УДК 004.056.55

О.О. Бобришов[1]
Кіровоградський національний технічний університет

Дослідження інтерфейсу програмування додатків CryptoAPI

Вступ. На даний час багато додатків використовують для обміну даними відкриті канали зв'язку, і, насамперед, Internet. Проте у ряді випадків, наприклад у банківській та фінансовій сферах, потрібно обмежити доступ до конфіденційної інформації, переданої по таким каналам. При цьому важливо мати можливість перевірити, від кого виходять прийняті одержувачем конфіденційні дані і чи не були вони спотворені при пересиланні.

Основна частина. Для безпечної передачі конфіденційних даних по відкритим каналам використовується криптографія. У сфері захисту комп'ютерної інформації криптографія застосовується в основному для: шифрування і дешифрування даних; а також створення та перевірки цифрових підписів.

Шифрування даних дозволяє обмежити доступ до конфіденційної інформації, зробити її незрозумілою для сторонніх. Застосування цифрових підписів залишає дані відкритими, але дає можливість верифікувати відправника і перевіряти цілісність отриманих даних. Для захисту інформації фахівцями Microsoft був розроблений інтерфейс CryptoAPI, який дозволяє створювати додатки, що використовують криптографічні методи.

В CryptoAPI існує поняття «криптопровайдер» (Cryptography Service Provider, CSP). Криптопровайдер - це незалежний модуль, що містить бібліотеку криптографічних функцій із стандартизованим інтерфейсом. Криптопровайдер відповідає за реалізацію функцій інтерфейсу, а також грає роль сховища для ключів. CryptoAPI підтримує роботу з асиметричними і симетричними ключами, тобто дозволяє шифрувати і розшифровувати дані, а також працювати з електронними сертифікатами. Набір підтримуваних криптографічних алгоритмів залежить від конкретного криптопровайдера. Криптопровайдер відповідає за зберігання і руйнування ключів.

Висновки. Виходячи з усього вище описаного використання CryptoAPI в операційних системах, таких як Windows, linux, freebsd надає можливість прикладному рівню доступ до криптографічних функцій для генерації ключів, формування або перевірки електронного цифрового підпису, шифрування/дешифрування даних, що в свою чергу не потребує від програміста вивчення особливостей реалізації того або іншого алгоритму.

[1] асистент кафедри програмного забезпечення