

Инструмент `astra-mic-control` включает и выключает МКЦ, описанный в 4.8, а также изменяет значение максимального уровня целостности системы. Управление МКЦ осуществляется путем изменения значения параметра ядра `parsec.max_ilev`.

Параметры вызова, используемые данным инструментом, приведены в таблице 63. При использовании вызова `enable` возможно указать необязательный параметр `-i <уровень>`, задающий значение максимального уровня целостности (после установки ОС максимальный уровень целостности равен 63).

2.31. Пункт «16.4.32. Управление AstraMode и MacEnable»

Ввести новый пункт 16.4.32:

16.4.32. Управление AstraMode и MacEnable

Инструмент `astra-mode-apps` включает и выключает режим AstraMode сервера Apache2, а также управляет состоянием параметра `MacEnable` сервера печати CUPS.

Параметры вызова, используемые инструментом `astra-mode-apps`, приведены в таблице 63.

При использовании команды-переключателя вносятся изменения в конфигурационные файлы `/etc/apache2/apache2.conf` и `/etc/cups/cupsd.conf`, изменяя значения параметров `AstraMode` и `MacEnable` соответственно.

Для применения изменений требуется перезапуск служб.

Описание инструмента приведено в `man astra-mode-apps`.

Включение и выключение режима AstraMode и управление параметром `MacEnable` также можно осуществлять с помощью графической утилиты `fly-admin-smc` (см. электронную справку).

2.32. Подраздел «16.6. Модуль безопасности lockdown»

В подразделе 16.6 исключить последний абзац и добавить следующий текст:

Состояние модуля lockdown задается в файле `/sys/kernel/security/lockdown`, который по умолчанию имеет следующий вид:

```
[none] integrity confidentiality
```

В данном файле приведены возможные состояния модуля lockdown, а его текущее состояние отображается в квадратных скобках (`[none]` — модуль отключен).

После загрузки ОС модуль lockdown по умолчанию отключен.

Активировать модуль lockdown в нужном режиме работы можно в процессе функционирования ОС путем записи соответствующего режима в файл `/sys/kernel/security/lockdown` командой:

```
sudo echo <режим_модуля> > /sys/kernel/security/lockdown
```

где `<режим_модуля>` — `integrity` (для активации режима `integrity`) или `confidentiality` (для активации режима `confidentiality`), при этом:

- если модуль был отключен, то возможно активировать любой режим;
- если модуль работал в режиме `integrity`, то возможно только активировать режим `confidentiality`;
- если модуль работал в режиме `confidentiality`, то изменить режим или отключить модуль невозможно.

Изменения в режиме работы модуля lockdown сохраняются до перезагрузки ОС.

2.33. Подраздел «17.2. Указания по эксплуатации ОС»

В подразделе 17.2 в пункте 17.2.1 перечисление 3) изложить в новой редакции и добавить перечисление 7). В пункте 17.2.4 добавить перечисление 8):

17.2.1. Перед началом эксплуатации ОС администратор безопасности должен обеспечить следующие условия:

3) используя графическую утилиту `fly-admin-smc` («Управление политикой безопасности», см. электронную справку), запущенную от имени администратора через механизм `sudo`, в категории «Политики учетной записи» задать значения для настройки блокировки при неуспешных входах пользователя в систему. Данные параметры также могут быть установлены путем корректировки файла `/etc/pam.d/common-auth` (параметры `deny`, `lock_time` и `unlock_time`);

7) задать значение времени неактивности для блокировки экрана, отредактировав (или создав, если отсутствует) файл `usr/share/fly-wm/theme.master/themerc`, указав в нем строки:

```
[Variables]
ScreenSaverDelay=<время_неактивности_в_секундах>
LockerOnSleep=true
LockerOnDPMS=true
LockerOnLid=true
LockerOnSwitch=true
```

17.2.4. При использовании мандатного управления доступом должны дополнительно быть выполнены следующие условия:

8) не должен использоваться программный коммутатор Open vSwitch.

2.34. Подраздел «17.3. Условия применения ПО»

В пункте 17.3.1 перечисление 4) изложить в редакции:

4) изменять параметры аутентификации в конфигурационных файлах PAM-сценариев, находящихся в каталоге `/etc/pam.d`, результатом чего может являться снижение установленного уровня доверия к результатам идентификации и аутентификации (по ГОСТ Р 58833-2020);