

László Tímea, V. évf. programtervező matematikus

Szegedi Tudományegyetem Informatikai Tanszékcsoport
Szoftverfejlesztés Tanszék

Konzulens: Kiss Ákos

Egyetemi Tanársegéd

C++ programok vezérlési folyamának átalakítása a kód védelme céljából

A védelem sosem egyszerű feladat. Ahogy azonban az élet más területein is, az informatikában is egyre nagyobb szükség van rá. Tökéletes biztonságot garantálni persze lehetetlen feladat, hiszen amit emberi kéz hoz létre, azt emberi kéz le is rombolhat. Megfelelő idő- és energiaráfordítással egy hozzáértő számára nem létezik megfejthetetlen kód. A cél tehát az, hogy a lehető legjobban megnehezítsük az illetéktelen hozzáférés és felhasználás lehetőségét. A rendszerek biztosságossá tétele többféleképp történhet, a hardverek szintjétől, az operációs rendszerek szintjén keresztül a forráskód szintjéig. Ebben a munkában az obfuscálással, azaz a forráskód szintű védelemmel foglalkozunk. A cél az, hogy a program megértésével járó fáradság a lehető legnagyobb legyen, és így csak kevésbé vagy akár egyáltalán ne érje meg a kódot visszafejteni. Ennek érdekében a forráskódon egy olyan egyirányú átalakítást végzünk, ami a megértést megnehezíti, ám a program működését változatlanul hagyja.

Az elérhető nyílt forráskódú, illetve kereskedelmi célú obfuscáló programok közül a leggyakoribbak a Java rendszerek védelmére írottak. Léteznek ugyan C++ programozási nyelvre specializált megoldások is, ám azok csak egyszerű, szerkezeti átalakításokon alapuló algoritmusokat alkalmaznak. Nem szabad elfelejteni azonban azt a tényt, hogy még ma is előszeretettel írnak nagyobb rendszereket C++ nyelven, így ezek védelme nem elhanyagolható fontosságú. Ebben a dolgozatban a vezérlési folyamat átalakító módszerek egyikének, a vezérlési folyam kilapításának C++ nyelvre való alkalmazásával foglalkozunk részletesen. A módszer alapötlete ugyan már korábban informálisan leírásra került, de nem jelent még meg olyan publikáció, amely a módszer valamely programozási nyelvre való adaptálását taglalná. Dolgozatunkban ismertetjük az C++ nyelvre való alkalmazás során felmerült problémákat és megoldásokat is adunk rájuk, valamint megadjuk az algoritmus teljes formális leírását. Továbbá annak ellenére, hogy a dolgozatban közölt algoritmust kifejezetten C++ programokra specializáltuk, a felmerülő problémákra adott megoldások általánosak, azaz a hasonló programozási nyelvekben a megfelelő nyelvi szerkezetek adaptálásával alkalmazhatók.