Безопасная настройка Astra Linux SE

Лабораторные задания для слушателей курса повышения квалификации

Версии ОС	Astra Linux SE 1.7, 1.8
Темы	Парольная политика, SSH, Apache2
Адрес лендинга	astra-hardening.pikov.expert
Формат	Три лабораторные работы: **легкая**, **средняя**, **сложная**

Материал рассчитан на специалистов с высшим образованием и начальными знаниями Astra Linux.

Перед каждой работой рекомендуется создать снимок виртуальной машины.

Содержание комплекта

N	Лабораторная работа	Сложность	Основной результат
1	Настройка парольной политики	**Легкая**	Отклоняются простые пароли, заданы сроки и история паролей.
2	Организация безопасного доступа по SSH	**Средняя**	Работает вход по ключу, парольный вход и root-вход ограничены.
3	Установка Apache и настройка безопасности	**Сложная**	Работает отдельный virtual host, HTTPS и базовое усиление Apache.

Лабораторная работа 1. Настройка парольной политики Astra Linux SE

Сложность: **легкая**

Время: 30-45 минут

Целевая ОС: Astra Linux SE 1.7 или 1.8

Формат: индивидуальная работа на учебной виртуальной машине

Цель

Настроить базовые параметры парольной политики для локальных пользователей Astra Linux SE и проверить, что система отклоняет слишком простые пароли.

После выполнения работы слушатель должен уметь:

- находить параметры политики учетных записей в графическом интерфейсе Astra Linux;
- настраивать минимальную длину и сложность пароля;
- проверять параметры через конфигурационные файлы;
- объяснять, зачем нужны ограничения на повторное использование и срок действия паролей.

Исходные условия

У вас есть учебная виртуальная машина с Astra Linux SE 1.7 или 1.8. Учетная запись слушателя должна входить в группу администраторов и иметь возможность выполнять команды через `sudo`.

Рекомендуемый режим защищённости для выполнения лабораторной — «Воронеж» (Усиленный) или «Орёл» (Базовый). В режиме «Смоленск» (Максимальный) возможны ограничения со стороны мандатного контроля целостности (МКЦ) и замкнутой программной среды (ЗПС) при правке системных конфигурационных файлов.

Перед началом работы создайте снимок виртуальной машины.

Выберите один способ настройки: вариант А через графический интерфейс или вариант В через терминал. Не смешивайте оба варианта в одной попытке: при последующем запуске графического менеджера безопасности ручные изменения в PAM-конфигурации могут быть перезаписаны штатными настройками Astra Linux.

Что нужно настроить

Настройте для локальных пользователей следующую политику:

Параметр	Требуемое значение
Минимальная длина пароля	8 символов
Минимум строчных букв	1
Минимум заглавных букв	1
Минимум цифр	1
Минимум специальных символов	1
Проверка имени пользователя в пароле	включена

Проверка GECOS в пароле	включена
Проверка сложности для root	включена
История паролей	запоминать 5 последних паролей
Минимальный срок действия пароля	1 день
Максимальный срок действия пароля	60 дней
Предупреждение до истечения пароля	7 дней

Вариант A. Настройка через графический интерфейс

1. Откройте меню настроек безопасности:

`Пуск` -> `Системные` -> `Политика безопасности` или `Менеджер безопасности`.

В разных сборках Astra Linux SE 1.7 и 1.8 название пункта меню может немного отличаться. Если пункт не найден, запустите менеджер безопасности из терминала:

```
fly-admin-smc
```

2. Откройте раздел `Сложность`.
3. Включите проверку имени пользователя и GECOS.
4. Установите минимальную длину пароля `8`.
5. Включите требования к составу пароля:

- строчные буквы: `1`;
- заглавные буквы: `1`;
- цифры: `1`;
- другие символы: `1`.

6. Включите применение проверки сложности для пользователя `root`.
7. Откройте раздел `История`.
8. Включите поддержку истории паролей, применение для `root` и укажите `5` последних паролей.
9. Откройте раздел `Срок действия`.
10. Установите минимальный срок действия `1`, максимальный срок действия `60`, предупреждение `7`.

Вариант B. Настройка через терминал

Используйте терминал, если графический интерфейс недоступен. Сначала сделайте резервные копии файлов:

```
sudo cp /etc/pam.d/common-password /etc/pam.d/common-password.bak
sudo cp /etc/login.defs /etc/login.defs.bak
```

Если в вашей версии Astra Linux доступна штатная утилита `astra-passwd-policy`, ее можно использовать как альтернативный административный способ применения парольной политики. В этой лабораторной работе основной CLI-вариант показан через PAM-файлы, чтобы слушатель видел, какие параметры реально применяются системой.

Найдите строку с модулем `pam_pwquality.so`:

```
grep pam_pwquality.so /etc/pam.d/common-password
```

Отредактируйте файл:

```
sudo nano /etc/pam.d/common-password
```

В строке `pam_pwquality.so` задайте параметры:

```
minlen=8 lcredit=-1 ucredit=-1 dcredit=-1 ocredit=-1 usercheck=1 gecoscheck enforce_for_root
```

Если в файле нет строки с `pam_pwhistory.so`, добавьте после строки `pam_pwquality.so`:

```
password requisite pam_pwhistory.so use_authtok enforce_for_root remember=5
```

Настройте сроки действия паролей:

```
sudo nano /etc/login.defs
```

Проверьте или измените значения:

```
PASS_MIN_DAYS   1
PASS_MAX_DAYS   60
PASS_WARN_AGE   7
```

Проверка

Проверьте параметры сложности:

```
grep pam_pwquality.so /etc/pam.d/common-password
grep pam_pwhistory.so /etc/pam.d/common-password
grep -E 'PASS_MIN_DAYS|PASS_MAX_DAYS|PASS_WARN_AGE' /etc/login.defs
```

Создайте тестового пользователя:

```
sudo adduser student-test
```

При создании пользователя adduser запросит пароль. Если на вашей версии Astra adduser не вызвал проверку pam_pwquality (зависит от версии PAM-модулей), задайте начальный простой пароль и продолжите проверкой через passwd — это гарантированный способ убедиться, что политика работает:

sudo passwd student-test

Попробуйте назначить простой пароль (короче 8 символов или без цифр). Система должна отказать в установке такого пароля с сообщением вида «BAD PASSWORD: ...».

Затем назначьте пароль, который соответствует политике. Например, используйте учебный пароль вида `Astra!2026`. Не применяйте этот пароль в реальной эксплуатации.

После проверки удалите тестового пользователя:

```
sudo deluser --remove-home student-test
```

Что нужно сдать

Подготовьте короткий отчет:

11. Версия Astra Linux SE: 1.7 или 1.8.
12. Скриншот или текстовый вывод строки `pam_pwquality.so`.
13. Скриншот или текстовый вывод строки `pam_pwhistory.so`.
14. Вывод параметров `PASS_MIN_DAYS`, `PASS_MAX_DAYS`, `PASS_WARN_AGE`.
15. Краткое объяснение, почему пароль `student123` не является достаточным для защищенной системы.

Контрольные вопросы

16. Почему недостаточно задать только минимальную длину пароля?
17. Зачем включать проверку имени пользователя в пароле?
18. Чем полезна история паролей?
19. Почему прямой вход под `root` обычно ограничивают?

Критерии зачета

Работа считается выполненной, если парольная политика настроена, простой пароль отклоняется, сложный пароль принимается, а слушатель может показать, в каких файлах или графических разделах находятся примененные параметры.

Использованные материалы

- Методические рекомендации по настройке Astra Linux SE 1.7 и 1.8.
- Параметры `pam_pwquality.so`, `pam_pwhistory.so` и `login.defs` из разделов политики учетных записей Astra Linux SE.
- Штатные инструменты Astra Linux для управления политикой безопасности: `fly-admin-smc`, при наличии - `astra-passwd-policy`.

Лабораторная работа 2. Организация безопасного доступа по SSH

Сложность: **средняя**

Время: 45-60 минут

Целевая ОС: Astra Linux SE 1.7 или 1.8

Формат: работа на двух учебных виртуальных машинах

Цель

Настроить безопасный удаленный доступ по SSH между двумя узлами Astra Linux SE: включить службу, создать ключи, настроить вход по ключам и ограничить небезопасные варианты подключения.

После выполнения работы слушатель должен уметь:

- проверять состояние службы SSH;
- создавать пару SSH-ключей;
- переносить публичный ключ на удаленный узел;
- входить на сервер без ввода пароля;
- настраивать базовые параметры безопасности `sshd_config`;
- проверять, что SSH-доступ работает ожидаемо.

Схема стенда

Используйте две виртуальные машины:

Роль	Имя	Пример IP-адреса
Клиент	`infra`	`10.0.1.254`
Сервер	`server1`	`10.0.1.1`

Если в вашем классе используются другие адреса, замените IP-адреса в командах на свои.

Учетная запись для работы: `sa`.

Рекомендуемый режим защищённости для выполнения лабораторной — «Воронеж» (Усиленный) или «Орёл» (Базовый). В режиме «Смоленск» (Максимальный) возможны ограничения со стороны мандатного контроля целостности (МКЦ) и замкнутой программной среды (ЗПС) при правке системных конфигурационных файлов или установке пакетов.

Перед началом создайте снимки обеих виртуальных машин.

Часть 1. Проверка службы SSH

На сервере `server1` проверьте службу:

```
sudo systemctl status ssh
```

Если служба не запущена:

```
sudo systemctl start ssh
sudo systemctl enable ssh
```

Проверьте, что сервер слушает порт SSH:

```
ss -tlnp | grep :22
```

С клиента `infra` проверьте сетевую доступность сервера:

```
ping 10.0.1.1
```

Часть 2. Создание ключей на клиенте

На `infra` под пользователем `sa` создайте пару ключей:

```
ssh-keygen -t rsa -b 3072
```

При выполнении команды:

- путь к файлу ключа можно оставить по умолчанию;
- парольную фразу в учебной работе можно задать или оставить пустой по указанию преподавателя;
- существующий ключ не перезаписывайте без согласования.

Проверьте, что ключи созданы:

```
ls -l ~/.ssh/
```

Ожидаемые файлы:

- `id_rsa` - закрытый ключ, его нельзя передавать другим людям;
- `id_rsa.pub` - открытый ключ, его можно размещать на сервере.

Часть 3. Копирование ключа на сервер

С клиента `infra` скопируйте публичный ключ на `server1`:

```
ssh-copy-id sa@10.0.1.1
```

В первый раз система может спросить, доверяете ли вы ключу удаленного хоста. Сравните адрес и подтвердите подключение.

Проверьте вход:

```
ssh sa@10.0.1.1
```

Если все настроено корректно, вход должен пройти без ввода пароля пользователя `sa` на сервере.

Завершите удаленную сессию:

```
exit
```

Часть 4. Безопасная настройка SSH-сервера

На `server1` сделайте резервную копию конфигурации:

```
sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.bak
```

Откройте файл:

```
sudo nano /etc/ssh/sshd_config
```

Проверьте или добавьте параметры:

```
PermitRootLogin no
PubkeyAuthentication yes
PasswordAuthentication no
X11Forwarding no
MaxAuthTries 3
ClientAliveInterval 300
ClientAliveCountMax 2
AllowUsers sa
```

Смысл параметров:

Параметр	Назначение
`PermitRootLogin no`	запрещает прямой вход под `root`
`PubkeyAuthentication yes`	разрешает вход по ключам

`PasswordAuthentication no`	отключает вход по паролю
`X11Forwarding no`	отключает проброс графических приложений
`MaxAuthTries 3`	ограничивает число попыток аутентификации
`ClientAliveInterval 300`	проверяет активность клиента каждые 300 секунд
`ClientAliveCountMax 2`	завершает зависшие сессии после двух неудачных проверок
`AllowUsers sa`	разрешает SSH-вход только указанному пользователю

Если на сервере есть другие пользователи, которым нужен SSH-доступ, добавьте их в `AllowUsers` через пробел. Например:

```
AllowUsers sa student admin
```

Перед перезапуском проверьте синтаксис:

```
sudo sshd -t
```

Если команда не вывела ошибок, перезапустите SSH:

```
sudo systemctl restart ssh
```

Часть 5. Проверка ограничений

С клиента `infra` проверьте вход по ключу:

```
ssh sa@10.0.1.1
```

Проверьте, что вход по паролю отключен:

```
ssh -v -o PubkeyAuthentication=no sa@10.0.1.1
```

Ожидаемый результат: подключение не должно пройти по паролю. В диагностическом выводе `-v` должно быть видно, что сервер не предлагает парольный вход, например остается только метод `publickey`.

Проверьте запрет прямого входа под `root`:

```
ssh root@10.0.1.1
```

Ожидаемый результат: вход должен быть запрещен.

Аварийное восстановление

Если после изменения `sshd_config` подключение перестало работать, войдите на `server1` через консоль виртуальной машины и восстановите резервную копию:

```
sudo cp /etc/ssh/sshd_config.bak /etc/ssh/sshd_config
sudo systemctl restart ssh
```

Что нужно сдать

Подготовьте короткий отчет:

20. IP-адрес клиента и сервера.
21. Вывод `systemctl status ssh` на сервере.
22. Вывод `ssh sa@10.0.1.1`, подтверждающий вход по ключу.
23. Фрагмент `sshd_config` с настроенными параметрами.
24. Результат проверки `ssh -v -o PubkeyAuthentication=no sa@10.0.1.1`.

Контрольные вопросы

25. Почему закрытый ключ нельзя копировать на сервер?

26. Чем вход по ключу безопаснее входа по паролю?
27. Почему прямой SSH-вход под `root` лучше запретить?
28. Что произойдет, если отключить `PasswordAuthentication`, но не скопировать ключ заранее?

Критерии зачета

Работа считается выполненной, если слушатель может подключиться к серверу по ключу, вход по паролю запрещен, прямой вход под `root` запрещен, а конфигурация проходит проверку `sudo sshd -t`.

Использованные материалы

- Сборник практических заданий, модуль 2: настройка удаленного доступа по SSH.
- Практики безопасной настройки службы OpenSSH для учебного стенда Astra Linux SE.

Лабораторная работа 3. Установка Apache и настройка безопасности веб-сервера

Сложность: **сложная**

Время: 90-120 минут

Целевая ОС: Astra Linux SE 1.7 или 1.8

Формат: работа на одной или двух учебных виртуальных машинах

Цель

Установить Apache2 на Astra Linux SE, создать отдельный виртуальный хост, ограничить лишние возможности сервера, включить HTTPS с самоподписанным сертификатом и проверить доступность сайта.

После выполнения работы слушатель должен уметь:

- устанавливать Apache2 и управлять службой;
- создавать отдельный каталог сайта и виртуальный хост;
- отключать сайт по умолчанию;
- настраивать базовые защитные HTTP-заголовки;
- ограничивать отображение служебной информации Apache;
- выпускать учебный самоподписанный сертификат;
- проверять HTTP и HTTPS с помощью `curl`.

Схема стенда

Основной вариант - одна виртуальная машина:

Роль	Имя	Пример IP-адреса
Веб-сервер	`infra`	`10.0.1.254`

Дополнительная проверка может выполняться со второй машины `server1`.

Доменное имя для учебного сайта: `infra.astra.test`.

Если DNS в стенде не настроен, добавьте временную запись в `/etc/hosts` на клиенте:

```
10.0.1.254 infra.astra.test
```

Если IP-адрес вашей ВМ отличается от `10.0.1.254`, замените адрес во всех примерах, записях `/etc/hosts`, командах `curl` и в параметре `subjectAltName` сертификата. Если работа выполняется на одной машине без отдельной сети, можно использовать локальную запись:

```
127.0.0.1 infra.astra.test
```

Рекомендуемый режим защищённости для выполнения лабораторной — «Воронеж» (Усиленный) или «Орёл» (Базовый). В режиме «Смоленск» (Максимальный) возможны ограничения со стороны мандатного контроля целостности (МКЦ) и замкнутой программной среды (ЗПС) при правке системных конфигурационных файлов или установке пакетов.

Рекомендуемая версия Astra Linux SE: 1.7.4 или новее, либо 1.8. На старых обновлениях 1.7 (до 1.7.4) Apache может не учитывать значение AstraMode, заданное во включаемом конфигурационном файле — в этом случае перенесите директиву AstraMode off непосредственно в /etc/apache2/apache2.conf или в блок <VirtualHost>.

Перед началом создайте снимок виртуальной машины.

Часть 1. Установка Apache2

Перед установкой уточните IP-адрес вашей ВМ:

```
ip -4 addr
```

Далее в лабораторной работе используется пример `10.0.1.254`. При другом адресе подставляйте свой IP.

Обновите сведения о пакетах:

```
sudo apt update
```

Установите Apache2:

```
sudo apt install apache2
```

Проверьте службу:

```
sudo systemctl status apache2
sudo systemctl enable apache2
```

Проверьте, что сервер слушает порт 80:

```
sudo ss -tlnp | grep :80
```

Часть 2. Отключение AstraMode для Apache

Создайте отдельный конфигурационный файл:

```
sudo nano /etc/apache2/conf-available/astramode_off.conf
```

Добавьте строку:

```
AstraMode off
```

Активируйте конфигурацию:

```
sudo a2enconf astramode_off
sudo systemctl reload apache2
```

Пояснение: в учебной работе параметр используется для совместимости с типовыми практическими заданиями по Apache в Astra Linux SE. В реальной системе решение об отключении режима должно приниматься по требованиям эксплуатации и документации к конкретному контуру.

Часть 3. Создание виртуального хоста

Создайте каталог сайта:

```
sudo mkdir -p /var/www/infra
sudo chown -R root:www-data /var/www/infra
sudo find /var/www/infra -type d -exec chmod 750 {} \;
```

Создайте главную страницу:

```
sudo nano /var/www/infra/index.html
```

Содержимое:

```
<!doctype html>
<html lang="ru">
<head>
  <meta charset="utf-8">
  <title>infra.astra.test</title>
</head>
<body>
  <h1>infra.astra.test</h1>
  <p>Учебный защищенный веб-сервер Astra Linux SE.</p>
</body>
</html>
```

Ограничьте права на файл:

```
sudo chown root:www-data /var/www/infra/index.html
sudo chmod 640 /var/www/infra/index.html
```

Создайте конфигурацию виртуального хоста:

```
sudo nano /etc/apache2/sites-available/infra.conf
```

Содержимое:

```
<VirtualHost *:80>
    ServerName infra.astra.test
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/infra

    <Directory /var/www/infra>
        Options -Indexes -FollowSymLinks
        AllowOverride None
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/infra_error.log
    CustomLog ${APACHE_LOG_DIR}/infra_access.log combined
</VirtualHost>
```

Активируйте сайт и отключите сайт по умолчанию:

```
sudo a2ensite infra
sudo a2dissite 000-default.conf
sudo apache2ctl configtest
sudo systemctl reload apache2
```

Проверьте:

```
curl http://infra.astra.test
curl http://10.0.1.254
```

Часть 4. Базовое усиление конфигурации Apache

Создайте файл с параметрами безопасности:

```
sudo nano /etc/apache2/conf-available/security-hardening.conf
```

Добавьте:

```
ServerTokens Prod
ServerSignature Off
TraceEnable Off

Header always set X-Content-Type-Options "nosniff"
Header always set X-Frame-Options "SAMEORIGIN"
Header always set Referrer-Policy "no-referrer"
Header always set Permissions-Policy "geolocation=(), microphone=(), camera=()"
```

Включите модуль заголовков и конфигурацию:

```
sudo a2enmod headers
sudo a2enconf security-hardening
sudo apache2ctl configtest
sudo systemctl reload apache2
```

Проверьте заголовки:

```
curl -I http://infra.astra.test
```

Проверьте, что листинг каталога не работает. Создайте пустой каталог без `index.html`:

```
sudo mkdir /var/www/infra/test-no-index
sudo chown root:www-data /var/www/infra/test-no-index
sudo chmod 750 /var/www/infra/test-no-index
```

Затем выполните:

```
curl -I http://infra.astra.test/test-no-index/
```

Ожидаемый результат: не должно быть открытого списка файлов.

Часть 5. Настройка HTTPS с самоподписанным сертификатом

Создайте каталог для ключей:

```
sudo mkdir -p /etc/apache2/ssl
sudo chmod 700 /etc/apache2/ssl
```

Создайте закрытый ключ и сертификат:

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 \
  -keyout /etc/apache2/ssl/infra.key \
  -out /etc/apache2/ssl/infra.crt \
  -subj "/CN=infra.astra.test" \
  -addext "subjectAltName=DNS:infra.astra.test,IP:10.0.1.254"
```

Ограничьте права:

```
sudo chmod 600 /etc/apache2/ssl/infra.key
sudo chmod 644 /etc/apache2/ssl/infra.crt
sudo chown root:root /etc/apache2/ssl/infra.key /etc/apache2/ssl/infra.crt
```

Создайте HTTPS-виртуальный хост:

```
sudo nano /etc/apache2/sites-available/infra-ssl.conf
```

Содержимое:

```
<VirtualHost *:443>
    ServerName infra.astra.test
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/infra

    SSLEngine on
    SSLProtocol all -SSLv3 -TLSv1 -TLSv1.1
    SSLCipherSuite HIGH:!aNULL:!MD5:!3DES
    SSLHonorCipherOrder on
    SSLCertificateFile /etc/apache2/ssl/infra.crt
    SSLCertificateKeyFile /etc/apache2/ssl/infra.key

    <Directory /var/www/infra>
        Options -Indexes -FollowSymLinks
        AllowOverride None
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/infra_ssl_error.log
    CustomLog ${APACHE_LOG_DIR}/infra_ssl_access.log combined
</VirtualHost>
```

Включите SSL:

```
sudo a2enmod ssl
sudo a2ensite infra-ssl
sudo apache2ctl configtest
sudo systemctl restart apache2
```

Проверьте:

```
curl -k https://infra.astra.test
curl -k -I https://infra.astra.test
```

Ключ `-k` нужен только потому, что сертификат самоподписанный и не доверен клиенту.

Бонусное задание для самостоятельной проверки: объясните, зачем в HTTPS-виртуальном хосте используются директивы `SSLProtocol`, `SSLCipherSuite` и `SSLHonorCipherOrder`, и какие устаревшие протоколы они исключают.

Часть 6. Дополнительное ограничение доступа

Если по заданию преподавателя веб-сервер должен быть доступен только локально, измените `/etc/apache2/ports.conf`:

```
Listen 127.0.0.1:80
Listen 127.0.0.1:443
```

После изменения:

```
sudo apache2ctl configtest
sudo systemctl restart apache2
sudo ss -tlnp | grep apache2
```

Проверьте локальный доступ:

```
curl http://127.0.0.1
curl -k https://127.0.0.1
```

Проверьте, что по сетевому адресу сайт недоступен:

```
curl http://10.0.1.254
```

Ожидаемый результат: подключение по сетевому адресу должно завершиться ошибкой, например `curl: (7) Failed to connect to 10.0.1.254 port 80: Connection refused`. Если у вашей ВМ другой IP-адрес, в сообщении будет указан ваш адрес.

Что нужно сдать

Подготовьте отчет:

29. Вывод `systemctl status apache2`.
30. Файл или фрагмент `/etc/apache2/sites-available/infra.conf`.
31. Файл или фрагмент `/etc/apache2/conf-available/security-hardening.conf`.
32. Вывод `apache2ctl configtest`.
33. Вывод `curl -I http://infra.astra.test`.
34. Вывод `curl -k -I https://infra.astra.test`.
35. Краткое объяснение, зачем отключены `Indexes`, `ServerSignature`, `TraceEnable` и прямое раскрытие версии Apache.

Контрольные вопросы

36. Чем виртуальный хост отличается от сайта по умолчанию?
37. Почему не стоит оставлять листинг каталогов включенным?
38. Почему самоподписанный сертификат вызывает предупреждение браузера?
39. Какие риски снижает заголовок `X-Content-Type-Options: nosniff`?
40. Почему перед перезапуском Apache важно выполнять `apache2ctl configtest`?

Критерии зачета

Работа считается выполненной, если Apache установлен, сайт `infra.astra.test` открывается по HTTP и HTTPS, сайт по умолчанию отключен, конфигурация проходит `apache2ctl configtest`, листинг каталогов не раскрывает файлы, а HTTP-ответ содержит настроенные защитные заголовки.

Использованные материалы

- Сборник практических заданий, модуль 9: веб-сервер на основе Apache.
- Фрагменты практики из модуля 2: локальное ограничение Apache и доступ через SSH-туннель.
- Методические рекомендации по настройке Astra Linux SE 1.7 и 1.8.