Управление сертификатами и криптографией: OpenSSL и Let's Encrypt

Управление сертификатами и криптографией является важной задачей для обеспечения безопасности в современной сети. В этом руководстве мы рассмотрим использование OpenSSL и Let's Encrypt для управления сертификатами и выполнения криптографических операций в операционной системе НАЙС ОС.

Основы OpenSSL

OpenSSL — это мощный и широко используемый инструмент для выполнения различных криптографических операций. Он поддерживает создание и управление сертификатами, генерацию ключей, шифрование и дешифрование данных.

Установка OpenSSL

В большинстве современных дистрибутивов Linux OpenSSL предустановлен. Чтобы проверить, установлен ли OpenSSL, выполните команду:

```
openssl version
```

Если OpenSSL не установлен, вы можете установить его с помощью пакетного менеджера:

```
sudo tdnf install openssl
```

Создание и управление сертификатами с помощью OpenSSL

OpenSSL предоставляет средства для создания сертификатов, управления ключами и выполнения других криптографических операций.

Генерация приватного ключа

```
openssl genpkey -algorithm RSA -out private.key -aes256
```

Эта команда генерирует приватный ключ RSA с использованием алгоритма шифрования AES-256 и сохраняет его в файл `private.key`.

Создание запроса на подписку сертификата (CSR)

© 2026 NiceSOFT — z.niceos.ru

```
openssl req -new -key private.key -out request.csr
```

Эта команда создает запрос на подписку сертификата (CSR) с использованием ранее созданного приватного ключа `private.key` и сохраняет его в файл `request.csr`.

Самоподписанный сертификат

```
openssl req -x509 -nodes -days 365 -key private.key -in request.csr -out certificate.crt
```

Эта команда создает самоподписанный сертификат, действительный в течение 365 дней, и сохраняет его в файл `certificate.crt`.

Просмотр информации о сертификате

```
openssl x509 -in certificate.crt -text -noout
```

Эта команда отображает детальную информацию о сертификате, содержащемся в файле `certificate.crt`.

Конвертация форматов сертификатов

OpenSSL поддерживает конвертацию сертификатов между различными форматами, такими как PEM, DER и PKCS#12.

Конвертация PEM в DER

```
openssl x509 -in certificate.crt -outform der -out certificate.der
```

Конвертация PEM в PKCS#12

```
openssl pkcs12 -export -out certificate.p12 -inkey private.key -in certificate.crt
```

Использование Let's Encrypt для получения SSL/TLS сертификатов

Let's Encrypt — это бесплатный и автоматизированный сервис для получения SSL/TLS

© 2026 NiceSOFT — z.niceos.ru

сертификатов. Он позволяет легко и быстро получать сертификаты, обеспечивая безопасность веб-сайтов.

Установка Certbot

Certbot — это клиентское ПО для взаимодействия с Let's Encrypt. Чтобы установить Certbot, выполните команду:

```
sudo tdnf install certbot
```

Получение сертификата с помощью Certbot

Чтобы получить сертификат для вашего домена, выполните следующую команду:

```
sudo certbot certonly --standalone -d example.com
```

Эта команда запускает Certbot в режиме standalone, который автоматически запрашивает сертификат для домена `example.com`.

Автоматическое обновление сертификатов

Let's Encrypt сертификаты действительны в течение 90 дней. Certbot может автоматически обновлять сертификаты. Настройте cron для автоматического обновления:

```
sudo crontab -e
```

Добавьте следующую строку для ежедневной проверки и обновления сертификатов:

```
0 0 * * * certbot renew --quiet
```

Использование сертификатов в веб-сервере

После получения сертификатов, их необходимо настроить в веб-сервере, таком как Apache или Nginx.

Настройка SSL в Apache

```
sudo nano /etc/httpd/conf.d/ssl.conf
```

© 2026 NiceSOFT — z.niceos.ru

Добавьте или измените следующие строки:

```
SSLCertificateFile /etc/letsencrypt/live/example.com/fullchain.pem
SSLCertificateKeyFile /etc/letsencrypt/live/example.com/privkey.pem
```

Перезапустите Apache для применения изменений:

```
sudo systemctl restart httpd
```

Настройка SSL в Nginx

```
sudo nano /etc/nginx/conf.d/example.com.conf
```

Добавьте или измените следующие строки:

```
server {
    listen 443 ssl;
    server_name example.com;

    ssl_certificate /etc/letsencrypt/live/example.com/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/example.com/privkey.pem;

    ...
}
```

Перезапустите Nginx для применения изменений:

```
sudo systemctl restart nginx
```

Дополнительные возможности OpenSSL

OpenSSL предоставляет множество дополнительных возможностей, таких как шифрование и дешифрование данных, генерация случайных чисел и выполнение различных криптографических операций.

Шифрование и дешифрование файлов

```
openssl enc -aes-256-cbc -salt -in file.txt -out file.txt.enc
```

Эта команда шифрует файл `file.txt` с использованием алгоритма AES-256-CBC и сохраняет

зашифрованный файл как `file.txt.enc`.

Дешифрование файла

```
openssl enc -d -aes-256-cbc -in file.txt.enc -out file.txt
```

Эта команда дешифрует файл `file.txt.enc` и сохраняет результат как `file.txt`.

Генерация случайных чисел

```
openssl rand -hex 16
```

Эта команда генерирует случайное число длиной 16 байт в шестнадцатеричном формате.

Управление сертификатами и криптографией с помощью OpenSSL и Let's Encrypt в НАЙС ОС предоставляет мощные и гибкие средства для обеспечения безопасности. Использование OpenSSL для создания и управления сертификатами, а также Let's Encrypt для автоматического получения SSL/TLS сертификатов помогает поддерживать высокий уровень безопасности в сети. Следование описанным шагам и примерам позволит вам эффективно управлять сертификатами и выполнять криптографические операции в вашей системе.

© 2026 NiceSOFT — z.niceos.ru