

TDK-dolgozat

Készítette:

Barta Ákos

Budapesti Corvinus Egyetem
Gazdálkodástudományi kar
Infokommunikációs Tanszék

Az online látogatók követésének módszerei

Anonimitás az interneten

Konzulens: Balogh Zoltán
(Tudományos segédmunkatárs)

Barta Ákos

Gazdálkodástudományi Kar

Alapképzés

Gazdaságinformatikus szak

IV. évfolyam

2012. március 26.

TARTALOMJEGYZÉK

<i>Nyitó gondolat.....</i>	<i>1</i>
<i>A téma áttekintése és fontossága.....</i>	<i>2</i>
<i>Technológiai háttér – az azonosítás lehetőségei</i>	<i>4</i>
A böngészőből elérhető adatok.....	4
Közösségi oldalak felhasználása.....	7
Widget-ek	8
<i>Védekezés alapvető módjai.....</i>	<i>9</i>
<i>A védekezés egy másik oldala.....</i>	<i>12</i>
A User Agent módosítása.....	12
Szolgáltatói adatok hamisítása	13
Proxy szerverek használata	13
IP cím cseréje: HideMyIp	14
Belépés egy titkosított hálózatba: A TOR Projekt	14
Környezeti adatok módosítása	16
<i>Azonosítás és követés a gyakorlatban</i>	<i>17</i>
YouHaveDownloaded.....	17
Minta projekt: BCE Anonymity crawler	20
Bemutató	20
Felépítés és megvalósítás	20
Felhasználás	23
Googly Analytics.....	23
<i>Összefoglalás</i>	<i>25</i>
<i>Forrásjegyzék.....</i>	<i>26</i>

NYITÓGONDOLAT

Dolgozatom témájának az internethasználók azonosítását, tevékenységük követését és az ezekben rejlő lehetőségek vizsgálatát, elemzését választottam. Mivel egyre többször és egyre intenzívebben kerülünk kapcsolatba a webbel, ezért véleményem szerint a téma mindinkább foglalkoztatja az embereket. Már felvetődik annak a kérdése is, hogy milyen platformon milyen adatokat és kikkel osztunk meg, valamint elkezdünk fokozottan ügyelni is ezek kényességére.

Rengeteg olyan – többnyire ingyenes – Web 2.0-ás szolgáltatást használunk nap, mint nap, melyek valamilyen szinten monitorozzák tevékenységünket és gyűjtenek rólunk adatot, ilyen például a Google Mail. A GMail kapcsán felmerül egy *új* trend, az internetes felhő (Cloud Computing) térnyerése is, melynek lesarkítva és a témámhoz kapcsolódóan az a lényege, hogy adatainkat már nem a saját számítógépeinken, hanem valahol a felhőben, interneten tároljuk, olvassuk és szerkesztjük. Jogosan fogalmazódik meg a felhasználókban, hogy ez információtechnológiailag mennyire biztonságos és, hogy visszaélnek-e a feltöltött tartalmakkal.

A kérdéskör, amire válaszolok a kutatásom során az, hogy az egyének internethasználat közben milyen szinten bújhatnak az internetes anonimitás mögé – tehát tényleg névtelenül böngészünk a világhálón vagy ez az alapkoncepció 2012-re már megváltozott? Kik és hogyan láthatják, hogy mit csinálunk egy-egy oldal olvasása közben; mit szűrhetnek le ezekből? Jelennek ezek valamiféle veszélyforrást?

A TÉMA ÁTTEKINTÉSE ÉS FONTOSSÁGA

A nyitó gondolatban már utaltam arra, hogy szerintem a téma rendkívül aktuális és ezt misem támasztja még jobban alá, minthogy hétről hétre felreppen egy-egy hír a közösségi oldalak vagy az e-kereskedelmi óriások és egyéb vállalatok adatkezelési politikája kapcsán.

2012. februárjában olvashattuk például az Index hírportálon (Index.hu, 2012), hogy egyes mobil alkalmazások hozzáférnek a készüléken tárolt személyes adatainkhoz, így többek közt az SMS-einkhez és névjegyzékünkhöz is. Ebben számomra az zavaró, hogy például a Facebook alkalmazásának mi szüksége van az SMS-eimre, holott nincs hozzá köze. A cikkből kiderül, hogy a közösségi oldal mobil verziója a hivatalos magyarázat szerint azért „olvassa” az üzeneteket, hogy tesztadatokat gyűjtsön egy éppen fejlesztés alatt álló szoftverükhöz. A szolgáltatók több ponton is el tudták háritani a támadásokat, hiszen van ahol ez is része az általános szerződési feltételeknek és van ahol pedig egyszerűen hozzáférést kér az alkalmazás a fentebb említett adatokhoz.

Alig egy évvel korábban nagy port kavart, hogy az Apple iPhone és iPad készülékei rögzítik, hogy a tulajdonosuk mikor merre jár. Alasdair Allan és Pete Warden jelentette be 2011. áprilisában, hogy ezek a készülékek folyamatosan rögzítik pozíciójukat. Ezt egy lokális, nem titkosított adatbázis-fájlban tárolják a GPS koordinátákkal, közeli WIFI hálózatok adataival, cellainformációkkal és időbélyegekkel együtt, ami azért is aggasztó, mert ez a fájl megjelent minden olyan számítógépen, amivel a mobileszközt szinkronizálták és bárki számára könnyedén olvasható volt, valamint a készülék elküldte ezeket az Apple felé is. A legnagyobb magyar Apple termékekkel foglalkozó oldalon, a szifon.com-on részletesebben is olvashatunk az ügygel kapcsolatban, ahol leírják, hogy hivatalosan ez a fájl is szükséges az akkoriban bevezetett többszálú programfuttatás helyes működése szempontjából (Jadeye, 2011). Ez tehát azt jelenti, hogy amikor a navigációs alkalmazásból út közben átváltunk egy másikra – például egy bejövő hívás esetén – akkor a készülék továbbra is mondja az útvonalat. Másik előnye ennek a lokális adatbázisnak, hogy gyorsabban és pontosabban reagálnak a térképes alkalmazások. A cikkben olvasható az Apple hivatalos állásfoglalása is, mely szerint az adatbázisfájl készítéséhez a készülék nem a saját, hanem a közeli adótornyok és WIFI hálózatok GPS koordinátáit rögzítette annak érdekében, hogy hamarabb meg tudja határozni a pozíciókat ha később szükséges lenne – más kérdés, hogy ez nem változtat a kiinduló problémafelvetésen.

A következő szoftverfrissítésben a cég befoltozta ezt a „hibát”: csökkentette a megőrzött rekordok számát és titkosította a fájl tartalmát.

A fenti két példával kapcsolatban elmondhatjuk, hogy valóban aggaszthat minket, hogy az SMS üzeneteinket vagy a napi útvonalunkat látja egy külső fél, de nem szabad figyelmen kívül hagynunk, hogy egy olyan céget, mint például az Apple valószínűleg nem érdekli, hogy mi hol fogyasztjuk a reggelinket. Ettől függetlenül mégis felhasználta ezeket az adatok valamiért. Vajon mi céljuk lehet ezeknek a furcsa adatgyűjtőgetéseknek? A kérdés megválaszolásához nézzük meg a Google Adatvédelmi Irányelveinek (Google Privacy Policy) kiemelt részletét:

„A Google által gyűjtött adatok

Az adatokat azért gyűjtjük, hogy jobb minőségű szolgáltatásokat nyújthassunk valamennyi felhasználónknak – kezdve az alapvető dolgokkal, például azzal, hogy milyen nyelven beszélhet, a bonyolultabb kérdésekig, például hogy Ön melyik hirdetéseket találhatja leginkább hasznosnak, vagy hogy kik azok, akik a legfontosabbak az Ön számára az online felületen.”

A kulcsgondolat az egész mögött szerintem tehát az, hogy ezek a szolgáltatók adatokat gyűjtenek azért, hogy javítsák a szolgáltatásaik minőségét és, hogy minél pontosabban kielégítsék a felhasználók igényeit. Mint tudjuk ingyen ebéd nincsen, tehát tulajdonképp ezekért a többnyire ingyenes alkalmazásokért mi, a felhasználók az adatainkkal fizetünk. Ha alaposabban megnézzük az idézett részt láthatjuk egy másik fél érdekeit is – a hirdetőkéét. Tulajdonképp a Google hirdető partnerei pontosan azért hajlandók nagyon sok pénzt fizetni, mert tudják, hogy ezen a csatornán keresztül sokkal célzottabban jut el a hirdetésük a felhasználókhöz. A dolgozatom során kifejtem részletesebben, hogy milyen technológiai lehetőségei vannak egy oldal üzemeltetőjének arra, hogy monitorozza a felhasználók viselkedését, mikre tudja a kapott adatokat felhasználni és, hogy a felhasználók miképp védekezhetnek ez ellen.

TECHNOLÓGIAI HÁTTÉR – AZ AZONOSÍTÁS LEHETŐSÉGEI

A BÖNGÉSZŐBŐL ELÉRHETŐ ADATOK

Az internetes közegre való tekintettel a legnyilvánvalóbb azonosítási lehetőség az oldalak üzemeltetőinek eszköztárában a böngészőprogram, illetve az azon keresztül elérhető adatok elemzése, aggregálása.

Mindenek előtt tisztáznunk kell, hogy mégis miket lehet megtudni az internethasználókról a böngészőjükön keresztül. Ezeket (a teljesség igénye nélkül) az alábbi kategóriákba sorolnám:

- A program adatai
- Szolgáltatói adatok
- A számítógép környezeti tulajdonságai
- HTML5 és egyéb új technológiák adta lehetőségek

Az első csoportba tartoznak magának a programnak a tulajdonságai, mint annak gyártója, neve és aktuális verziója. A felhasználók azonosításának szempontjából nyilvánvalóan jó kiindulópontnak tűnik ezen adatok felhasználása, azonban itt több akadályba is ütközünk. Például egy számítógépen több böngésző is lehet telepítve, egy gépet több ember is használhat, egy személy több platformon netezhet és lehet az is, hogy valaki nem mindig azt a programot használja, amelyiket leginkább szereti (hiszen egyes webes alkalmazások futtatása csak egy-egy konkrét böngészőn lehetséges). Az eddigiek alapján ez az információ alkalmas arra, hogy az egy számítógépet használók között különbséget tudjunk tenni, azonban nagy nehézséget okoznak a programfrissítések, amik egyben tönkre is tennék a látogatók követését, ha csak ezen támaszkodnánk.

A második csoportot alkotó szolgáltatói adatok alatt értem például az IP címet, amiből következik, hogy egyben tudjuk a hostnevet, szolgáltatót, valamint visszafejthető a pozíció is, de az utóbbira később jobb megoldásokat mutatok. Ezek az adatok már jók arra, hogy ne csak (lokálisan) egy konkrét számítógépet használó több ember között tudjunk különbséget tenni, hanem idézőjelesen (globálisan) a számítógépek között is – persze valójában itt hozzáférési pontokat értünk. Ez tovább mutat egy következő problémára, miszerint a belső hálózatok is bonyolítják a helyzetet, mert egy adott IP címmel több számítógép is csatlakozik a világhálóra. Ha csak statikus IP címek lennének, akkor egyértelműen egy biztos pontnak mondhatnánk ezt

az adatot, azonban ezeket többségében (főleg a magánszemélyek körében) dinamikusan osztják ki a szolgáltatók és így a személyi számítógépek és otthoni hálózatok körbeadhatják egymásnak a címeket, tehát ebben az esetben az IP cím csak rövid ideig alkalmas azonosításra.

A harmadik kategóriába a számítógép környezeti tulajdonságai közé sorolok olyan adatokat, mint a képernyő felbontása, a futtató operációs rendszer, a telepített betűtípusok, nyelvi, területi információk, süti beállítások, telepített beépülő modulok. Ezeket felhasználva már meglehetősen jól tudunk azonosítani egy konkrét számítógépet, hiszen ezek olyan tulajdonságok, melyek nagyon ritkán vagy szinte „sosem” változnak. Tehát ha kiválasztunk néhány szignifikáns elemet ezek közül és képezzük ezeknek egy lenyomatát, például ezek MD5 hash kódját, akkor szinte pontosan fel tudunk ismerni egy gépet.

A lenyomatgyártó (hasító) algoritmusukról tudni kell, hogy bemenetük minimális változására kimenetüket nagy mértékben megváltoztatják és elméletileg két különböző bemenő jelsorozatra nem adnak azonos kimeneti értéket, pontosabban ennek az egyezésnek a valószínűsége minimális. Azt kell tehát alaposan átgondolni, hogy mely adatokból képezzünk lenyomatot. A fent felsoroltak között szerepelnek olyanok amelyek relatíve kevés diszkrét értéket vehetnek fel (például operációs rendszer), de van olyan is, ami sokkal-sokkal többet, mint például a telepített betűtípusok. A fontokkal kapcsolatban az a probléma, hogy egy átlagos felhasználó nem feltétlen telepít újabb betűkészletet, így nagyon sok lehet az „alapbeállítás”, ami visszavezet az előző akadályhoz, hiszen lesz rengeteg bázisadat és lesznek ettől (jelentősen) eltérő értékek. Ebből a néhány példából látszik, hogy ha nagyon szeretnénk, akkor mindegyik tökéletlensége mellett tudunk érvelni, viszont minél több ilyen környezeti tulajdonságot adunk át a hash függvényünknek, annál valószínűbb, hogy nem lesz hibás egyezés.

A böngészők egyedi ujjlenyomatának megállapítására (tehát a megfelelő adatok megfelelő hash-algoritmussal való kombinálására) láthatunk gyakorlati példát egy internetes jogvédő szervezet, az Electronic Frontier Foundation (EFF) Panopticlick nevezetű kutatásában. A panopticlick.eff.org oldalon tesztelhetjük a böngészőnket, hogy mennyire egyedi, mennyi információt lehet belőle kinyerni. Láthatjuk, hogy az eddig felsorolt adatok valóban lekérdezhetők a böngészőprogramunkon keresztül, valamint, hogy az egyes bejegyzések milyen gyakran fordultak elő az eddig tesztelt több, mint kétmillió látogató között.

Az eddig leírtakat nevezhetném a böngészőből való adatelérés hagyományos megközelítésének és ezeket összekötve már relatíve egész jól tudunk azonosítani egy felhasználót. A teljesség kedvéért mindenképp érdemes megemlíteni az új technológiák adta lehetőségeket is, mely

alatt elsősorban a jelenleg is fejlesztés alatt álló HTML5 szabványt értem*. A célhoz egyik leginkább illeszkedő újdonság a geolokalizáció használata, mely segítségével pontosíthatjuk az eddigi eljárásunkat és egészen konkrét pozíciót rendelhetünk a látogatónkhoz (az eddigi IP cím visszafejtéshez képest). A helymeghatározás mellett számtalan egyéb lehetőségünk van még; lekérdezhethetjük például a kapcsolat módját (vezetékes, WIFI, mobilnet stb.) és kibővíti az eszközkészletünk helyi adatbázisokkal, melyek a sütithez hasonlóan a böngészőben tárolnak adatokat.

Az eddigieket összefoglalva tudunk azonosítani hálózatokat (szolgáltatói adatok), azokon belül számítógépeket (környezeti és egyéb tulajdonságok lenyomata) és az egyes számítógépeket használó felhasználókat (böngészőhasználati szokások). Az első két komponenst kellőképp megbízhatónak érzem, azonban az emberek azonosítása kapcsán sok kételyem támad. Csupán elegendő arra gondolni, hogy a böngészők között hármat (Mozilla Firefox (36,6%, Google Chrome 36,3%, Internet Explorer 19,5%)) használ az internetezők 92,4%-a (W3Schools, 2012). A piac igen koncentrált, ami azt sugallja, hogy egy számítógépen (amit több ember használ) a mi szempontunkból legjobb esetben lesz majd két-három böngészőprogram, amik közül az emberek egyéni preferenciáik alapján tudatosan és hosszú távon használnak egyet. Érezzük, hogy ez túlságosan merev és tulajdonképp lehetetlen elképzelés, hiszen egy Windows operációs rendszerrel szemben a Mac-felhasználók – az internetezők 9,1%-a – közül minden második használja a Safari-t (W3Schools, 2012), ami már alapjaiban is megrengeti a feltételezésünket.

Egy másik aggályom a felhasználói azonosítás ezen módjával, hogy két külön rekordunk lenne sokakról az okostelefonok és táblagépek miatt. Mivel feltételezzük, hogy egyvalaki csak egy böngészőt használ, ezért amikor mobilkészületről és hagyományos (asztali vagy hordozható) számítógépről is látogatjuk az oldalt, akkor az ottani tevékenységünk kettéválik. A következő alfejezetben erre a problémára mutatok megoldást.

* Annak ellenére, hogy a HTML5-nek még nincs végleges verziója és folyamatos tervezés alatt van, már több egyértelmű jel mutat arra, hogy nagy sikere lesz. Konceptiójában alapvetően igyekszik megreformálni a webet leíró technológiákat (HTML, CSS, JS) és maga az a tény, hogy a Microsoft 2011. decemberétől megszüntette a Silverlight támogatását és a konkurencia Flash-t szolgáltató Adobe is inkább ebben a technológiában látja a jövőt, is e mellett szól. Ezen kívül már több oldal is HTML5-re épül és a böngészők is támogatják az újabb szolgáltatásait.

KÖZÖSSÉGI OLDALAK FELHASZNÁLÁSA

A közösségi oldalak népszerűségét nem kell bemutatnom és pontosan ez az amiért segítségünkre lehetnek az online azonosítás kapcsán. A személyek felismerése okozott az eddigiekben problémát, hiszen a számítógépeket egész jól tudtuk lokalizálni az eddigiek alapján. A megoldásom azon alapszik, hogy ki kell választanunk legalább egy olyan nagyon elterjedt online szolgáltatást, amit a célközönségünk megfelelő arányban használ. Ez még nem lenne elegendő, hiszen azzal nem jutunk sokkal előrébb ha a mi oldalunk látogatóiról tudjuk, hogy közel mindenki rendel vagy rendelt már a netpincérről, mert ehhez a szolgáltatáshoz nem tudunk külső félként hozzáférni, nincs hozzá alkalmazásprogramozási felület (API, Application Programming Interface).

A cél tehát nem csak az, hogy olyan internetes szolgáltatást kell találni, amit az emberek használnak, hanem amihez mi is hozzáférünk. Ha a közösségi oldalakat nézzük, akkor láthatjuk, hogy a Facebook dominanciáját a Google Plus-nak (Google+) sem sikerült megtörnie. A közösségi oldalnak közel 840 millió felhasználója van világszerte, amiből Európa ad majdnem 232 millió főt, magyar viszonylatban pedig közel négy millió felhasználó van, ami az aktív internethasználók 64,28%-a (SocialBakers.com, 2012). Ez azt jelenti, hogy háromból két magyar internetező rendelkezik Facebook fiókkal. A jó hír még az, hogy a Facebook-hoz egy nagyon kiterjedt és jól támogatott API társul, amivel lényegében bármit meg tudunk csinálni az oldalunkon belül, amire szükségünk lehet.

A koncepció tehát az, hogy a közösségi oldalon használt egyéni azonosítójával biztosan tudjuk azonosítani a látogatókat az internetezés helyétől és platformtól függetlenül. Technológiai megvalósítás szempontjából ha érkezik egy új látogató az oldalunkra, akkor bejegyzünk az adatbázisunkba mindent amit tudunk róla, illetve célszerűen azt amire szükségünk van. Mindenképp érdemes rögzíteni az IP címet, a számítógép tulajdonságait (és/vagy ezek lenyomatát) és a Facebook azonosítóját amennyiben rendelkezik aktív kapcsolattal, ha nem, a mező üresen marad. Későbbi látogatás során persze kell biztosítani lehetőséget a globális azonosító utólagos bekötésére, amihez érdemes sütiket (cookies) vagy lokális adatbázist (HTML5 esetenként) használni a kézenfekvőbb megvalósítás kedvéért.

Amit így kapunk az egy olyan rendszer, ami akár több oldalon is képes működni, össze tudja kötni az ezeken megfigyelt felhasználói viselkedést, amit elemezve különböző előrejelzéseket, becsléseket, ajánlásokat stb. tud tenni akár a felhasználói élmény növelése, akár az üzemeltetői értékteremtés érdekében – de ezt még a későbbiekben részletezem.

Én a dolgozatom során egyéni azonosításra a Facebook fiókot fogom használni, mert Magyarországon ez a legdominánsabb közösségi portál, azonban szeretném megemlíteni, hogy számos alternatíva létezik. Talán jobb választás is lehet egy olyan e-kereskedelmi óriás, amely szerteágazóbb szolgáltatás-portfólióval rendelkezik – gondolok itt a Google-re. A vállalat igyekszik a szolgáltatásait összekötni (ez olvasható a téma létjogosultsága című fejezet idézetében is), amik között ott a Gmail, YouTube, GoogleMaps, Google+ és ha ezek felhasználói összesen nagyobb részt képviselnek az internetezők között, mint a Facebook 64,28%-a, akkor érdemes váltani.

Jogosan merülhet fel az a kérdés, hogy mi lesz azokkal, akiket nem tudunk pontosan felismerni? Sajnos itt el kell fogadnunk a keletkező hibaarányt és használjuk a hagyományos módszert a globális azonosító nélkül.

WIDGET-EK

Az előző részben tárgyalt közösségi oldalakhoz kapcsán említettem, hogy némelyikeknél igen fejlett API-val találkozhatunk, de általában elmondható, hogy legalább egy „tetszik” vagy „megosztás” funkció elérhető a programozók számára. A Facebook például számos beépülő modult (widget-et) bocsát a webfejlesztők rendelkezésére, amikkel percek alatt, könnyedén berakhatunk oldalunkra például egy Tetszik gombot.

Ez az azonosítás egy másik oldala, mert nem magukhoz az oldal üzemeltetőihez jut el az információ, hanem az adott közösségi oldalhoz. Jelen esetben tehát azt mondhatjuk, hogy ha egy oldalba be van építve a Facebook Like widget-je, akkor tulajdonképp a Facebook tudja az oldal látogatóiról, hogy miket néznek és a későbbiekben felhasználhatja szolgáltatásai javítására. Ezzel arra utalok, hogy egyre jobban tudják finomítani a felhasználóikról tárolt profilekat, aminek segítségével a későbbiekben például sokkal célzottabb hirdetéseket jeleníthetnek meg nekik.

VÉDEKEZÉS ALAPVETŐ MÓDJAI

Miután bemutattam a technológiai megvalósítás egyes lépéseit, nézzük meg, hogy a látogatók hogyan tudnak kijátszani minket – mivel az egyes lehetőségeknél felhívtam a figyelmet a gyengeségekre, így ez megkönnyíti a feladatot.

Felhasználóként több módunk is van arra, hogy elbújjunk a szolgáltatók elől. Először is ha nem használunk Facebook-ot, akkor nyilvánvalóan pontatlanabb lesz a nyomkövetés. Azonban, mint ahogy már fentebb említettem több szolgáltatással is helyettesíthetnénk a közösségi oldal nyújtotta előnyöket, így szintén hasznosnak tartom a kijelentkezés opció sűrű használatát – főleg a nagyobb volumenű internetes oldalakról, mint például a levelező alkalmazások, a közösségi oldalak vagy bármi egyéb portál, ami mondjuk OpenID bejelentkezést támogat. Ez az utóbbi technológia a *facebook-connect*-hez hasonlóan lehetővé teszi azt, hogy több különböző weboldalra egy már meglevő fiókunkkal (pl. Yahoo) tudjunk bejelentkezni (OpenID.net, 2012).

Egy másik alapvető biztonsági teendő a sütik törlése és esetleg tiltása. A sütik olyan kis adatcsomagok, melyeket a webalkalmazás helyez el a kliens böngészőben. Könnyen találkozhatunk ezeknek egy illegális, vagy inkább morálisan megkérdőjelezhető felhasználásával, ami az internetes anonimitáshoz az alábbi módon kapcsolódik. Tegyük fel, hogy egy portálon különböző termékeket, szolgáltatásokat értékesítünk, melyek között a látogatóink kereshetnek, válogathatnak. Ha valaki vásárol egy prémium minőségű terméket, akkor erről készítünk egy *cookie*-bejegyzést és majd ha legközelebb is nézelődik nálunk, akkor a listázásoknál előrébb kerülnek a drágább találatok vagy esetleg egyenesen nagyobb árrést alkalmazunk vele szemben. Ehhez hasonló átverésről hallhattunk tavaly márciusban a Ryanair légitársasággal kapcsolatban, akik az előbb felvázolt trükköt alkalmazták a repülőjegyeik árazásakor (AirObserver.wordpress.com, 2011). Ezt végül nem lehetett bizonyítani és közönséges pletykaként maradt meg a köztudatban, azonban mindenképp érdemes gondolni arra, hogy a sütikben ilyen jellegű információk is rejtőzhetnek. Persze ennek nem csak etikátlan alkalmazásai vannak, mert a rögzített adatok szolgálhatnak azt is, hogy pontosabban, jobban, személyre szabottabban jelenítse meg az oldal az egyes találatokat és persze nyilvánvaló, hogy nem a visszaélések érdekében születtek meg a sütik.

Az előző tipphez kapcsolódva, úgy is csökkenthetjük az azonosítás esélyét ha böngészőnket inkognitó módban használjuk. Egy-két éve már minden modernebb webböngésző rendelkezik

ilyen lehetőséggel, aminek lényege, hogy nem tárol semmi adatok az aktuális munkamenetből (jelen esetben nyilván a sütik a fontosak), tehát például nem rögzíti az előzményeket, törli a sütiket és kijelentkeztet az előre beléptetett szolgáltatásokból.

A hagyományos óvintézkedések mellett vannak még egyéb lehetőségeink is. A Ghostery egy beépülő modul, ami a webböngészőben fut és „nyomoz a nyomkövetők után” (Ghostery.com, 2011). Működése során, miközben látogatjuk az oldalakat, detektálja azon programkódokat, amik a hirdetésekért felelnek, amivel követik viselkedésünket vagy amikkel bármiképp hozzáférnek hasonló információkhoz és lehetőséget ad a felhasználónak arra, hogy blokkolja is ezeket. Hogyan is működik ez a plugin? Tulajdonképp ahhoz, hogy felismerje ezeket a „fenyegetéseket” muszáj előre beolvasnia az oldalakat, előre letöltenie a csatolt kódcsomagokat (scripteket), ami jelentős teljesítménycsökkenéshez vezet, tehát nagyon belassul a böngésző.

A fenti javaslataim mindegyikén látszik, hogy a „biztonság” képzele számos hátulütővel jár, melyek közül leginkább a felhasználói élmény radikális visszaesését hangsúlyoznám. Tegyük fel, hogy ezeket a tippeket mind alkalmazzánk. Mitől is fosztanak meg ezek minket? Először is a napi szinten látogatott portálokba minden egyes látogatás során be kell jelentkeznünk. Továbbá webshopokban sokszor megjelennek a legutóbb megnézett termékek és ezek alapján kaphatunk javaslatokat – ezekkel se számolhatunk a továbbiakban. A különböző tartalmak letiltása – gondolok itt az egyes JavaScript-ekre, hirdetési modulokra, formázási stíluslapokra – könnyedén eredményezi azt, hogy a weboldal egészen egyszerűen szétesik, nem megfelelően jelenik meg, funkcionalitásában pedig sok esetben lebénul. Ne felejtjük el, hogy mindezt a megcsonkított tartalmat még nagyon lassan is érjük el, de legalább egy-egy cikk betöltődése közben bőven lesz időnk elgondolkodni ezek szükségességén.

A kényelmetlenség mellett szerintem mindenképp érdemes megemlíteni azt, amit már feljebb írtam, hogy ezekre az adatokra, leszűrt információkra egyfajta fizetségként tekinthetünk. Ha egy ingyenes szolgáltatást használunk tudatosan ilyen védekező, elzárkózó módon, akkor azal valahol károsítjuk a szolgáltatókat. Kicsit hasonlít a helyzet a szoftverek illegális letöltésére, bár ez nagyon kisarkított, ámde szemléletes példa – természetesen a privát böngészésre bárkinek lehetősége van és ezzel nem csal.

A böngésző-beállításokon túl van vannak személyes óvintézkedések is, ilyen az általános szerződési feltételek figyelmes átolvasása, hiszen itt leírnak elméletileg minden olyat, hogy miféle adatokat használnak fel, milyen jogait vannak stb. Az ÁSZF-ekről általánosan elmondható, hogy igencsak terjedelmesek és nagyon ritka, hogy az emberek simán csak tovább-

lépnek rajtuk, ám sok reklamációs helyzetben visszatérnek rá. A Google adatvédelmi irányelvei március 1-étől megváltoztak és ezt kiírták minden szolgáltatásukhoz, valamint közölték, hogy ez az időpont után a felhasználók már nem törölhetik a róluk eddig összegyűjtött adatokat és automatikusan hozzájárulnak ahhoz, hogy a Google felhasználja ezeket az adatokat.

Hasonló figyelmesség az engedélykérések átgondolása. Például sok olyan Facebook-alkalmazás van (mondjuk maga a *connect* komponens is), ami engedélyt kér bizonyos felhasználói információkhoz és ha nem figyelünk oda, könnyen azon kaphatjuk magunkat, hogy az újonnan engedélyezett program már kis is írta az üzenőfalunkra az indiánnevünket. Tipikusan a „bejelentkezés Facebook fiókkal” opció annyira megtetszik egy-egy látogatónak, hogy meg se nézi az alkalmazás által lekért adatokat. Sokszor előfordul, hogy egy site-on engedélykéréssel találkozunk, hiszen ez egyfajta biztosíték a részükről azért, mert senki nem terelheti a felháborodását jogi utakra egy általánosságban nem tetsző dologgal kapcsolatban ha már egyszer hozzájárulását adta hozzá. Jogos kérdés, hogy akkor miért engedélyezzük ezeket a programokat, komponenseket. Erre több egészen egyszerű válasz adható: előfordulhat, hogy ellenérzésünk csak a használat megkezdése után alakul ki vagy csak nem olvassuk végig, hogy mihez is kér engedélyt az alkalmazás, hanem reflex-szerűen már kattintunk is tovább. Az a kellemetlen szituáció is adódhat, hogy teljes tudatában vagyunk annak, hogy egy adott szoftver fel fogja használni az adatainkat, de az engedély nélkül nem tudjuk futtatni így kénytelenek vagyunk elfogadni a feltételeit.

Az itt leírtakat tartom a legfontosabb alapvető teendőknek és óvintézkedéseknek az internetes biztonságunk megóvása érdekében. Persze ezekre eszközökként tekintek és szerintem mindenki összeállíthatja belőlük a számára optimális eszközkészletet. Ha valaki nagyon érzékeny az anonimitására, akkor valószínűleg a fent említett „eszközökből” sokat fog használni, de ez – ahogy már hangsúlyoztam – a felhasználói élmény csökkenésével jár. Személyes véleményem szerint leginkább az egyes alkalmazások jogosultságigényeire kell nagyon figyelnünk, mert ebből sok kellemetlenségünk adódhat. A süti-mentes böngészést és a különböző beépülő modulokat, melyek letiltanak tartalmakat túlzásnak tartom, mert ezek könnyedén szabotálhatják számomra fontos tartalmak megjelenését, alkalmazások helyes működését és olyan kényelmi szolgáltatásokat korlátoznak, amik összességében megkönnyítik a mindennapi böngészést. Az inkognitó mód tapasztalataim szerint nem csak akkor lehet jó, amikor nem akarjuk, hogy böngészőnk megőrizze a cookie-kat, hanem akkor is, amikor kedvenc böngészőnket duplikálni szeretnénk. Erre akkor lehet szükség, amikor egy adott munkamenetben futó alkalmazást több példányban akarunk futtatni – például egy ismerősünk a mi számítógépünkön

lép be egy közösségi oldalra vagy egy levelezőrendszerbe, melyre mi is be vagyunk jelentkezve, de nem szeretnénk megszakítani munkamenetünket.

A VÉDEKEZÉS EGY MÁSIK OLDALA

Az előző alfejezetben leírtam, hogy mit tartok az online anonimitás védelme érdekében leg-alapvetőbb eszközöknek, teendőknek. Jogosan merülhetett fel a kérdés, hogy miért illetem ezeket az alapvető jelzővel. Azért, mert van még sok egyéb lehetőségünk nem is csupán megtartani névtelenségünket, hanem egyenesen álcázni azt.

Abból kiindulva, hogy a korábbi fejezetekben kiemeltem, hogy ha egy oldal monitorozza viselkedésünket egy szolgáltatásért cserébe, akkor az adataink bizonyos értelemben fizetségnek tekinthetők, így az ebben a fejezetben bemutatott trükköket, cseleket, alkalmazások inkább nevezhetném támadásnak, mint védekezésnek, hiszen ezekkel szándékosan megtevesztjük az oldal üzemeltetőit és ellehetetlenítjük a pontos adatgyűjtést. Ezen alkalmazásokhoz működésének megértéséhez megint visszanyúlok a technológiai háttérnél leírtakhoz.

A USER AGENT MÓDOSÍTÁSA

Rögtön az első kategóriába soroltam a böngészőprogram tulajdonságait, melyeket később a felhasználó azonosításához kötöttem. Ezeknek az adatoknak a legfontosabbika az úgynevezett *User Agent* HTTP fejléc, ami azonosítja a böngészőt: meghatározott formátumban tartalmazza a termék nevét és verziószámát, valamint egyéb információkat, mint például a gyártó, a platform és az operációs rendszer. Értelemszerűen ha valahogy tudjuk módosítani ez az információt böngészőváltás nélkül, akkor becsaphatjuk a követőrendszereket.

Könnyen találhatunk valamilyen segédalkalmazást, beépülő modult, mely segítségével manipulálhatjuk a webböngészőnk által küldött *User Agent* fejlécet. A Mozilla Firefox egyik kiegészítője, az IE Tab, lehetőséget nyújt arra, hogy egy lapot ne a Firefox motorjával fordítsunk le, hanem a beépített Internet Explorer motorjának segítségével. Ezt értelmezhetjük úgy is, hogy tulajdonképpen futtatunk egy Internet Explorer-t a Firefox-on belül és azt használjuk az oldal megjelenítésére. Ez a beépülő modul egy nagyon hasznos eszköz a webfejlesztők körében csakúgy, mint a másik lehetőség, amit bemutatok a böngésző által küldött fejlécek meghamisítása céljából. A Google Chrome fejlesztői eszközei között (CTRL+SHIFT+I) a beállítások alatt van egy olyan opció, hogy *Override User Agent*. Ha engedélyezzük ezt, akkor

lehetőségünk van választani nem csak az Internet Explorer és a Firefox különböző verziói közül, hanem egyes mobilböngészőkké is tudjuk hamisítani a Chrome-ot. Ha alaposabban megnézzük a legördülő listát, láthatjuk az alján, hogy mi magunk is begépelhetünk egy ott nem szereplő *User Agent*-et – akár hibásat is. Fontos megjegyezni, hogy ez nem módosít semmit a böngésző motorjában, hanem csak ezt a fejléct módosítja a webserverek irányába. Miért ad a Google egy ilyen eszközt a felhasználói kezébe, aminek segítségével valamilyen mértékben kijátszható az online azonosítás? A kérdés megválaszolásához fontos tudni, hogy a webfejlesztésben az egyik legnagyobb nehézség lekezelni a böngészők különbözőségeit, de ennek egyik módja, hogy *User Agent* alapján határozzák meg, hogy például egy látogatónak melyik stíluslap írja le a megjelenő oldalt vagy egyenesen meg se jelenítenek olyan komponenteket, amik hibásan jelennének meg nála. Ezzel részben meg is válaszoltam a kérdést, tehát ez az opció nem a mindennapi felhasználók kedvéért elérhető, hanem a fejlesztők támogatására – ahol pedig nagy versenyt vívnak a Firefox-ba beépülő Firebug-gal. Szándékosan fogalmaztam úgy a kérdésben, hogy ezek segítségével „valamilyen mértékben kijátszható az azonosítás”, hiszen feltételezhetjük, hogy a Google rendszerei jóval szofisztikáltabbak annál, minthogy egy-két kattintással átverjük őket. Ennek ellenére nem kell elkeserednünk, mert semmi sem reménytelen.

SZOLGÁLTATÓI ADATOK HAMISÍTÁSA

A technológiai résznél második csoportot a szolgáltatói adatok képezik, elsősorban az IP cím. Nem meglepő, hogy ezt is tudjuk álcázni, azonban ennek valamivel bonyolultabb módszerei vannak.

PROXY SZERVEREK HASZNÁLATA

A mindennapokban is sokan használnak proxy szervereket, amik lényege röviden annyi, hogy kliensként kapcsolódunk rájuk és rajtuk keresztül megy a forgalmunk. Tehát a hagyományos kliens-szerver kérések közé épülnek be, így a szolgáltatói oldalon nem a felhasználó, hanem a proxy szerver IP címét látják. Ennek a technológiának a használatának lényege nem elsősorban a követésünk megakadályozása és azoknak a szolgáltatóknak a megtévesztése, hanem biztonsági, adatszűrési és forgalomoptimalizálási szempontok. Biztonsági oldalról betölthetik a tűzfalak szerepét, ugyanitt megvalósítható az adatszűrés – tehát bizonyos tartalmakat vagy oldalak látogatását lehet korlátozni. Forgalomoptimalizálásban gyorsítótárazó szerepet tölthetnek be. Ez akkor a leglényegesebb, amikor sok kliens nézi ugyanazokat a tartalmakat a

szerveren keresztül. Ilyenkor célszerű eltárolni ezeket a cache-ben és a már letöltött adatokat visszaküldeni, mert ezzel nagy mértékben csökkentjük a lekérések teljesítésének válaszidejét.

IP CÍM CSERÉJE: HIDE MY IP

Az IP cím hamisítás maga nagyon alapvető komponense egy hacker eszközkészletének, de átlagos felhasználók is könnyedén élhetnek vele. Könnyű találni olyan programot, mely nagyon egyszerűen megteszi ezt nekünk, ilyen például a képen látható Hide-My-IP (hide-my-ip.com, 2011). A program lényege, hogy felcsatlakozunk egy általunk választott ország távoli kiszolgálójára és annak az azonosítóját fogjuk használni. További opcióként megjelenik még, hogy ezt az IP címet automatikusan is tudja váltogatni, melynek segítségével teljes mértékben megvalósíthatjuk az anonimitásunk védelmét. Egyéb alkalmazási területe az olyan tartalmak megtekintése, melyek csak bizonyos országokból elérhetők. Persze ha belegondolunk ennek a módszernek is van bőven hátulütője. A legnagyobb probléma, hogy mivel egy (nagyon) távoli címre irányítjuk át magunkat, ezért a kérések átviteli ideje drasztikusan megnő és az effektív sávszélességünk csökken, tehát nagyon lelassul a tartalmak letöltése, az oldalak megjelenítése.

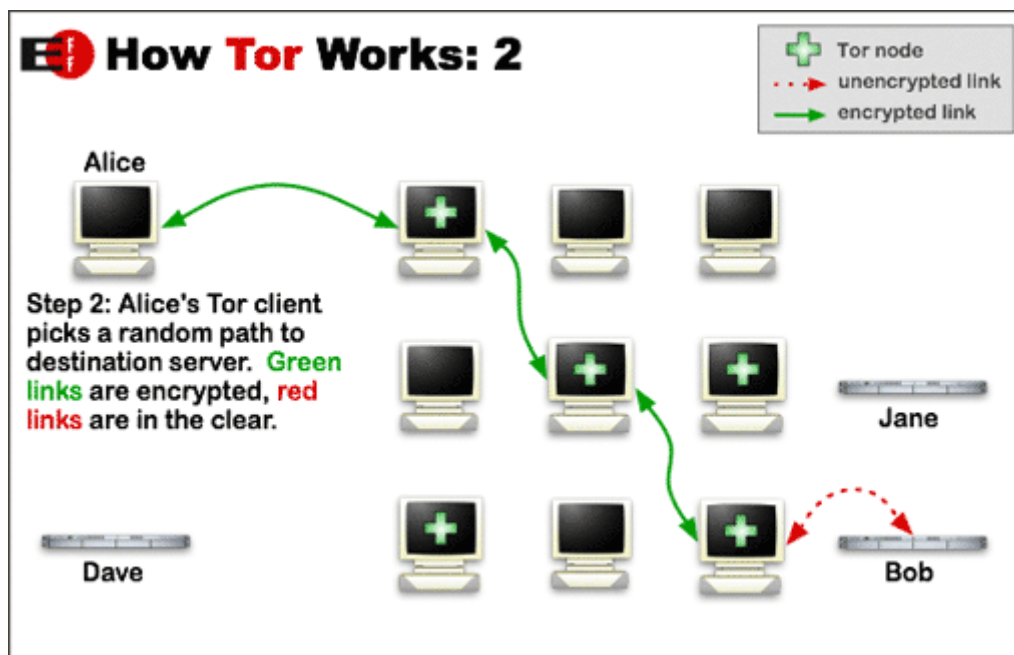


1. ábra HideMy ip ablaka

(Forrás: hide-my-ip.com)

BELÉPÉS EGY TITKOSÍTOTT HÁLÓZATBA: A TOR PROJEKT

Egy igen szofisztikált és az EFF által is ajánlott módszer internetes szabadságunk megőrzésére a TOR (The Onion Router) projekt. Ennek lényege, hogy önkéntesek által üzemeltetett szerverek alkotta hálózaton keresztül folyik a kommunikáció kliens és server között egy titkosított és több hálózati csomóponton áthaladó kérés segítségével.



2. ábra Tor hálózat működése (Forrás: torproject.org)

Az ábra alapján – ahol Alice küld üzenetet Bob számára – könnyebb megérteni a rendszer működését. Először is szükségünk van a TOR ingyenes kliensprogramjára, ami beköt minket a hálózatukba és kiválaszt egy véletlenszerű útvonalat a csomópontokon keresztül, amin végighalad a titkosított információ és végül a kilépési ponton át megérkezik Bobhoz. A hatékonyság kedvéért a program tíz percenként generál új véletlenszerű útvonalat. A legfontosabb része ennek a modellnek, hogy a TOR-hálózaton belül az információ titkosítva halad át és a sok szerver között tulajdonképp visszakövethetetlen lesz, ami a célszerver számára elrejtí, hogy valójában honnan is jött a hozzá beérkező kérés (torproject.org, 2011).

A TOR weboldalon olvasható, hogy kik veszik igénybe szolgáltatásukat, illetve pontosabban, hogy kiknek miért lehet ez előnyös. A magánszemélyeken kívül szerepelnek itt például katonai szervezetek, újságírók, üzleti vezetők, bloggerek és informatikusok is és mindegyik csoporthoz tartozik legalább három érv. Ezeknek a bemutatása túlmutat dolgozatom keretein, de érdekes belegondolni abba, hogy egy katonai információgyűjtés során, amikor megfigyelnek egyes oldalak veszélyes lehet ha a látogatott oldalak szerverti naplófájljaiban katonai IP címek jelennek meg. A példa másik oldalát nézve egy bűnöző vagy terrorista is igyekszik – legalábbis igyekeznie kellene – nem leleplezni magát az általa látogatott weboldalakon keresztül visszafejtett információk alapján.

A TOR projekt nagyon alkalmasnak tűnik arra, hogy védjük internetes anonimitásunk, azonban nyilván magával hordozza a hálózatán belüli bonyolult, több csomópontos útválasztás –

ami egyben a nagy előnye is – miatt megnövekedett átfutási időből adódó felhasználói élménycsökkenést.

KÖRNYEZETI ADATOK MÓDOSÍTÁSA

A technológiai leírásomban utolsó kategóriákba sorolt környezeti adatok hamisítása nehezebb művelet – olyannyira, hogy talán nem is érdemes vele próbálkozni. Elképzelésem szerint járható út, ha virtuális gépekkel próbálkozunk, hiszen ilyenkor egyértelműen más a környezet. A virtuális gépek lényege, hogy a hagyományos rendszerünkön futtatunk több alrendszert például a vmware nevezetű szoftver segítségével (vmware.com). Ez úgy működik, hogy megosztjuk gépünk kapacitását és bizonyos erőforrásokat átallokálunk és lefoglaljuk a program számára majd ezeket felhasználva elkészül egy virtuális gép, amire közel bármilyen operációsrendszert feltehetünk, így tulajdonképp több különböző „számítógépet” tudunk futtatni egy fizikai gépen.

A fenti néhány példán is jól látszik, hogy van lehetőségünk védekezni a minket követő rendszerek ellen, de még egyszer kiemelném, hogy az ebben a fejezetben bemutatott technológiák, trükkök inkább tekinthetők agresszív védekezésnek, támadásnak az oldal szolgáltatóival szemben.

AZONOSÍTÁS ÉS KÖVETÉS A GYAKORLATBAN

YOUHAVEDOWNLOADED

Az online azonosítással kapcsolatos első gyakorlati példámban egy olyan oldalt mutatok be, aminek a léte merőben megdönti azt a tévhitet, amit az internetes anonimitásról gondolunk.

Az internetes peer-to-peer fájlmegosztási technológiában a BitTorrent protokoll hatalmas népszerűségnek örvend, főleg az illegális tartalmak le- és feltöltése kapcsán. 2012 januárjában jelentette be a San Francisco-i BitTorrent Ltd. – akik a két legkedveltebb torrentklienst (BitTorrent, µTorrent) szolgáltatják, hogy világszerte a havi aktív felhasználók száma 2011. decemberében átlépte a 150 milliót (bittorrent.com, 2012). Több különböző torrentoldal létezik, amiken keresztül a felhasználók számos illegális és lopott tartalmat tölthetnek le, ezek között van nyilvános és privát elérhetőségű is – a különbség nyilván az, hogy a publikus oldalról bárki tölthető, a zártakhoz viszont csak egy korlátozott felhasználói kör férhet hozzá. Ezen kívül a privát oldalak többségében megnyugtatnak minket, hogy nem naplózunk felhasználói adatokat (elsősorban IP címeket).

Tehát azt mondhatjuk, hogy van egy valamilyen szinten mindenképp zárt közösség, aminek tagjai tiltott tartalmakat osztanak meg és van még olyan köztük, ami külön biztosít is minket az anonimitásunk megőrzéséről. Valószínűleg ezek láttán kevesekben merülne fel a kérdés, hogy vajon mégis lehetséges-e, hogy visszakövetik a letöltőket, illetve, hogy kinek állna ez érdekében.

A YouHaveDownloaded.com oldalon viszont egy nagyon meglepő szolgáltatást találunk: azt állítják, meg tudják mondani, hogy milyen torrenteket töltöttünk le (IP cím alapján) – sőt mi több, lehetőségünk van rákeresni mások miket vagy, hogy egyes tartalmakat kik töltenek. Maga az oldal elég egyszerű – tulajdonképp a főoldala a lényeges, ahol egyből kapunk egy keresősávot, alatta pedig egy értékelést. Az ő adatbázisukban jelenleg több, mint 55 millió felhasználó és 123 ezer torrentfájl szerepel, melyek mérete meghaladja a 123 TB-ot – ez nyilván kicsinek tűnik a világszerte torrenzetők számához képest, de mégis figyelemre méltó. Hogyan lehetséges ez és mi a céljuk?

Mindenek előtt, mivel az azonosítás IP cím alapján történik így visszautalok a böngészőből elérhető adatok között már említett statikus és dinamikus címkiosztás torzító erejére. A web-

lapon több hozzászólás is igyekszik hangsúlyozni, hogy az itt kapott eredmény egyáltalán nem perdöntő, hiszen a dinamikus IP cím-kiosztást nem veszik figyelembe, ami néhány esetben okozott is problémát a „felhasználók” számára.

A feltett kérdés megvalósíthatósági részének megválaszolásához egy a lifehacker.com oldalon 2011 decemberében megjelent cikk segít, miszerint a YouHaveDownloaded mögötti üzemeltetők figyelik a publikus torrentoldalak 20%-át. Ez szerintem úgy lehetséges, hogy valójában ők is feltöltik azokat a bizonyos tartalmakat, hiszen a peer-to-peer technológia lényege, hogy egyenrangú felek között áramlik az információ, tehát nincs dedikált szerver és kliens, így a csomópontok kommunikálnak egymással. Ha valóban beépülnek a torrentoldalak hálózataiba, akkor egész egyszerűen tudják társítani a fájlokhoz az IP címeket, hiszen ezek közvetlenül elérhetők bármelyik torrentkliensből, ahogy azt az alábbi ábra is szemlélteti.

Files	Info	Peers	Ratings	Speed						
IP	Client	Flags	%	Down Speed	Up Speed	Reqs	Uplo...	Downloaded	Peer dl	
46-29-143-219.boly.hu [uTP]	µTorrent 3.1.2	D P	100.0	910.0 kB/s	0.9 kB/s	146 0		26.8 MB	819.1 kB/s	
94-21-236-252.pool.digikabel.hu [uTP]	µTorrent 3.1.2	D P	100.0	942.1 kB/s	0.9 kB/s	160 0		17.9 MB	819.1 kB/s	
catv-89-134-149-182.catv.broadband....	µTorrent 2.2.1	D P	100.0	686.4 kB/s	0.7 kB/s	146 0		17.0 MB	819.1 kB/s	
catv-80-98-75-241.catv.broadband.hu...	µTorrent 3.1.2	D P	100.0	376.3 kB/s	0.4 kB/s	62 0		12.8 MB	819.1 kB/s	
79-113-46-157.rdsnet.ro [uTP]	µTorrent 3.1.2	D P	100.0	96.6 kB/s	0.1 kB/s	35 0		6.14 MB	819.1 kB/s	
kabelnet-254-228.parisat.hu [uTP]	µTorrent 3.0	D P	100.0	7.7 kB/s		41 0		2.93 MB	819.1 kB/s	
92-249-243-168.pool.digikabel.hu [uTP]	µTorrent 2.0.4	D P	100.0	47.4 kB/s		17 0		2.76 MB	819.1 kB/s	
dsl4E5CC559.pool.t-online.hu [uTP]	µTorrent 3.1.2	D P	100.0	1.5 kB/s		8 0		800 kB	819.1 kB/s	
lentrake.sch.bme.hu [uTP]	µTorrent 3.0	D P	100.0	11.5 kB/s		4 0		240 kB	819.1 kB/s	
2E6B9AD4.dsl.pool.telekom.hu [uTP]	µTorrent 1.9	D IP	100.0	4.7 kB/s		3 0		112 kB	819.1 kB/s	
dsl51B6B5D3.pool.t-online.hu [uTP]	µTorrent 3.1.2	D P	100.0	0.8 kB/s		2 0			819.1 kB/s	

3. ábra uTorrent Peer listázása

Maga ez az oldal azt a fenyegetést hordozza magában, hogy így gyakorlatilag bármikor bekoportathatnak hivatalos szervek az letöltők ajtaján, mert az IP címből és letöltés idejéből megállapítható az internetszolgáltató kiléte, akik biztos információkat tudnának nyújtani az ügyfelük lakcíméről és egyértelmű bizonyítékok is lennének a személyek ellen. Ez elméleti síkon működhet, de sok kibúvó vagy kétes eset adódhat. Például mi van azokkal, akik proxy szervereket üzemeltetnek? Hozzájuk valószínűleg nagyon sokszor kopogtatnának, viszont nekik vélhetően semmi érdekük nem lenne arra, hogy továbbirányítsák a hatóságokat és ha nem is naplózzák a szolgáltatásukat igénybevevőket (legalábbis nem mentenek IP címeket), akkor a kifogásuk is megvan – megjegyzésképp pont ezért az itthoni torrenttrazziák alkalmával is nagyon hasznos az üzemeltetőknek, hogy nem tárolnak efféle adatokat.

Szintén a fenti cikkben olvashatjuk, hogy a YouHaveDownloaded oldal tulajdonosainak nincs ehhez hasonló céljuk, csupán annyi, hogy ráébresszék a torrentező közösséget a tevékenységeik elrejtésének, álcázásának fontosságára és használjanak például proxy szervereket vagy VPN kapcsolatot internetes forgalmuk titkosítására. Szerintem hagyományos internet-

felhasználóként is meglepő és elgondolkodtató, hogy tulajdonképp ha egy ilyen viszonylag zárt térben sem bújhatunk egy internetes becenév mögé, akkor mennyire böngészünk biztonságosan a mindennapokban.

FRICITIONLESS SHARING

Második példám az előbb említett mindennapos böngészés biztonságához és a Facebook-hoz kapcsolódik. Először bemutatnám a probléma megfogalmazásának körülményeit. Dave Winer írt 2011 szeptember 24-én egy blog-bejegyzést, amiben kifejti, hogy mennyire félelmetesnek tartja az akkor megjelenő surlódásmentes megosztás* (*frictionless sharing*) funkciót. Az ezt használó alkalmazások, például a The Guardian hírolvasó, amivel a közösségi oldalon keresztül tudunk híreket olvasni. Amikor megnyitunk egy cikket akkor erről egy bejegyzés automatikusan (felhasználói közbelépés nélkül) kikerül az idővonalunkra és láthatja minden ismerősünk és feliratkozónk – sőt arra is lehetőség nyílik, hogy megnézzük például ki, mit olvas és mik a legnézettebb tartalmak (TheGuardian, 2011). Dave Winer ezzel kapcsolatban azt tartja félelmetesnek, hogy lényegében ha elolvas egy cikket akkor arról „értesítés érkezik” a Facebook-hoz, pedig nem osztja meg a falán, nem ír hozzá kommentet és nem is jelzi, hogy tetszik neki, csupán meglátogat egy oldalt. Azt is írja, hogy tisztában van vele, hogy tulajdonképp minden oldal, ami be van kötve a Facebook-hoz (akár egy tetszik gomb segítségével) az folyamatos visszajelzéseket küld a Facebook számára, így tulajdonképp követnek bármerre is megyünk. Ezek elleni védekezésépp azt javasolja, hogy jelentkezzünk ki a Facebook-ról, azonban néhány nappal később Nik Cubrilovic megcáfolta ezt bebizonyítva, hogy kijelentkezés után is követnek minket sütik segítségével (Cubrilovic, 2011). A „hibát” bemutató részletes leírás után két nappal közzétett egy másik bejegyzést, melyben leírja, hogy hogyan dolgozott együtt a közösségi oldal fejlesztőivel. Két célja volt: egyrészt azonosítani a kijelentkező folyamat hibáit, másrészt pedig magyarázatot kérni a megtartott cookie-k felhasználására. Ami a „hibát” illeti kijavították amit tudtak (eltávolították például azt a sütit, ami a felhasználói azonosítót tárolta*) és el is magyarázták, hogy mik a kijelentkezési folyamat lépései és,

* Saját fordítás

* Ez nagyon jó eredmény lenne, de szerintem nem jelent túl sokat. A cikk írója kifejti az egyes sütik szükségességét és van egy olyan, aminek segítségével könnyedén áthidalhatják a felhasználói azonosító hiányát. Ez az *act* nevezetű cookie, ami 1316962370811/2; formátumban tárolja minden egyes Facebook bejelentkezés UNIX-os időbélyegét milliszekundumban a perjel előtt, utána pedig az egy időpillanatban beérkező kérések sorszáma szerepel így kialakítva egy teljesen egyedi kulcsot, amihez ők nyilván eltárolják a látogató kulcsát is. Ez állítólag

hogy miért szükségesek a sütik. Hivatalos állásfoglalás szerint teljesítménybeli és biztonsági okokból volt ezekre szükség és tulajdonképp semmi sem indokolja, hogy ezen a koncepción változtassanak, így többségük továbbra is megmarad kijelentkezés után is (Cubrilovic, 2011). Szinte biztosra vehetjük azt is, hogy továbbra is követnek minket a beépülő modulok segítségével és mivel ezeket annyira szeretik az oldaltulajdonosok – ami érthető is, hiszen nagyon gyorsan tudnak implementálni egy főleg marketing szempontból meglehetősen hasznos funkcionalitást (például tetszik gomb) – villámgyorsan el is terjedtek és terjednek a mai napig is. Megoldásként Cubrilovic a kijelentkezés utáni sütitörlést javasolja vagy egy külön böngésző használatát – szerintem az utóbbi biztosabb, de akkor már jobban járunk ha nem a kedvenc böngészőnket fordítjuk az ilyen és hasonló oldalak látogatására, hanem inkább ezeknek keressünk egy külön (kevésbé preferált) programot.

MINTA PROJEKT: BCE ANONIMITY CRAWLER

BEMUTATÁS

Balogh Zoltánnak köszönhetően betekintést nyerhettem a Budapesti Corvinus Egyetem és GKINET Internetkutató és tanácsadó Kft. közös kutatásába, aminek célja a tanácsadó cég szempontjából az események terjedésének vizsgálata az informális közösségeken belül, az egyetemen pedig a dolgozatom témájához hasonló kutatáshoz gyűjtenének adatokat.

A kutatáshoz technológiailag úgy lesz megoldva, hogy weboldalakba (például az egyetem Moodle e-learning rendszere) beépítenek egy kisebb applikációt, mely nem terheli sem a kliens- sem a szerveroldali munkavégzést feltűnő mértékben. A következőkben ezt a beépülő modult fogom bemutatni és közben húzhatunk párhuzamot más követő oldalakkal.

FELÉPÍTÉS ÉS MEGVALÓSÍTÁS

Programozás-technikailag számomra legérdekesebb – és a dolgozat témájához leginkább kapcsolódó – rész maga a beágyazott JavaScript kódsor ami a főoldalakra kerül. Ez valósítja meg az adatok gyűjtését, de persze ki kell egészíteni egy jól működő és optimalizált backend felü-

csak a teljesítményadatok gyűjtéséhez szükséges és nem vezetik vissza a felhasználók azonosításához (Cubrilovic, 2011), de valahogy mégis megoldják – személy szerint meglepődnék ha nem ennek segítségével.

lettel, ami felelős az adatbázis-kapcsolatért, az új felhasználói rekordok beszúrásáért, meglevők azonosításáért és módosításáért.

Ebben az alfejezetben kiemelek néhány kódrészletet a célból, hogy demonstráljam az adatok kinyerésének módját. Még a program elején el kell döntenünk, hogy a lokálisan (azaz a felhasználó gépén) miképp rögzítünk bizonyos adatokat. Maga a tárolási mód ideális esetben a HTML5 adta LocalStorage, mely – ahogy már említettem – a hagyományos sütiket hivatott kiváltani. A program tervezője úgy döntött, hogy a felhasználó gépén eltárolja a bejelentkezési egyéni azonosítóját (*userId*) és a tárolás helyének kiválasztásához az alábbi kódrészlet segít:

```
<!-- LocalStorage, cookie -->
if (Modernizr.localstorage) {
    if (localStorage.getItem('userId') != undefined) {
        oldUserId = userId = localStorage.getItem('userId');
        data = "userId="+userId+"&hasFbId=1&";
    }
} else {
    if ($.cookie("userId") != undefined) {
        oldUserId = userId = $.cookie('userId');
        data = "userId="+userId+"&hasFbId=1&";
    }
}

<!-- OS, browser -->
var os = PluginDetect.OS, OS = "";
switch (os) {
    case 1:
        OS = "Windows";
        break;
    case 2:
        OS = "Macintosh";
        break;
    case 3:
        OS = "Linux";
        break;
    case 21.1:
        OS = "iPhone";
        break;
    case 21.2:
        OS = "iPod";
        break;
    case 21.3:
        OS = "iPad";
        break;
    default:
        OS = "Other";
}

var browser, version;

if (PluginDetect.isChrome) {
    browser = "Chrome";
    version = PluginDetect.verChrome;
}
```

Mint láthatjuk ez úgy működik, hogy a Modernizr API használatával megállapítja, hogy elérhető-e a LocalStorage, ha pedig nem, akkor sütiket fog használni és az elágazásokon belül pedig kiszedi a tároló tartalmát és befűzi a *data* nevű szöveges típusú változóba, ami majd az új adatbázis-rekord létrehozásához szükséges.

Egy másik igen hasznos kiegészítő a programban a *PluginDetect*, melyet arra használnak, hogy megállapítsák a látogató operációsrendszerét, illetve az alkalmazott böngésző típusát és verzióját. Az operációs rendszer detektálása közben megkülönböztet Windows, Macintosh, Linux, iPhone, iPod és iPad variánsokat, majd hasonlóképp kinyeri a böngészőre vonatkozó információkat is. Utóbbit a lenti elágazás mutatja be, ami azt csinálja, hogy ha a *PluginDetect.isChrome* értéke igaz, akkor a látogató

a Google Chrome-ot használja és könnyedén megkaphatjuk a program verzióját is. Természetesen ezeket az adatokat is hozzáfűzzük a data változónkhoz.

Még egy nagyon érdekes és hasznos kódrészletet szeretnék kiemelni: a geolokalizáció alkalmazását. Ezt azért tartom lényegesnek, mert látszik rajta a régi és új technológiák közti választás. Ez a kódrészlet felelős azért, hogy a látogató pozícióját rögzíteni tudjuk. Mindkét esetben IP cím alapján történik a helymeghatározás, de az elv különbözik.

```
<!-- GeoLocation -->
$.getJSON(
  'http://api.ipinfodb.com/v3/ip-city/' +
  '?key=178043c468512dd9488a9b8e5c7d92528742bdf6aa48df35fe57140007e35c4' +
  '&format=json&callback=?',
  function(position) {
    data += "positionLatitude="+position.latitude+"&";
    data += "positionLongitude="+position.longitude+"&";
  }
);

$("#geoLine a").click(function() {
  if (navigator.geolocation) {
    navigator.geolocation.getCurrentPosition(
      function (position) {
        data = "positionLatitude="+position.coords.latitude+"&";
        data += "positionLongitude="+position.coords.longitude+"&";

        updateData("geo");
      }
    );
  }
});
```

Az első esetben ismét egy külső API-t használunk, mely lehetővé teszi, hogy egy IP címből megállapítsuk a látogató városát. Ez a szolgáltatás webes API-hívással elérhető el, amihez szükséges egy egyéni alkalmazás kulcs, melyet az ipinfodb.com oldalon regisztrálhatunk, majd végül megadjuk, hogy milyen formában szeretnénk visszakapni az adatokat. Az eredmény egy JSON tömb lesz, amit könnyedén feldolgozhatunk és befűzhetjük a megfelelő információkat a *data* sztringünkbe. Fontos megjegyezni, hogy a lokalizálás ezen módja viszonylag pontatlan, ugyanis város szinten csupán 60%-os pontossággal azonosít az Amerika Egyesült Államokon belül, ország szinten viszont 99,5%-os biztonságot gaszantálnak (IPInfoDB, 2012). A városszintű magas pontatlanság elég aggasztó, de ha nincs más lehetőségünk ez is használható. Szerencsére a HTML5 adta Geolocation API helymeghatározási pontosságának konfidenciaszintje 95%-os (Popescu, 2011), azonban ennek használatához a felhasználó engedélyezése szükséges. Ezt mutatja be a második programblokk, amiben így már az alap változathoz hasonló könnyedséggel nyerhetők ki a GPS koordináták.

A fenti kódrészletekből az látszik, hogy egy azonosító és nyomkövető alkalmazás létrehozása nem lehetetlen feladat és nagyon sok segédalkalmazásra támaszkodhatunk, tehát semmi sem gátolja, hogy egyre több weblap alkalmazzon hasonló technikát a látogatói viselkedésének vizsgálatára.

FELHASZNÁLÁS

A közös kutatás célját már felvázoltam a projekt bemutatásában, tehát az egyetem részéről előnyös az együttműködés, mert a GKIE.NET megtámogatja például egy nyereményjátékkal a kutatást, így motiváltabbak a felhasználók, a tanácsadó cég pedig felhasználhatja a kinyert adatokat a saját hatáselemző modelljébe és vizsgálhatja a felhasználói kapcsolatok által kialakított informális hálózatban az információ terjedését és azt, hogy hogyan lehet a különböző eseményeket reprodukálni.

Gondoljuk el, hogy milyen jó lenne ha egy általunk készített látogatókövető rendszer terjedését nem nekünk kellene erőltetni, hanem az egyes oldaltulajdonosok, webfejlesztők lennének érdekeltek benne. Hogy mire is gondolok? Mi lenne ha nem kifejezetten nyomkövető alkalmazást próbálnánk terjeszteni, hanem mondjuk egy olyat, ami különböző szolgáltatások nyújt egy oldal üzemeltetőjének mint, hogy látogatói mennyi időt töltöttek nála, miket néztek meg, honnan jutottak a weblapra, hova mentek tovább és ezeket kiegészítenénk még különböző aggregált statisztikákkal, diagramokkal. Ezeket olvasva kirajzolódhat előttünk egy internetes óriás internetes elemzési megoldása – a Google Analytics.

GOOGLE ANALYTICS

Az előző alfejezetben a hagyományostól eltérő módon vezettem fel a Google internetes elemző megoldását. Ez egy igazán jó termék, amely „rendkívül jó rálátást biztosít webhelye forgalmára és marketinghatékonyágára”. Beköthetjük és optimalizálhatjuk AdWords kampányunkat is a szolgáltatás segítségével, pontos adatokat láthatunk a látogatók szokásairól, tevékenységeiről, így hatékonyabban alakíthatjuk marketingstratégiánkat. A fejlett és egyénre szabható különböző jelentések generálásának lehetősége és az egyes adatok grafikus ábrázolása támogatja az egyes kontrollingfunkciók működését is, ami még vonzóbbá teheti ezt a megoldást (Google Analytics, 2012).

Az előbb felsorolt eszközök és a termék ingyenessége alapján nem meglepő, hogy igen elterjedt – a BuiltWith Trends számításai alapján több, mint 15 millió weboldal használ Google

Analytics-et az általunk vizsgált 90 millió különböző oldal közül. Ez azt jelenti, hogy a Google a +1-es Google Plus widget-ek nélkül is rengeteg oldal működésébe, látogatói viselkedésébe lát bele. Még egyszer hangsúlyozom, hogy mindezt azért van lehetősége, mert egy ennyire jó szolgáltatást nyújt, amit szeretnek is az emberek.

ÖSSZEFOGLALÁS ÉS KONKLÚZIÓ

A dolgozatom írása során igyekeztem bemutatni az online azonosítás lehetőségeit, valamint a módszereket, amelyekkel a felhasználók megnehezíthetik saját maguk visszakövethetőségét. A feltett kérdések közül egy még nyitva maradt: Jelentenek-e ezek a látogatókra valami féle veszélyt?

Egyrészt ezt mindenkinek saját magának kell eldöntenie, de én alapvetően úgy gondolom, hogy ezek a módszereket nem szabad a személyekkel szembeni támadásként kezelni és fontos látni – amit egy szemléletes példán keresztül mutatnék be, hogy egy Google-t nem érdekel, hogy mi kinek vallunk szerelmet, az viszont igen, hogy hamarosan rózsát szeretnénk rendelni. Ezzel azt szeretném mondani, hogy nekik az számít, hogy minél jobb szolgáltatást nyújtsanak, hogy a felhasználók minél jobban szeressék őket és ragaszkodjanak hozzájuk, mert így a másik (például hirdetési) oldalról egyre nagyobb bevételeket tudnak realizálni.

Az okozhat még esetleg aggodalmat, hogy ha ezek a nagyon alaposan kiépített profilinformációk esetleg rossz kézbe kerülnének például egy hacker támadás során, akkor ez meglehetősen komoly problémákat szülne. Több cikk is kihangsúlyozza ezt a veszélyforrást és azt, hogy ha például a Google által tárolt titkos és néha kényes információk nyilvánosak lennének, akkor az könnyedén megtörhetné egy-egy ember életét – gondolok itt például peres eljárásokra, emberi kapcsolatokra, válásokra stb.

Összességében azt gondolom, hogy napjainkra az internetes anonimitás közel teljes megőrzése, az, hogy majdnem névtelenül böngésszünk a világhálón még megvalósítható, azonban sok-sok kényelmetlenséggel, nehézségekkel, felhasználói élmény csökkenésével és teljesítmény csökkenéssel is jár. Internethasználóként azt kell mérlegelnünk, hogy mekkora mértékű névtelenséget vagyunk hajlandóak feláldozni az egyes kényelmi funkciókért, de nagyon fontos, hogy ezeket tudatosan tegyük és az általunk használt alkalmazások által kért jogosultságokra fordítsunk kellő figyelmet.

FORRÁSJEGYZÉK

Index.hu (2012): *A Facebook olvassa az sms-einket*

http://index.hu/tech/2012/02/27/a_Facebook olvassa az sms-einket/ (2012. 03. 01)

Alasdair Allan (2011): *Got an iPhone or 3G iPad? Apple is recording your moves*

<http://radar.oreilly.com/2011/04/apple-location-tracking.html> (2012. 03. 01)

Jadeye (2011): *Tényleg kémkedig utánad az iPhone?*

<http://szifon.com/2011/04/21/tenyleg-kemkedik-utanad-az-iphone/> (2012. 03. 01)

W3Schools.com (2012): *Browser Statistics*

http://www.w3schools.com/browsers/browsers_stats.asp (2012. 03. 01)

SocialBakers.com (2012): *World Continents Facebook Statistics*

<http://www.socialbakers.com/countries/continents> (2012. 03. 02)

AirObserver.wordpress.com (2011): *Ryanair tricking passengers into paying more than necessary using cookies information?*

<http://airobserver.wordpress.com/2011/03/23/ryanair-tricking-passengers-into-paying-more-than-necessary-using-cookies-information/> (2012. 03. 02)

OpenID.net (2012): *Get an OpenID*

<http://openid.net/get-an-openid/> (2012. 03. 05)

Ghostery.com (2011): *About*

<https://www.ghostery.com/about> (2012. 03. 05)

Hide-My-Ip.com (2011): *Hide My IP 5.3*

<http://hide-my-ip.com/> (2012. 03. 10)

Peter Ecklersey (2010): *How unique is your web browser?*

<https://panopticlick.eff.org/browser-uniqueness.pdf> (2012. 03. 02)

W3C.org (2011): *14.43 User-Agent*

<http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html> (2012. 03. 07)

TORproject.org (2011): *Overview*

<https://www.torproject.org/about/overview.html.en> (2012. 03. 06.)

YouHaveDownloaded.com (2011)

<http://www.youhavedownloaded.com/> (2012. 03. 01.)

BitTorrent.com (2012): *BitTorrent and μ Torrent Software Surpass 150 Million User Milestone*

http://www.bittorrent.com/intl/es/company/about/ces_2012_150m_users (2012. 03. 20)

Alan Henry (2011): *This Web Site's Been Watching Your BitTorrent Habits and Can Show What You've Been Downloading*

<http://lifehacker.com/5867161/this-web-sites-been-watching-your-bittorrent-habits-and-can-show-what-youve-been-downloading> (2012. 03. 20)

Dave Winer (2011): *Facebook is scaring me*

<http://scripting.com/stories/2011/09/24/facebookIsScaringMe.html> (2012. 03. 24)

Nik Cubrilovic (2011): *Logging out of Facebook is not enough*

<http://nikcub.appspot.com/posts/logging-out-of-facebook-is-not-enough> (2012. 03. 24)

Nik Cubrilovic (2011): *Facebook Fixes Logout Issue, Explains Cookies*

<http://nikcub.appspot.com/posts/facebook-fixes-logout-issue-explains-cookies> (2012. 03. 24)

Guardian.co.uk (2011): *Guardian Facebook app: FAQ*

<http://www.guardian.co.uk/info/2011/sep/22/guardian-facebook-app-faq?intcmp=239> (2012. 03.24)

Nick O'Neill: *Almost 65 Million Facebook Users "Like" Things Daily*

<http://www.allfacebook.com/almost-65-million-facebook-users-like-things-daily-2010-07> (2012. 03. 24)

Index.hu (2012): *Így óvhatja meg adatait a Google-től*

http://index.hu/tech/2012/02/29/igy_ovja_meg_szemelyes_adatait_a_google-tol/ (2012. 03. 14)

Balogh Zoltán (2012): *BCE-Anonymity_crawler*

https://github.com/zols/BCE-Anonymity_crawler (2012. 03. 22)

Modernizr.com (2012): *Documentation*

<http://www.modernizr.com/docs/> (2012. 03. 22)

Eric Gerds (2012): *Browser Plugin Detection with PluginDetect*

<http://www.pinlady.net/PluginDetect/> (2012. 03. 22)

IPInfoDB.com (2012): *IP Address Geolocation XML API*

http://ipinfodb.com/ip_location_api.php (2012. 03. 22)

Andrei Popescu (2011): *Geolocation API Specification*

<http://dev.w3.org/geo/api/spec-source.html> (2012. 03. 22)

Google.com (2012): *Nagyvállalati szintű funkciók biztosítása a Google világszínvonalú platformján.*

<http://www.google.com/analytics/features.html> (2012. 03. 22)

Trends.BuiltWith.com(2012): *Google Analytics Usage Statistics*

<http://trends.builtwith.com/analytics/Google-Analyticsv> (2012. 03. 22)