Ссылки на источники для
вопросов к экзамену по курсу Безопасность Операционных Систем
январь 2026

Вопросы

Физическая безопасность сервера на базе ОС Linux

1. **Сброс пароля пользователя ОС Linux: Single User Mode и внедрение в ход загрузки ОС. Оболочки `/sbin/sushell` и `/sbin/sulogin`.**
2. **Способы защиты от сброса пароля пользователя ОС Linux.**
3. **Прозрачное шифрование дисков и файловых контейнеров. Служба `dm-crypt` и подсистема LUKS.**
   - конспект [БезОС] Тема 1. Физическая безопасность сервера на базе ОС Linux.pdf
   - ссылки в конспекте

Управление и безопасность УЗ пользователя в ОС Linux

По всему разделу:
- конспект [БезОС] Тема 2. Механизмы безопасности и управления УЗ пользователя в ОС Linix.pdf
- методичка Lab1.pdf
- ссылки в конспекте и методичке

4. **Пользователи в ОС Linux. Файл паролей `/etc/passwd` и теневой файл паролей `/etc/shadow`, их структура и назначение.**
5. **Группы пользователей в ОС Linux. Назначение групп пользователей, файл `/etc/group`.**
6. **Хранение паролей в ОС Linux. Структура записи хэша пароля. Виды алгоритмов хэширования паролей.**
   - [Керриск, глава 8]
   - [cyberciti.biz] Understanding /etc/passwd File Format
   - [cyberciti.biz] Understanding /etc/shadow file format on Linux
   - [cyberciti.biz] Understanding /etc/group File in Linux

7. **Система подключаемых модулей аутентификации (Pluggable Authentication Module, PAM). Основные компоненты архитектуры PAM и их назначение. Типичное расположение модулей и конфигурационных файлов в ОС Linux.**
8. **Конфигурация системы PAM для конкретного сервиса. Формат конфигурационного файла PAM, группы управления и контрольные флаги (сокращённая и подробная форма).**
9. **Исполнение сценария PAM: порядок запуска модулей, обработка результатов исполнения модулей. Понятие стека модулей.**
   - The Linux-PAM System Administrators' Guide
   - [freebsd.org] FreeBSD жив. Подключаемые Модули Аутентификации (PAM)
   - [redos.red-soft.ru] Подключаемые модули аутентификации (PAM)

10. **Типичная конфигурация подсистемы аутентификации и авторизации ОС Linux на примере ОС Ubuntu 24.04.**
    - комментарии в конфиг. файлах `/etc/pam.d/` типичной установки ОС Ubuntu 24.04:
        - `common-auth`
        - `common-account`
        - `common-password`
        - `common-session`
        - `login`
        - `sshd`

11. **Реализация классической модели парольной аутентификации Unix (с помощью файлов `/etc/passwd` и `/etc/shadow`) в ОС Linux средствами системы PAM.**
    - pam_unix(8) — Linux manual page
    - [redhat.com] Sample PAM Configuration Files

12. **Способы блокировки пользователей средствами системы PAM.**
    - примеры 1-3 в конспекте по Теме 2, слайды 6-8
    - pam_time(8) — Linux manual page
    - time.conf(5) — Linux manual page
    - pam_limits(8) — Linux manual page
    - limits.conf(5) — Linux manual page
    - что-то ещё?..

13. **Конфигурация парольной политики средствами системы PAM.**
    - пример 5 в конспекте по Теме 2, слайды 10-11
    - pam_pwquality(8) — Linux man page

14. **Поддержка аутентификации и авторизации в собственных приложениях средствами системы PAM: программный интерфейс и конфигурация сценария.**
    - [medium.com] Linux PAM — How to create an authentication module
    - [github.com] linux-pam Examples

Механизмы разграничения доступа ОС Linux

По всему разделу:
- конспект [БезОС] Тема 3. Механизмы разграничения доступа ОС Linux.pdf
- ссылки в конспекте

15. **Классическая модель разграничения доступа Unix. Триплеты прав, расширенные права доступа (`sticky bit`, `SUID`, `SGID`).**
    - [Керриск, глава 15]
    - конспект по Теме 3, раздел 1
    - [wikipedia.org] File system permissions
    - [wiki.archlinux.org] File permissions and attributes
    - [wiki.archlinux.org] umask

- [wikipedia.org] setuid
- [wikipedia.org] sticky bit

16. **Файловые списки управления доступом (Access Control List, ACL, `facl`). Назначение, типы списков, примеры использования.**
    - [Керриск, глава 17]
    - конспект по Теме 3, раздел 4
    - [wikipedia.org] Access control list
    - [wiki.archlinux.org] Access Control Lists
    - RHEL blog. Linux Access Control Lists

17. **Атрибуты файла (file attributes). Назначение, основные типы атрибутов. Примеры использования атрибутов `append only`, `immutable` и `secure deletion`.**
    - конспект по Теме 3, раздел 2
    - [wikipedia.org] File attribute
    - [wiki.archlinux.org] File attributes

18. **Идентификаторы пользователя и группы запущенного процесса (`real`, `effective`, `saved IDs`). Алгоритм определения права доступа к объекту ОС Linux.**
    - [Керриск, глава 9]
    - [Керриск, раздел 15.4]
    - [Керриск, раздел 17.2]

19. **Эффект битов `SUID` и `SGID` исполняемого файла на значения идентификаторов процесса. Назначение `setuid-root` программ. Написание защищённых привилегированных приложений: концепция защищённого программирования, secure programming.**
    - [Керриск, раздел 38]

20. **Linux Capabilities. Назначение, типы разрешений. Механизмы работы с привилегиями.**
    - [Керриск, глава 39]
    - [habr.com] В двух словах о привилегиях Linux (capabilities)
    - [wiki.archlinux.org] Capabilities
    - capabilities(7) — Linux manual page

Разграничение ресурсов и изоляция процессов ОС Linux

По всему разделу:
- конспект [БезОС] Тема 4. Разграничение ресурсов ОС Linux.pdf
- конспект [БезОС] Тема 5. Изоляция процессов ОС Linux.pdf
- ссылки в конспекте

21. **Контроль выделенных пользователю ресурсов средствами системы `ulimit`. Основные типы ресурсов, `hard` и `soft` лимиты. Управление лимитами ресурсов с помощью команды `ulimit`.**
    - конспект по Теме 4, раздел 1

- [securitylab.ru] Руководство по использованию команды ulimit в Linux с примерами
- [baeldung.com] Guide to Linux ulimit Command
- ulimit(3) — Linux manual page

22. **Управление лимитами ресурсов средствами системы PAM.**
    - конспект по Теме 2, слайд 9
    - limits.conf(5) — Linux manual page

23. **Механизм `chroot`: назначение, принцип работы, примеры использования.**
    - конспект по Теме 5, раздел 1
    - [wikipedia.org] chroot
    - [wiki.archlinux.org] chroot
    - [habr.com] Chroot(): первая попытка изоляции

24. **Механизм `seccomp`: назначение, принцип работы, примеры использования.**
    - [habr.com] Контейнеры и безопасность: seccomp
    - [lwn.net] A seccomp overview
    - seccomp(2) — Linux manual page

25. **Механизм пространств имён `namespaces`. Назначение, типы пространств имён, способы создания. Примеры использования пространств имён.**
    - конспект по Теме 5, раздел 2
    - [habr.com] Механизмы контейнеризации: namespaces
    - [lwn.net] Namespaces in operation: namespaces overview
    - [redhat.com] The 7 most used Linux namespaces
    - [redhat.com] Building a Linux container by hand using namespaces
    - namespaces(7) — Linux manual page

26. **PID namespace: назначение, принцип работы, способы создания и примеры использования.**
    - конспект по Теме 5, раздел 2
    - [redhat.com] Building containers by hand: The PID namespace
    - [lwn.net] Namespaces in operation: PID namespaces

27. **NET namespace: назначение, принцип работы, способы создания и примеры использования.**
    - конспект по Теме 5, раздел 2
    - [redhat.com] Building containers by hand using namespaces: The net namespace
    - [lwn.net] Namespaces in operation: Network namespaces

28. **MOUNT namespace: назначение, принцип работы, способы создания и примеры использования.**
    - конспект по Теме 5, раздел 2
    - [redhat.com] Building a container by hand using namespaces: The mount namespace
    - [lwn.net] Mount namespaces and shared subtrees
    - [lwn.net] Mount namespaces, mount propagation, and unbindable mounts

29. **Механизм `cgroup v2`, его назначение. Основные компоненты архитектуры системы: понятие контрольной группы и контроллера, интерфейс системы `cgroup`. Принцип единой иерархии контрольных групп.**
30. **Создание новой контрольной группы. Файлы `cgroup.controllers` и `cgroup.subtree_control`, их назначение. Добавление процессов в контрольную группу, файл `cgroup.procs`.**
31. **Настройка лимитов потребления CPU и привязка к подмножеству ядер (например, NUMA-узлу) группы процессов средствами системы `cgroup`.**
32. **Настройка лимитов потребления RAM и операций ввода-вывода (I/O) группы процессов средствами системы `cgroup`.**
    - конспект по Теме 4, раздел 2
    - [man7.org] Michael Kerrisk. An introduction to control groups (cgroups) v2
    - Linux Documentation. Control Group v2
    - [habr.com] Механизмы контейнеризации: cgroups

Безопасность процессов ОС Linux

По всему разделу:
- методичка Lab2.pdf
- методичка Lab3.pdf
- методичка Lab4.pdf
- [Yandex for Security] Андрей Ковалев. Безопасность бинарных приложений

33. **Переполнение на стеке: источники уязвимости, типичные способы эксплуатации. Понятие шелл-код, его назначение.**
    - методичка Lab3.pdf
    - [techorganic.com] 64-bit Linux stack smashing tutorial: Part 1
    - [phrack.org] Aleph One. Smashing The Stack For Fun And Profit
    - [Эриксон, раздел 0x320, глава 0x500]
    - [wikipedia.org] Stack buffer overflow

34. **Return oriented programming. Назначение техники эксплуатации, понятие гаджета и ROP-цепочки.**
    - методичка Lab4.pdf
    - [wikipedia.org] Return-oriented programming
    - [techorganic.com] 64-bit Linux stack smashing tutorial: Part 2
    - [techorganic.com] 64-bit Linux stack smashing tutorial: Part 3

35. **Уязвимость форматной строки: источники уязвимости, типичные способы эксплуатации.**
    - методичка Lab3.pdf
    - [Эриксон, раздел 0x350]
    - [ctf101.org] Format String Vulnerability
    - [owasp.org] Format string attack
    - [cs155.stanford.edu] Exploiting Format String Vulnerabilities

36. **Предотвращение исполнения данных (Data Execution Prevention, DEP). Назначение механизма защиты, способы реализации (аппаратные и программные).**
    - методичка Lab3.pdf
    - [wikipedia.org] Executable-space protection
    - [wikipedia.org] NX bit

37. **Рандомизация адресного пространства (Address Space Layout Randomization, ASLR). Назначение механизма защиты, способы реализации.**
    - методичка Lab3.pdf
    - [wikipedia.org] Address space layout randomization

38. **Позиционно-независимый код/исполняемый файл (Position-Independent Code, PIC, также Position-Independent Executables, PIE), его назначение.**
    - методичка Lab3.pdf
    - [wikipedia.org] Position-independent code
    - [mropert.github.io] PIC/PIE Sanitizers
    - [gentoo.org] Position Independent Code internals

39. **Защита от переполнения на стеке (Stack Smashing Protector, SSP). Назначение механизма защиты, способы реализации.**
    - методичка Lab3.pdf
    - [wikipedia.org] Buffer overflow protection
    - [wiki.osdev.org] Stack Smashing Protector

40. **Понятие безопасного стека, принцип разделения стека. Механизм LLVM/Clang Safe Stack.**
    - методичка Lab3.pdf
    - [clang.llvm.org] Safe Stack

41. **Контроль целостности потока исполнения (Control-Flow Integrity, CFI). Реализация CFI в LLVM/Clang.**
    - [wikipedia.org] Control-flow integrity
    - [clang.llvm.org] Control Flow Integrity

42. **Контроль целостности потока исполнения (Control-Flow Integrity, CFI). Механизм Intel Control-flow Enforcement Technology (CET).**
    - [wikipedia.org] Control-flow integrity
    - [wikipedia.org] Shadow stack
    - [wikipedia.org] Indirect branch tracking
    - [lwn.net] Indirect branch tracking for Intel CPUs
    - [intel.com] A Technical Look at Intel’s Control-flow Enforcement Technology
    - [microsoft.com] Understanding Hardware-enforced Stack Protection

Литература

1. Майкл Керриск. Linux API. Исчерпывающее руководство. Издательский дом Питер, 2018.
2. Джон Эриксон. Хакинг: искусство эксплойта, 2-е издание. Издательский дом Питер, 2022.