# Использование Pluggable Authentication Modules (PAM) для обеспечения безопасности в НАЙС ОС

## Введение

Pluggable Authentication Modules (PAM) - это гибкий механизм аутентификации, который позволяет системным администраторам легко изменять правила аутентификации для различных служб. В данной документации рассмотрены установка, настройка и использование PAM в контексте обеспечения безопасности в НАЙС ОС, а также примеры использования команд.

## Установка PAM

Для установки PAM в НАЙС ОС используется пакетный менеджер `tdnf` или `dnf`. Воспользуйтесь следующими командами для установки необходимых пакетов:

```
$ sudo tdnf install pam
```

## Основные файлы конфигурации PAM

Конфигурационные файлы PAM расположены в каталоге `/etc/pam.d/`. Каждый файл в этом каталоге соответствует конкретной службе или программе. Главный конфигурационный файл PAM - `/etc/pam.conf`, но обычно используются файлы в `/etc/pam.d/` для каждой службы отдельно.

## Структура конфигурационных файлов PAM

Каждая строка в конфигурационном файле PAM состоит из четырех полей:

- **Модульная группа** - указывает, к какому этапу аутентификации относится правило (auth, account, password, session).
- **Контрольный флаг** - указывает, как обрабатывать результат модуля (required, requisite, sufficient, optional).
- **Модуль** - путь к модулю PAM, который будет выполняться.
- **Аргументы** - дополнительные параметры для модуля.

Пример конфигурационного файла для службы `sshd`:

```
#%PAM-1.0
auth       required     pam_sepermit.so
auth       include      password-auth
account    required     pam_nologin.so
account    include      password-auth
password   include      password-auth
session    optional     pam_keyinit.so force revoke
session    include      password-auth
```

© 2026 NiceSOFT — z.niceos.ru

## Основные модули PAM

Рассмотрим некоторые часто используемые модули PAM:

- **pam_unix.so** - стандартный модуль для аутентификации через /etc/passwd и /etc/shadow.
- **pam_deny.so** - модуль, который всегда завершает аутентификацию неудачей.
- **pam_permit.so** - модуль, который всегда завершает аутентификацию успехом.
- **pam_env.so** - модуль для установки переменных окружения.
- **pam_limits.so** - модуль для управления ограничениями ресурсов.
- **pam_tally2.so** - модуль для учета неудачных попыток входа.

## Примеры настройки PAM

Рассмотрим несколько примеров настройки PAM для различных сценариев:

### Блокировка учетной записи после нескольких неудачных попыток входа

Для блокировки учетной записи после нескольких неудачных попыток входа можно использовать модуль `pam_tally2.so`. Добавьте следующие строки в файл `/etc/pam.d/common-auth`:

```
auth required pam_tally2.so onerr=fail deny=5 unlock_time=900
```

Эти настройки блокируют учетную запись после 5 неудачных попыток входа и разблокируют ее через 15 минут (900 секунд).

### Принудительное использование сложных паролей

Для принудительного использования сложных паролей можно использовать модуль `pam_cracklib.so`. Добавьте следующие строки в файл `/etc/pam.d/common-password`:

```
password requisite pam_cracklib.so retry=3 minlen=12 difok=3
```

Эти настройки требуют, чтобы пароль был длиной не менее 12 символов и содержал как минимум 3 различных класса символов.

### Ограничение ресурсов для пользователей

Для ограничения ресурсов, доступных пользователям, можно использовать модуль `pam_limits.so`. Добавьте следующие строки в файл `/etc/pam.d/common-session`:

```
session required pam_limits.so
```

Настройте ограничения в файле `/etc/security/limits.conf`:

```
* hard nproc 100
* hard nofile 1024
```

Эти настройки ограничивают максимальное количество процессов (nproc) до 100 и максимальное количество открытых файлов (nofile) до 1024 для всех пользователей.

## Рекомендации по безопасности

- Используйте модули PAM для повышения безопасности аутентификации и управления доступом.
- Регулярно проверяйте и обновляйте конфигурацию PAM для соответствия текущим требованиям безопасности.
- Используйте сложные пароли и требуйте их регулярной смены.
- Ограничивайте доступ к критически важным системам и ресурсам.
- Логируйте и анализируйте все неудачные попытки аутентификации для выявления возможных атак.

## Заключение

PAM предоставляет мощный и гибкий механизм аутентификации и управления доступом в НАЙС ОС. Следуя приведенным рекомендациям и примерам, вы сможете эффективно контролировать и защищать свою систему, обеспечивая высокий уровень безопасности.

© 2026 NiceSOFT — z.niceos.ru