

Szoftverjogi csiki-csuki, avagy a szoftver analízis és annak határai

A szoftverek visszafejthetőek. Ezzel vélhetően nem árultam el nagy újdonságot. Azzal talán annál inkább, hogy a szoftverek visszafejtése bizonyos körülmények között jogszerű is lehet.¹

A szoftver visszafejtése

A szerző engedélye nélkül is jogosult a program használatára licenccben vagy más módon jogosított felhasználó, vagy megbízottja (hiszen egy mezei felhasználó az esetek 99,99%-ában erre úgysem lenne képes) vagy egy programozó akár munkaköri kötelességként is a kód többszörözésére, szükség esetén (vissza)fordítására egy speciális információhoz való hozzájutás érdekében. Ezen információ kizárólagosan azon célt szolgálhatja, hogy a visszafejtett szoftvert más szoftverekkel együtt tudja működtetni a felhasználó. A központi fogalom tehát az „interoperabilitás” megteremtése.

A reverse engineering

Az alábbi bekezdést legjobb lesz, ha a profik átugorják, ez tartalmazza ugyanis annak összefoglalóját, hogy mit ért egy jogász a *reverse engineering* fogalma alatt.² Azon eljárást, melynek keretében a tárgykódból különféle, a továbblépéshez szükséges információkat nyernek, reverse engineeringnek nevezzük. Ennek egyik eszköze a dekompiláció, ami a tárgykódnak egy fordítóprogram segítségével forráskódba fordítását jelenti. Ám – lévén, hogy rengeteg fordító (kompiláló) program létezik, – amennyiben a visszafordításhoz nem jó dekompiláló programot (azaz nem az eredeti kódolásnak megfelelőt) használunk, a kívánt információk helyett egy értelmetlen és értelmezhetetlen karaktorsorhoz jutunk. Ezért általában, (ha a fordító program „ujjlenyomata” (fingerprint) alapján nem képes a fordítást végző azonosítani a programot) akkor egy alap programozási nyelvre, assembly-be fordítja. Ezen lehetőség bármiféle tárgykód esetében fennáll, ám azzal a kockázattal, hogy a szerző kommentár sorait nem nyerjük vissza, és az assembly csak szakember számára olvasható programnyelv. A kompilált sorokat aztán az esetleges változtatásokat követően természetesen lehet recompileálni, amely eljárást követően újra tárgykódban lévő programhoz jutunk.

A de-, recompileáció lényegi elemei tehát egyrészt a fordítás, másrészt pedig egy speciális többszörözés. Ezen eljáráshoz való hozzájárulás tehát a szerzőknek általában nem is áll érdekükben. Ezért csak különböző előfeltételek együttállása esetében lehet egyáltalán szó ezen eljárások jogszerű alkalmazásáról. Lássuk most ezeket.

Az információhoz való alternatív hozzájutás hiánya

Amennyiben a megkívánt információkhoz szabad hozzájutást biztosít a szerző, a programot nem kell felfejteni. Erre példa a GPL licenclésű a szabad szoftverek esete, hiszen itt kötelező jelleggel megadják a kódot magát, vagy azt elérhetővé teszik az Interneten. Felmerülhet azonban kérdésként, hogy milyen tágran értelmezhető „a megkívánt információt „könnyedén megismerhetősége”, mint kitétel. Ide sorolja a jogirodalom a régebben nyilvánosságra hozott, ám kis kutatással fellelhető információkat. Ellenpéldaként viszont ide tarthat, ha az az információt további ellenszolgáltatásért cserébe kínálják fel. Nem szabad elfelejteni, hogy a könnyen hozzáférhetőség mindig egyedi mérlegelést megkívánó fogalom. Ugyancsak kérdés, hogy kinek is áll érdekében az információhoz jutás biztosítása, elvárható-e a felhasználótól, hogy a fejlesztő felkeresésével időt töltsön, annak érdekében, hogy a dekompiláció ne minősüljön joggal való visszaélésnek? Valószínűleg, rövid határidő tűzés mellett a fejlesztő felkeresése és felszólítása az adat szolgáltatására jogos elvárás.

Az információk célhoz kötött volta

A másik alapvető feltétel, hogy a megszerzett információk ténylegesen a programok együttműködésének lehetőségét biztosítsák. Minden más céllal történő visszafejtés engedélyköteles. Ugyanúgy nem szabad ötletek megszerzése céljából dekompileálni, mint ahogy jogsér-

¹ Sztj. 60. §.

² Kritikai észrevételeket ide várok: agi@abend.hu

tés bizonyítása sem megengedett³ ezen a módon. (Amennyiben bíróság rendeli el a szakértő általi dekompilációt, az persze más eset, ám a bíróság jogosult a sokkal egyszerűbb utat is választani, kérni a forráskód kiadását.) Sőt a felhasználó azt sem teheti meg, hogy a szoftver hibáját ezúton bizonyítja, hiszen – bár a cél nemes – mégsem felel meg az irányelv és a törvény szövegének, mely egyértelműen csak „önállóan megalkotott szoftver más szoftverekkel való együttes működtetéséhez szükséges információ megszerzés érdekében” teszi lehetővé az eljárást. A visszafejtés nem használható sem a felhasználó érdekeit közvetlenül szolgáló testre szabás céljából, sőt még kutatási célokból sem.

Érdekes azonban, hogy a kompatibilissé tétel abban az esetben is engedélyezett, ha egy konkurens programmal akarják az adott szoftvert együttműködésre bírni.

Ugyancsak speciális eset a hardverrel való együttműködés, együttműködtetés, hiszen a szöveg szó szerint csak szoftverekhez való illesztést engedi. A mai modern hardver elemek lehetővé teszik a szoftverek hardveres implementálását, azaz szoftver feladatok hardverben történő ellátását. Ez pedig indokoltá tenné a dekompilálás engedélyezését erre az esetre is. Hiszen ennek elmaradása a piac megszűntetéséhez vezethet, ami versenyjogi aggályokat vet fel.

A szoftvergyártók általában ellátják terméküket egy interface specifikációval, mely esetenként azonban nem teljes körű, így a visszafejtés elkerülhetetlen. Ám az abból nyert információk (akár az interface alkotó kódja is) a visszafejtő rendelkezésére állnak. Ezek közül azonban a szabályozás alapelveinek megfelelően elsődlegesen a specifikációt kell kiegészíteni és azt használni, nem pedig a konkrét kódot áttemelni.

A személyi kör

A dekompilálás három személyt érinthet. A felhasználási szerződésben jogosítottat, vagy a szoftver felhasználására jogosult más személyt (pl.: munkaviszony alapján eljárást) illetve ezek megbízottját, aki semmiféle közvetlen viszonyban nem áll a szerzővel. Ezen személy eljárási jogosultságát elég egy polgári jogi jogviszonnyal igazolni. A személyi kör utolsó elemmel való kibővítését az tette szükségessé, hogy a dekompilálás – mint már a bevezetésben említésre került – speciális tudást igényel.

Az információ terjedelme

A szükséges programrészek jelentik az információ megismerésének határát. Feltétlen megismerést igénylő, tehát az együttműködéshez szükséges információt tartalmaznak például a (hivatalos és nem hivatalos⁴) csatlakozó felületek (interface).

A dekompilációt végző személytől elvárható, hogy a kézikönyvek és a rendelkezésre álló irodalom információi alapján kiderítse, hogy melyik programrészt kell dekompilálni. Egyes – az újabb programokra egyre jellemzőbb – megoldások esetében, melyek könyvtárak bevonásával

dolgoznak ez a hely könnyebben meghatározható, azaz nem kell valamennyi könyvtárat visszafordítani. Más esetekben viszont, mikor szorosan összefüggő programszövegről van szó, felmerülhet a teljes fordítás elkerülhetetlensége. A vizsgálat terjedelmi korlátait mindig speciális vizsgálat keretében kell ellenőrizni, vizsgálni kell, hogy adott szakértelem mellett előre látta-e vagy láthatta-e, hogy a program mekkora részének visszafordítása szükséges. Azonban még annak sincs kialakult gyakorlata, hogy a terjedelmi korlátok „túl nem lépését” ki köteles bizonyítani.

A megismert információ szabadsága

A szerző érdekeinek védelme szükségesé teszi, hogy a visszafordítás keretében elnyert információ ne legyen továbbadható. E védelmi szint még abban az esetben is fenntartandó, ha más felhasználók ugyancsak visszafordításokra lesznek kénytelenek, melyek végeredményeként természetesen ugyanazon eredményeket kapják. Az említett korlátozás olyannyira kereteket jelent, hogy a megtalált interfész nem adható közre a szakirodalomban, viszont az érem másik oldalaként az általunk írt szoftver kézikönyvében megjelenhet, hogy hogyan tehető kompatibilissé adott esetben egy operációs rendszerrel a programunk. Egyedi megoldás, hogy ha a szerzői oldalról visszaélést tapasztalnak, vagy piacot uraló helyzet áll fenn, a kartelljogi szabályokra hivatkozva követelhetjük a lényeges információk közreadását.

Az utánozási tilalom

Ismételt tilalomként jelenik meg, a nyomatékosság kedvéért, hogy a megszerzett ismeretek alapján nem lehet egy az eredetihez hasonló programot létrehozni, sem olyat, mely hasonló elvekre épülne, vagy hasonló funkciókkal lenne ellátva. Ez voltaképpen versenyjogi kérdésnek tekinthető. Kétségtelenül tisztességtelen piaci magatartást testesítene meg a fél, aki az említett módon hozzájutva az információkhoz kezdi meg konkurens szoftver terjesztését.

Ez azonban nem jelenti azt, hogy ne lehetne – az eredeti használata nélkül alkotni egy másikat, hasonló céllal, hiszen ezen tilalom léte teljesen monopolizálttá tenné a piacot... egyetlen operációs rendszer működne, amin egyetlen szótárprogramot és egyetlen könyvelő programot lehetne futtatni, de mivel az ötletek nem védettek, így lehetőséget kell biztosítani másnak is hogy egy adott probléma megoldására szoftvert fejlesszen.



Dr. Dudás Ágnes (dudas.agnes@abend.hu) ügyvédjelölt, az FSF egyik aktivistája. 2004-ben végzett az ELTE Jogtudományi Karán. Szakdolgozatát a szoftverek szerzői jogi védelméről írta, a 2003-as évet pedig e terület kutatásával a berlini Humboldt Egyetemen töltötte.

³ Dreier: GRUR 1993 781-785

⁴ Ezen megkülönböztetés versenyjogilag lehet releváns, hiszen a nem hivatalos interface-ek arra szolgálnak, hogy azokat csak maga a gyártó használhassa.