Лекция 2. Безопасность Unix систем

Структура лекции:

1. Семейства Unix систем
2. Безопасность из «коробки»
3. Немного о практике (разговор на лекции практических кейсах)

Семейства Linux.

На сегодняшний день существуют 7 семейств Unix систем:

1. Debian
2. Red Hat (RHEL)
3. Arch Linux
4. Gentoo
5. Slackware
6. SUSE
7. Independent (Независимые)

Безопасность Debian-based, Arch Linux-based, Slackware-based,

Debian-based дистрибутивы Linux, такие как **Debian**, **Ubuntu**, **Linux Mint**, и другие, известны своей надежностью и стабильностью, что также делает их хорошим выбором с точки зрения безопасности. Однако безопасность системы зависит от множества факторов, включая регулярное обновление, правильные настройки и использование встроенных инструментов. Рассмотрим основные аспекты безопасности Debian-based дистрибутивов:

1. Регулярные обновления

- **Debian** и его производные предлагают регулярные обновления безопасности для всех поддерживаемых пакетов. Эти обновления включают патчи для уязвимостей, которые обнаруживаются в операционной системе или стороннем ПО.
- Пользователям рекомендуется регулярно обновлять систему с помощью **APT**, чтобы убедиться, что все пакеты обновлены до последних стабильных и безопасных версий. Команда безопасности Debian следит за новыми уязвимостями и оперативно выпускает обновления.

Полное руководство по APT:

https://manpages.ubuntu.com/manpages/bionic/man8/apt.8.html

Команды для обновления системы:

bash
sudo apt update
sudo apt upgrade

2. Пакеты с долгосрочной поддержкой (LTS)

- **Debian Stable** и **Ubuntu LTS** фокусируются на долгосрочной поддержке (в случае Ubuntu — 5 лет), что делает их стабильными и безопасными для использования на серверах и в критически важных системах. Эти версии включают только проверенные и стабильные версии программного обеспечения, что минимизирует риски от уязвимостей.

3. AppArmor и SELinux

- В Debian-based дистрибутивах часто используется **AppArmor** (в Ubuntu он включен по умолчанию). AppArmor — это механизм контроля доступа на основе профилей, который ограничивает действия программ в системе, предотвращая несанкционированный доступ к системным ресурсам.

Подробно об AppArmor: https://gitlab.com/apparmor/apparmor/-/wikis/Documentation

Управление AppArmor:

bash
sudo systemctl status apparmor
sudo aa-status

- Также возможна установка и настройка **SELinux** (Security-Enhanced Linux), которая предлагает более сложную политику безопасности, но требует тщательной настройки.

Больше информации об SELinux: https://losst.pro/nastrojka-selinux

4. Шифрование данных

- Debian-based дистрибутивы поддерживают полное шифрование дисков с помощью **LUKS (Linux Unified Key Setup)** и **dm-crypt**. Это особенно важно для защиты конфиденциальных данных при физическом доступе к устройству.
- Во время установки можно выбрать шифрование домашнего каталога или всего диска, что добавляет дополнительный уровень безопасности.

Больше информации о LUKS: https://gitlab.com/cryptsetup/cryptsetup/

Больше информации о dn-crypt: https://en.wikipedia.org/wiki/Dm-crypt

5. Firewall (Межсетевой экран)

- В большинстве Debian-based систем встроен и легко настраивается межсетевой экран на основе **iptables** или его упрощенной версии **ufw (Uncomplicated Firewall)**, который можно использовать для фильтрации сетевого трафика и предотвращения несанкционированного доступа.
  - **UFW** установлен по умолчанию в Ubuntu и его производных, но может быть добавлен в других дистрибутивах.

Команды для настройки UFW:

bash
sudo ufw enable
sudo ufw allow ssh
sudo ufw deny 80

Подробнее об iptables: https://wiki.archlinux.org/title/Iptables
Подробнее об UFW: https://help.ubuntu.com/community/UFW

6. Регулярное обновление ядра и пакетов

- Debian и Ubuntu используют стабилизированные версии ядра Linux, но при необходимости пользователи могут обновлять ядро вручную. Это позволяет своевременно устранять уязвимости на уровне ядра.
- Существуют также **backports** — репозитории с более новыми версиями пакетов для стабильных релизов, которые можно использовать для установки критически важных обновлений без необходимости обновлять весь дистрибутив.

Подробнее об backports: https://wiki.debian.org/ru/Backports

7. Изоляция программ в контейнерах

- Современные Debian-based дистрибутивы поддерживают использование контейнеров, таких как **Docker**, которые изолируют приложения в среде с собственными зависимостями и конфигурациями, что улучшает безопасность за счет минимизации воздействия потенциальных уязвимостей.

  Подробнее о Docker: https://docs.docker.com/

- **LXC (Linux Containers)** — это другой вариант контейнеризации, поддерживаемый в Debian, который может использоваться для изоляции и повышения безопасности приложений.

  Подробнее о LXC: https://ubuntu.com/server/docs/lxc-containers

8. Проверка подписи пакетов

- Все пакеты в репозиториях Debian и его производных подписаны криптографическими ключами. Это предотвращает установку неавторизованных или поврежденных пакетов, поскольку система проверяет цифровые подписи перед их установкой.
- Пользователи также могут вручную добавлять репозитории и ключи подписи через APT, но нужно быть осторожным с неофициальными источниками.

9. Fail2Ban

- Для защиты от брутфорс-атак на сервисы, такие как SSH, можно использовать **Fail2Ban**. Этот инструмент отслеживает логи и автоматически блокирует IP-адреса после нескольких неудачных попыток входа.

  Подробнее об Fail2Ban: https://help.ubuntu.com/community/Fail2ban

Установка и настройка Fail2Ban:

bash
sudo apt install fail2ban
sudo systemctl enable fail2ban
sudo systemctl start fail2ban

10. Контроль учетных записей и прав

- В Debian-based системах строгое разграничение прав доступа пользователей и групп реализуется с помощью POSIX-разрешений и системы контроля привилегий (например, с помощью sudo).
- Также можно использовать механизмы вроде **PAM (Pluggable Authentication Modules)** для управления политиками аутентификации и доступа.

  Подробнее об Posix: https://ru.wikipedia.org/wiki/POSIX

  Подробнее об PAM: https://wiki.archlinux.org/title/PAM

11. Безопасная настройка SSH

- Для повышения безопасности удаленного доступа через SSH рекомендуется использовать ключи SSH вместо паролей, отключить root-доступ, а также использовать двухфакторную аутентификацию (например, через Google Authenticator).

Примеры настройки SSH:

bash
sudo nano /etc/ssh/sshd_config
# Отключить вход по паролю
PasswordAuthentication no
Отключить доступ root
PermitRootLogin no

12. Автоматическое применение обновлений

- В Debian-based системах можно настроить автоматическое применение обновлений безопасности с помощью утилиты **unattended-upgrades**, что особенно полезно для серверов.

Установка и настройка:

bash
sudo apt install unattended-upgrades
sudo dpkg-reconfigure unattended-upgrades

Подробнее об unattended-upgrades:

https://wiki.debian.org/UnattendedUpgrades

Безопасность Red hat-based

Базовые механизмы совпадают с Debian-based, рассмотрим те механизмы, которые отличаются от ранее рассмотренных:

1. Поддержка корпоративных стандартов и сертификаций

- **RHEL** сертифицирован в соответствии с различными международными стандартами безопасности, включая **Common Criteria** и **FIPS (Federal Information Processing Standards)**. Это делает его подходящим для использования в правительственных, финансовых и других отраслях, где требуются высокие стандарты безопасности.

Подробнее об Common Criteria: https://access.redhat.com/articles/1403233

Подробнее об FIPS:

https://en.wikipedia.org/wiki/Federal_Information_Processing_Standards

- Для использования в средах с повышенными требованиями к криптографии RHEL может быть настроен в режиме FIPS, обеспечивающем соблюдение стандартов криптографии.

2. Аутентификация и управление доступом

- Red Hat поддерживает различные методы аутентификации, включая **Kerberos**, **LDAP**, и **SSSD (System Security Services Daemon)**, что делает возможным интеграцию с централизованными системами управления доступом.

Подробнее об Kerberos : https://www.kerberos.org/software/tutorial.html

Подробнее об LDAP: https://ldapwiki.com/wiki/Wiki.jsp?page=LDAP

Подробнее об SSSD: https://sssd.io/docs/introduction.html

- Также используется **PAM (Pluggable Authentication Modules)** для гибкой настройки аутентификации и контроля доступа. Это позволяет настроить такие функции, как двухфакторная аутентификация (например, с использованием **Google Authenticator**).
-

3. Контейнерная безопасность

- В Red Hat контейнеры являются важным компонентом инфраструктуры, и для их безопасности применяются строгие механизмы контроля доступа. Red Hat предлагает **Podman** как замену Docker, который использует те же образы контейнеров, но обеспечивает лучшую изоляцию, поскольку Podman работает без необходимости запуска демон-процесса.

Подробнее о podman: https://podman.io/

- Контейнеры также работают под управлением **SELinux**, что обеспечивает дополнительный уровень защиты и предотвращает выход контейнеров за пределы своих прав.

4. SCAP (Security Content Automation Protocol)

- Red Hat предлагает **OpenSCAP**, набор инструментов для аудита безопасности и соответствия стандартам. Он используется для проверки систем на соответствие различным стандартам безопасности, таким как PCI-DSS или HIPAA, и предоставляет рекомендации по улучшению безопасности системы.
- OpenSCAP может автоматически проверять конфигурацию системы и обеспечивать отчетность по соблюдению политики безопасности.

Подробнее об OpenSCAP: https://www.open-scap.org/

5. Интеграция с Red Hat Satellite

- **Red Hat Satellite** — это инструмент управления жизненным циклом систем, который позволяет централизованно управлять обновлениями, патчами и конфигурацией безопасности большого количества систем. Это критически важно для крупных корпоративных сред с тысячами узлов.

Gentoo-based

Gentoo Linux — это высококонфигурируемый дистрибутив, известный своей гибкостью и мощной системой управления пакетами. Он предоставляет пользователям полный контроль над системой, что дает возможность настроить её так, чтобы обеспечить максимальную безопасность. Как и в случае с Arch Linux, безопасность Gentoo зависит от того, как пользователи настраивают и поддерживают свою систему. Рассмотрим основные аспекты безопасности в Gentoo Linux:

1. Гибкость и настройка безопасности на уровне компиляции

- Основное отличие Gentoo от других дистрибутивов заключается в том, что пакеты компилируются из исходных кодов с помощью **Portage**. Пользователь может использовать флаги **USE** для включения или отключения различных возможностей, что позволяет избежать установки ненужных или потенциально уязвимых функций.
- Также можно настроить компилятор с помощью **CFLAGS** и **LDFLAGS**, чтобы добавить дополнительные параметры безопасности, такие как защита от переполнения стека, защита памяти и другие методы харденинга.

  Пример настройки флагов безопасности:

  bash
  Копировать код
  CFLAGS="-O2 -pipe -fstack-protector-strong"
  LDFLAGS="-Wl,-z,relro,-z,now"

2. Hardened Gentoo

- **Hardened Gentoo** — это специальная сборка Gentoo, ориентированная на усиленные меры безопасности. Она включает в себя множество патчей и настроек для повышения безопасности, включая **PaX**, **grsecurity** (до прекращения его публичной доступности), и поддержку **PIE (Position Independent Executables)**, что делает её отличным выбором для систем, где безопасность имеет первостепенное значение.
- Включение профиля **hardened** автоматически настраивает системы для усиленной безопасности, добавляя поддержку защиты памяти и другие важные меры.

Как установить Hardened профиль:

bash
Копировать код
eselect profile list
eselect profile set <номер hardened профиля>

3. Пакеты с проверкой целостности

- В Gentoo все пакеты проверяются на целостность с использованием **GPG-подписей**, что защищает пользователей от установки поддельных или скомпрометированных пакетов. Система **Portage** использует сигнатуры для подтверждения того, что пакеты не были изменены.
- Пользователь может настроить дополнительную проверку целостности пакетов и использовать **eix** для проверки безопасности установленных пакетов.

4. Обновления и патчи безопасности

- Gentoo не имеет фиксированного цикла выпуска версий и использует модель rolling release. Это означает, что обновления безопасности могут быть получены сразу после их выпуска.
- Пользователи могут настроить систему так, чтобы она автоматически проверяла наличие обновлений и устанавливала только критические патчи безопасности.

Команды для обновления системы:

bash
Копировать код
emerge --sync
emerge -avuDN @world