ОПЕРАЦИОННАЯ СИСТЕМА СПЕЦИАЛЬНОГО НАЗНАЧЕНИЯ «ASTRA LINUX SPECIAL EDITION» РУСБ.10015-01

Руководство по КСЗ. Часть 1

Оперативное обновление 1.8.3

Бюллетень № 2025-0811SE18

Листов 15

2025

СОДЕРЖАНИЕ

1. ОБЩИЕ СВЕДЕНИЯ

В настоящем документе приведены кумулятивные изменения в документ РУСБ.10015-01 97 01-1 «Операционная система специального назначения «Astra Linux Special Edition». Руководство по КСЗ. Часть 1» из комплектности изделия РУСБ.10015-01 «Операционная система специального назначения «Astra Linux Special Edition» (далее по тексту — ОС).

При администрировании комплекса средств защиты ОС с установленным оперативным обновлением согласно бюллетеню № 2025-0811SE18 рекомендуется руководствоваться документом РУСБ.10015-01 97 01-1 совместно с настоящим документом.

Документ предназначен для администраторов безопасности.

2. ПЕРЕЧЕНЬ ИЗМЕНЕНИЙ

2.1. Подраздел «4.18. Настройка параметров ядра в загрузчике GRUB 2»

В подразделе 4.18 в таблице 31 изложить описание параметра `parsec.rename_mask` в следующей редакции:

Таблица 31

Параметр	Описание
parsec.rename_mask	Позволяет управлять механизмом сложения меток безопасности, правил PARSEC-аудита и дополнительного атрибута ssi при замене и перезаписи файлов. Данный механизм, в частности, предотвращает сброс меток безопасности, правил PARSEC-аудита и атрибута ssi у файлов при следующих операциях: 1) редактирование или переименование файлов текстовыми редакторами, использующими временные файлы (изменения записываются во временный файл, исходный файл заменяется временным при сохранении); 2) переименование файла командой: mv <старое_имя> <новое_имя> Если уже существует файл с указанным новым именем, то он будет перезаписан переименуемым файлом. Значение параметра представляется в виде битовой маски, в которой каждый бит отвечает за установку соответствующего флага. В системе битовая маска задается суммой соответствующих десятичных значений: 1) первый бит RENAME_MASK_LEV (двоичное значение 00001, десятичное значение 1) — выбор максимального из уровней конфиденциальности; 2) второй бит RENAME_MASK_CAT (двоичное значение 00010, десятичное значение 2) — объединение категорий конфиденциальности (побитовое ИЛИ); 3) третий бит RENAME_MASK_ILEV (двоичное значение 00100, десятичное значение 4) — объединение категорий целостности и выбор максимального линейного уровня целостности (побитовое ИЛИ); 4) четвертый бит RENAME_MASK_FLAGS (двоичное значение 01000, десятичное значение 8) — сохранение атрибута ssi при его наличии у исходного или перезаписываемого файлов; 5) пятый бит (двоичное значение 10000, десятичное значение 16) — сохранение правил PARSEC-аудита при их наличии у исходного или перезаписываемого файла. Если правила установлены для обоих файлов, то сохраняются правила исходного файла. Значение по умолчанию 31 (все биты включены, метки безопасности и дополнительные атрибуты сущностей объединяются)

2.2. Пункт «5.6.2. Применение механизма контроля целостности с использованием алгоритма работы с контрольными суммами («отпечаток конфигурации»)»

В пункте 5.6.2 изложить указанный абзац в следующей редакции:

Легитимное управление конфигурациями виртуальной инфраструктуры (в том числе создание новых ВМ) при включенном механизме «отпечаток конфигурации» возможно только в контексте доверенных процессов средства виртуализации `libvirt` и выполняется с помощью инструментов, представленных в таблице 32.

Таблица 32

Параметр	Описание
`virt-manager`	Управление средой виртуализации и виртуальными машинами в графическом интерфейсе. Подробная информация об использовании инструмента приведена в электронной справке «Менеджер виртуальных машин»
`virsh`	Управление средой виртуализации и виртуальными машинами в интерфейсе командной строки. Подробная информация об использовании инструмента приведена на справочной странице `man virsh`
`virt-clone`	Клонирование ВМ. Подробная информация об использовании инструмента приведена на справочной странице `man virt-clone`
`virt-export`	Экспорт ВМ. Подробная информация об использовании инструмента приведена на справочной странице `man virt-export`, а также в 5.9.7
`virt-import`	Импорт ВМ из файла архива, полученного при экспорте ВМ. Подробная информация об использовании инструмента приведена на справочной странице `man virt-import`, а также в 5.9.7
`virt-install`	Создание ВМ. Подробная информация об использовании инструмента приведена на справочной странице `man virt-install`
`virt-xml`	Изменение конфигурации ВМ. Подробная информация об использовании инструмента приведена на справочной странице `man virt-xml`
`virtnbdbackup`	Создание резервной копии ВМ. Подробная информация об использовании инструмента приведена на справочной странице `man virtnbdbackup`, а также в 5.9.5.1
`virtnbdrestore`	Восстановление ВМ из резервной копии. Подробная информация об использовании инструмента приведена на справочной странице `man virtnbdrestore`, а также в 5.9.5.2

В пункте 5.6.2 изложить предпоследний абзац в следующей редакции:

Создание и удаление объектов виртуальной инфраструктуры при использовании механизма

«отпечаток конфигурации» необходимо выполнять с помощью `virsh` или `virt-manager`, обеспечивающих легитимное создание и удаление эталонных значений хеша.

2.3. Пункт «5.6.3. Применение механизма контроля целостности файлов гостевой операционной системы»

2.3.1. В пункте 5.6.3 изложить указанный и следующий за ним абзацы в редакции:

Перед постановкой на контроль целостности файлов гостевой операционной системы ВМ необходимо для каждого файла определить:

1) полный путь к контролируемому файлу в гостевой операционной системе ВМ;

2) наименование или идентификатор (UUID) раздела диска, на котором хранится контролируемый файл гостевой операционной системы ВМ;

3) точку монтирования раздела диска ВМ.

Для определения наименования и идентификатора раздела диска, а также его точки монтирования в гостевой операционной системе ВМ возможно использовать команду, которая отобразит таблицу разделов:

```
lsblk -f
```

Пример

Выполнить команду:

```
 lsblk -f
```

Вывод команды:

```
NAME   FSTYPE FSVER LABEL  UUID                 FSAVAIL FSUSE% MOUNTPOINTS

vda
|-vda1
|-vda2 ext4   1.0   system 44c1be57-1593-4e6d-82a4 25,8G   5%     /
```

В выводе команды информация о том, что все файлы размещены на дисковом разделе `vda2` с идентификатором `44c1be57-1593-4e6d-82a4`, раздел `vda2` смонтирован в корневой каталог «/».

2.3.2. В пункте 5.6.3 изложить указанный абзац в следующей редакции:

Для постановки на контроль целостности файлов гостевой операционной системы ВМ необходимо в хостовой ОС для каждого файла выполнить команду:

```
virsh -c qemu:///system file-integrity <имя_ВМ> --path-add <путь_к_файлу> --mount-\
   point <точка_монтирования> --disk-name <имя_раздела>
```

или

```
virsh -c qemu:///system file-integrity <имя_ВМ> --path-add <путь_к_файлу> --mount-\
   point <точка_монтирования> --disk-uuid <UUID_раздела>
```

где <имя_ВМ> — наименование ВМ;
<путь_к_файлу> — полный путь к файлу, подлежащему контролю, в гостевой ОС;
<точка_монтирования> — точка монтирования раздела, на котором хранится файл в гостевой ОС;
<имя_раздела> — наименование раздела, на котором хранится файл в гостевой ОС;
<UUID_раздела> — идентификатор раздела, на котором хранится файл в гостевой ОС.

Примеры:

1. Постановить файл на контроль с указанием наименования раздела, на котором хранится файл:

```
virsh -c qemu:///system file-integrity alse --path-add /bin/afick --mount-\
   point / --disk-name vda2
```

2. Постановить файл на контроль с указанием идентификатора раздела, на котором хранится файл:

```
virsh -c qemu:///system file-integrity alse --path-add /bin/astra-version --\
   mount-point / --disk-uuid 44c1be57-1593-4e6d-82a4
```

2.4. Пункт «6.5.1. Регистрация событий и уведомления о событиях»

В пункте 6.5.1 изложить указанный абзац в следующей редакции:

В выводе списка событий значение рядом с событием:

- «1» — событие регистрируется;
- «0» — событие не регистрируется.

Синтаксис команды:

```
astra-admin-events [параметры]
```

Описание параметров инструмента приведено в таблице 41.

Таблица 41

Параметр	Описание
-E, --event-id	Вывести подробную информацию о событии (событиях), если не указаны другие параметры
-G, --group-id	Вывести статус событий из группы (групп), если не указаны другие параметры
-e, --enable	Включить события, которые указаны параметром -E или -G
-d, --disable	Отключить события, которые указаны параметром -E или -G
-p, --priority	Назначить приоритет событиям, которые указаны параметром -E или -G
-f, --facility	Назначить тип событиям, которые указаны параметром -E или -G
-P, --properties	Назначить свойства событиям, которые указаны параметром -E или -G
-i, --import <имя_файла>	Импортировать конфигурацию из указанного файла
-o, --export <имя_файла>	Экспортировать конфигурацию в указанный файл
-h, --help	Показать справочное сообщение и выйти

Порядок использования инструмента приведен в man astra-admin-events.

2.5. Пункт «12.8.4. Использование nftables для работы с классификационными метками»

Ввести новый пункт 12.8.4 в следующей редакции:

Фильтр сетевых пакетов nftables из состава ОС поддерживает классификационные метки (уровни и категории конфиденциальности) и может использоваться для фильтрации сетевого потока в условиях мандатного управления доступом.

Классификационные метки размещаются в дополнительном поле Опции заголовка IP-пакета. Для фильтрации IP-пакетов в условиях мандатного управления доступом в выражении для отбора пакетов ip option используется условие отбора astra.

Для управление конфигурацией фильтра сетевых пакетов, включая добавление и удаление таблиц, цепочек и правил, применяется инструмент командной строки nft.

Синтаксис команды для добавления правила фильтрации IP-пакетов, которые содержат классификационные метки:

```
sudo nft add rule <семейство_таблиц> <таблица> <цепочка> ip option \
    astra <параметры> <действие_или_переход>
```

Описание параметров условия отбора `astra` приведено в таблице 57.

Таблица 57

Параметр	Описание
`label <уровень_конфиденциальности> <категории_конфиденциальности>`	Задать значение классификационной метки. **ВНИМАНИЕ!** Не допускается использовать параметр `label`, чтобы задать нулевую классификационную метку. Вместо этого необходимо использовать параметр `missing`
`missing`	Задать нулевую классификационную метку
`exists`	Задать ненулевую классификационную метку

Порядок использования инструмента приведен на справочной странице `man nft`.

Примеры:

1. Добавить правило: не пропускать исходящие пакеты, которые имеют уровень конфиденциальности 2 и нулевой набор категорий конфиденциальности:

```
sudo nft add rule inet filter output ip option astra label 2 0 drop
```

2. Добавить правило: принимать входящие пакеты, которые имеют нулевой уровень конфиденциальности и набор категорий конфиденциальности `0x3` (установлены `Категория_1` и `Категория_2`). А также регистрировать факт поступления таких пакетов в журнале `/var/log/syslog`. При этом каждая запись будет начинаться с префикса `Набор_категорий_3`:

```
sudo nft add rule inet filter input ip option astra label 0 3 \
    log prefix \"Набор_категорий_3\" accept
```

3. Добавить правило: не пропускать исходящие пакеты, в которых для TCP-порта получателя задано значение 80 и которые имеют ненулевую классификационную метку:

```
sudo nft add rule inet filter output tcp dport 80 ip option \
    astra exists drop
```

4. Перенаправление входящих пакетов без классификационных меток на другой порт этого же хоста:

   а) создать цепочку с наименованием `before_routing`, которая имеет тип `nat`, приоритет `dstnat` и для которой назначен хук `prerouting` (обработка всех входящих пакетов до принятия решения о маршрутизации):

```
sudo nft add chain inet filter before_routing \
    { type nat hook prerouting priority dstnat \; }
```

б) в цепочку `before_routing` добавить правило: входящие пакеты, в которых для TCP-порта получателя задано значение 80 и которые не имеют классификационную метку, перенаправлять на TCP-порт 8080:

```
sudo nft add rule inet filter before_routing ip option \
    astra missing tcp dport 80 redirect to 8080
```

2.6. Пункт «16.1.1. Режимы функционирования»

В пункте 16.1.1 после четвертого абзаца добавить новый абзац в следующей редакции:

Результаты неудачных проверок цифровых подписей файлов (подпись неверна или отсутствует) заносятся в метаданные файлов и сохраняются в ядре до перезагрузки ОС. При последующих обращениях к таким файлам повторная проверка их подписей не выполняется и считается неудачной, даже если файл после первой проверки был подписан верной подписью. Информация о неудачной проверке цифровой подписи файла удаляется в следующих случаях:

- после перезагрузки ОС;
- при удалении файла или его копировании в другой файл с последующим удалением исходного файла:

```
cp <исходный_файл> <новый_файл> && rm <исходный_файл> && \
     mv <новый_файл> <исходный_файл>
```

- при открытии файла на запись даже без его изменения, например командой:

```
echo -n "" >> <имя_файла>
```

2.7. Пункт «16.6.3. Установка квот на использование системных ресурсов»

В конце пункта 16.6.3 добавить новый абзац в следующей редакции:

Дополнительные ограничения на получение доступа и выполнение кода на уровне ядра ОС для привилегированного пользователя обеспечиваются модулем lockdown в соответствии с 16.8.

2.8. Пункт «16.6.4. Запрет установки бита исполнения»

В конце пункта 16.6.4 добавить новый абзац в следующей редакции:

ВНИМАНИЕ! При включенном запрете установки бита исполнения запускаемым сценариям не передаются значения переменных окружения, позволяющих передать интерпретатору произвольный сценарий (`PYTHONPATH`, `PERL5LIB`, `NODE_OPTIONS` и др.). Для получения значений переменных окружения сценарии должны запускаться соответствующими интерпретаторами.

Пример

```
python3 script.py
```

2.9. Пункт «16.6.7. Блокировка интерпретаторов»

В пункте 16.6.7 в список блокируемых интерпретаторов добавить интерпретатор `mksh`.

2.10. Пункт «16.6.36. Блокировка загрузки и выгрузки модулей ядра»

Ввести новый пункт 16.6.36 с соответствующим изменением нумерации последующих пунктов:

Инструмент командной строки `astra-modban-lock` позволяет блокировать загрузку и выгрузку модулей ядра. При использовании инструмента изменяется значение параметра ядра `kernel.modules_disabled`.

Параметры вызова, используемые данным инструментом, приведены в таблице 69.

Блокировка вступает в действие немедленно. После снятия блокировки для применения изменений требуется перезагрузка.

Описание инструмента приведено на справочной странице `man astra-modban-lock`.

2.11. Подраздел «16.8. Модуль безопасности lockdown»

Ввести новый подраздел 16.8 с соответствующим изменением нумерации следующих подразделов:

Модуль безопасности lockdown устанавливает для привилегированного пользователя запрет на процедуры получения доступа и выполнения кода на уровне ядра ОС, которые в штатном режиме он может осуществить, например, с помощью подмены ядра ОС (используя системный вызов `kexec`) или через `/dev/kmem`, читая/записывая данные в память.

Модуль lockdown поддерживает два режима работы:

1) режим обеспечения целостности ядра ОС (режим `integrity`), включающий следующие ограничения:

а) LOCKDOWN_KEXEC — запрет выполнения системного вызова kexec(), обеспечивающего подмену текущего ядра ОС;

б) LOCKDOWN_HIBERNATION — запрет гибернации, предотвращающий возможность подмены ядра ОС при выходе из нее;

в) LOCKDOWN_MODULE_SIGNATURE — включение проверки подписей модулей ядра ОС;

г) LOCKDOWN_DEV_MEM — запрет чтения и записи в /dev/mem, /dev/kmem, /dev/port;

д) LOCKDOWN_PCI_ACCESS — блокировка оборудования, которое потенциально может генерировать прямую адресацию памяти (DMA);

е) LOCKDOWN_IOPORT — блокировка ioctl-вызывов (ioctl_console) KDADDIO, KDDELIO, KDENABIO и KDDISABIO для терминалов и виртуальных консолей;

ж) LOCKDOWN_EFI_TEST — запрет чтения /dev/efi_test;

з) LOCKDOWN_ACPI_TABLES — ограничение доступа к интерфейсам ACPI;

2) режим обеспечения конфиденциальности некоторых компонентов ядра ОС (режим confidentiality), включающий все ограничения режима integrity, а также:

а) LOCKDOWN_KCORE — запрет чтения /proc/kcore;

б) LOCKDOWN_KPROBES — ограничение доступа к отладочному режиму kprobes;

в) LOCKDOWN_BPF_READ — ограничение доступа к механизму фильтрации пакетов BPF;

г) LOCKDOWN_TRACEFS — ограничение доступа к ФС tracefs.

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.

2.12. Подраздел «17.5. Возможности по защите от угроз безопасности информации средствами защиты ОС»

2.12.1. В подразделе 17.5 ввести новый подпункт 17.5.1.10 с соответствующим изменением нумерации следующих подпунктов:

Модуль безопасности lockdown обеспечивает запрет получения несанкционированного доступа и выполнения кода на уровне ядра ОС, в том числе со стороны привилегированного пользователя (см. 16.8).

2.12.2. В подразделе 17.5 ввести новый подпункт 17.5.2.10 с соответствующим изменением нумерации следующих подпунктов:

Модуль безопасности lockdown обеспечивает запрет получения несанкционированного доступа и выполнения кода на уровне ядра ОС, в том числе со стороны привилегированного пользователя (см. 16.8).

2.12.3. В подразделе 17.5 ввести новый подпункт 17.5.3.6 с соответствующим изменением нумерации следующих подпунктов:

Модуль безопасности lockdown в том числе обеспечивает запрет задания потенциально опасных параметров загрузки ядра ОС, а также выполнения ряда небезопасных системных вызовов и прямого доступа на запись (через специальные файлы устройств) к пространству ядра ОС со стороны привилегированного пользователя (см. 16.8).

2.12.4. В подразделе 17.5 ввести новый подпункт 17.5.8.6 с соответствующим изменением нумерации следующих подпунктов:

Модуль безопасности lockdown обеспечивает запрет получения несанкционированного доступа к памяти и выполнения кода на уровне ядра ОС, в том числе со стороны привилегированного пользователя (суперпользователя root) в соответствии с 16.8.

2.12.5. В подразделе 17.5 ввести новый подпункт 17.5.9.7 с соответствующим изменением нумерации следующих подпунктов:

Модуль безопасности lockdown обеспечивает запрет получения несанкционированного доступа (чтение/запись) к памяти и выполнения кода на уровне ядра ОС, в том числе со

стороны привилегированного пользователя (суперпользователя `root`) в соответствии с 16.8.

2.12.6. В подразделе 17.5 ввести новый подпункт 17.5.12.7 с соответствующим изменением нумерации следующих подпунктов:

Модуль безопасности lockdown обеспечивает запрет получения несанкционированного доступа (чтение/запись) к памяти и выполнения кода на уровне ядра ОС, прямого доступа к портам ввода-вывода `ioport` и т. п., в том числе со стороны привилегированного пользователя (суперпользователя `root`) в соответствии с 16.8.

2.12.7. В подразделе 17.5 ввести новый подпункт 17.5.13.5 с соответствующим изменением нумерации следующих подпунктов:

Модуль безопасности lockdown обеспечивает запрет получения несанкционированного доступа (чтение/запись) к памяти и выполнения кода на уровне ядра ОС, прямого доступа к портам ввода-вывода `ioport` и т. п., в том числе со стороны привилегированного пользователя (суперпользователя `root`) в соответствии с 16.8.

2.13. Подраздел «18.2. Указания по эксплуатации ОС»

В подразделе 18.2 в пункте 18.2.1 перечисление 2а) изложить в редакции:

> 2) должна быть настроена политика паролей. Выполнить настройку возможно либо с помощью графической утилиты `astra-systemsettings` («Параметры системы», см. электронную справку), запущенной от имени администратора через механизм `sudo`, либо путем редактирования файлов сценариев:
>
> а) в файле `/etc/pam.d/common-password` указать параметры с требованиями к сложности пароля, отредактировав строку:
>
> ```
> password requisite pam cracklib.so minlen=<минимальная_длина_пароля>
> dcredit=-1 ucredit=-1 lcredit=-1
> ```