OpenSSL и ГОСТ. Настройка

Опубликовано **13 Август, 2015**

Все государственные структуры обязаны использовать в своих информационных системах только **ГОСТ** криптографические алгоритм. Поэтому, всем IT специалистам приходятся сталкиваться с криптографией по **ГОСТ** при разработке или сопровождению информационных систем, если в требованиях есть условие по взаимодействию с информационными системами государственных органов.

Одним из самым популярных кроссплатформенных инструментов для работы с криптографией является проект **OpenSSL**. Поддержка **ГОСТ** алгоритмов добавлена в **OpenSSL** версии *1.0.0*. Начиная с этой версии, после правильной настройки, мы получим полноценную работу с **ГОСТ** алгоритмами в наших проектах, если они используют **OpenSSl**.

Для использования **ГОСТ** алгоритмов в **OpenSSL** необходимо установить последнюю версию:

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install openssl
```

После этого нужно настроить **OpenSSL**. Для этого необходимо в начало конфигурационного файла **OpenSSL**, который расположен по адресу `/etc/ssl/openssl.cnf` после

```
#
# OpenSSL example configuration file.
# This is mostly being used for generation of certificate requests.
#
```

добавить

```
openssl_conf=openssl_def
```

и в конец файла добавить:

```
# OpenSSL default section
[openssl_def]
engines = engine_section

# Engine scetion
[engine_section]
gost = gost_section

# Engine gost section
[gost_section]
engine_id = gost
dynamic_path = /usr/lib/ssl/engines/libgost.so
default_algorithms = ALL
CRYPT_PARAMS = id-Gost28147-89-CryptoPro-A-ParamSet
```

- Параметр **engine_id** указывает на название движка. Устанавливаем значение **gost**
- Параметр **dynamic_path** указывает на путь до динамической библиотеки **libgost**
- Параметр **default_algorithms** указывает на использование движком всех алгоримов, которые есть в движке
- Параметр **CRYPT_PARAMS** опция только для библиотеки **libgost**. Позволяет пользователю выбирать наборы параметров симметричного алгоритма шифрования. Без этой опции не будет работать опция **-gost89**, что в свою очередь ведёт к тому, что при шифровании данных вместо **GOST 28147-89** используется **rc2-cbc**.

Сохраняем файл.

Далее находим расположение библиотеки libgost.so

```
sudo find / -name "libgost.so"
```

Если библиотека не расположена по путь, указанному в файле конфигурации с предыдущего шага (**/usr/lib/ssl/engines/libgost.so**), то создаём симлинк

```
sudo mkdir -p /usr/lib/ssl/engines
sudo ln -s /usr/lib/x86_64-linux-gnu/openssl-1.0.0/engines/libgost.so
```

- Внимательно проверяем **путь** и **архитектуру** библиотеки того места, **откуда** мы ставим симлинк!

Проверка правильности настройки можно командой

```
openssl ciphers|tr ':' '\n'|grep GOST
```

в ответ должны получить что-то в виде:

GOST2001-GOST89-GOST89
GOST94-GOST89-GOST89

На этом всё, поддержка ГОСТ алгоритмов в OpenSSL включена! С чем, собственно, можно и поздравить!

Источник — документация проекта OpenSSL

Запись опубликована автором **kirill.zak** в рубрике **Linux** с метками **Linux Mint**, **OpenSSL**, **Ubuntu**, **ГОСТ**, **криптография**. Добавьте в закладки **постоянную ссылку [https://kirill-zak.ru/2015/08/13/298]** .

OPENSSL И ГОСТ. НАСТРОЙКА: 3 КОММЕНТАРИЯ

Уведомление: OpenSSL и ГОСТ. Генерация ключей | Записки от Kirill.zak

Уведомление: Сервер проверки подписи OpenSSL+ GOST — activities.plekhov.ru

Евгений
говорит **28 Ноябрь, 2017 в 16:22**:

Отличная статья! Спасибо, помогла!