A fordított névvállalás biztonsági protokollja

The secure protocol of reversed user identification

Kusper Gábor, Biró Csaba, Tajti Tibor

Eszterházy Károly Főiskola, Matematikai és Informatikai Intézet

{gkusper, birocs, tajti}@aries.ektf.hu

Absztrakt

Fordított névvállaláson azt a metódust értjük, amikor a szolgáltatást igénylő személy vagy szervezet nem teszi közé a nevét vagy azonosítóját, ugyanakkor adatai elérhetőek. Az adatok alapján lehet ajánlatot küldeni nekik, a nélkül, hogy ismernénk, kinek adjuk az ajánlatot. Ez bizonyos szempontból fordítottja a szokásos anonim tendereknek, ahol az ajánlatkérő ismert, az ajánlatok viszont név nélküliek. Erre azért van szükségünk, mert egy olyan rendszert fejlesztünk, amiben rögzíthetjük életmódunkra vonatkozó adatainkat (étkezések, testmozgások, környezeti hatások). Ezek az adatok elérhetőek lesznek életmód tanácsadóknak, úgy hogy garantálni lehessen, hogy a tanácsadók semmilyen módon nem lesznek képesek kitalálni, kihez tartozhatnak az adatok. Ugyanakkor, ha egy tanácsadó úgy látja, hogy valakinek az életmódja nagyon kockázatos, mert pl. túlsúlyos és keveset mozog, akkor a tanácsadó képes üzenetet küldeni ennek az ismeretlennek. Ugyanakkor a tanácsadóknak vállalniuk kell a nevüket.

A rendszer a következő megoldásokat használja:

- Az anonimitás eléréséhez profilok alkalmazunk. A felhasználók adatait egy neuronháló klaszterbe szervezi. Az egy klaszterben lévő adatok átlaga egy profilt ad. A tanácsadók a profil adatait látják.
- Ha van olyan adatfajta, amiből csak nagyon kevés van, pl. Kik ettek Az arany kakasban, akkor a rendszer generál néhány nem létező személyt, akik szintén rendelkeznek ezekkel az adatokkal, úgy hogy az így generált adatok ne befolyásolják az átlagot. Ez azt biztosítja, hogy ha a tanácsadók képesek lennék a profilok mögé látni, akkor is nagyon nehéz legyen megállapítaniuk, ki valós személy.

Kulcsszavak: fordított névvállalás, biztonsági protokoll.

Abstract

We understand under the reversed user identification the following method: the source of (either a person or an organization) a request does not public his or her or its name or identification number, but the data of the request can be accessed. Based on this data one can send offer for the request, without knowing the requester. This is the reverse of the usual anonym tenders, where the requestor

(or announcer) is known, but the offers are anonym. We are interested in reversed user identification, because we develop a system, where the users can record data about their way of life (eating, physical activities, environmental impacts). These data can be accessed by way of life mentors, but it must be a guarantee, that the mentors cannot find out to whom belong the data. If a mentor see that someone has a dangerous way of life (because the owner of the data has overweight and moves to little), then he or she can send to him or her a message. The mentors have to identify themselves in the message.

The system uses the following solutions:

- We use profiles to obtain anonymity. The data of the users are put in clusters by a neuron network. The data of a profile is the average of the data of the users in the cluster. The mentors can see only the data of the profiles.
- If there is a kind of data, which has very few instances, for example who has been eating in the Golden rooster restaurant, then the system generates some non-existing people with this kind of data, but the new instances should not affect the average. This solution gives us a guarantee that if a mentor could see behind a profile, then still he or she has difficulties to find out how is a real person.
Keywords: reversed user identification, secure protocol

1. Bevezetés

Az eFilter [eFilter1-7] projekt folytatása a PHA (Personal Health Assistant) projekt, amely szintén a Wit-Sys ZRt. és az Eszterházy Károly Főiskola együttműködésében jön létre.

Az eFilter projekt keretében két nagy adatbázisunk volt, az élelmiszerek és az egészségügyi adatok (personal health record). E két adatbázis vizsgálatából szűrtük le, hogy a felhasználó milyen ételeket ehet meg, amik az egészségügyi adatai, pl. diéta, alapján megengedett neki.

A PHA projekt keretében három adatbázisunk lesz: élelmiszerek, egészségügyi adatok és életviteli adatok. Az életviteli adatok tartalmazzák, fogy a felhasználó mikor milyen testedzést folytat, milyen környezetben él, milyen testápoló szereket használ, illetve milyen egyéb behatások érik.

Mindezen adatok alapján a rendszer egyik fő szolgáltatása, hogy szakértők (dietetikusok, mesteredzők, orvosok, …) tanácsokat adhatnak a felhasználóknak, hogy hogyan változtassák meg életmódjukat, hogy egészségesek maradjanak, ne betegedjenek meg.

Ebben a cikkben azzal foglalkozunk, hogyan alakítsuk ki a szakértők és a felhasználók közötti

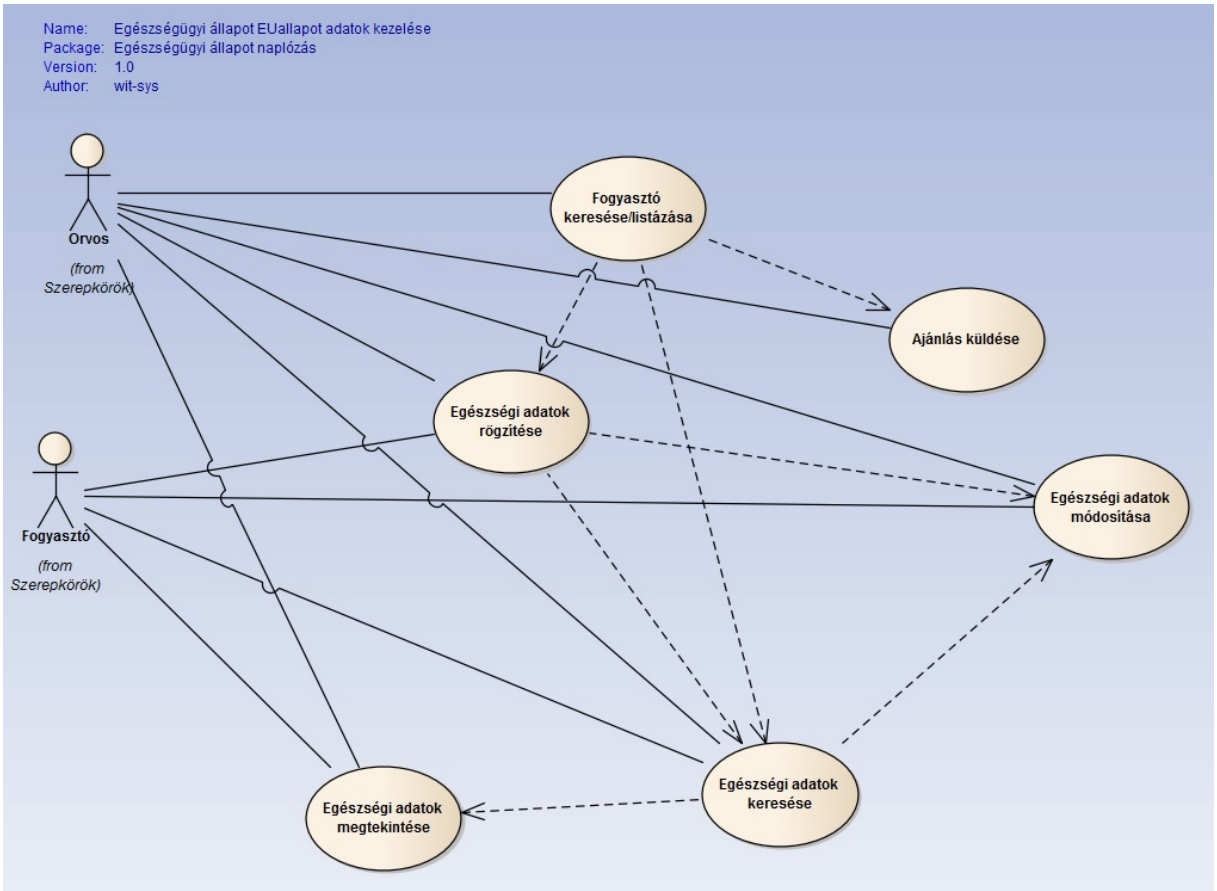
kommunikációt, hogy a szakértők ne tudhassák meg a személyazonosságát azoknak a felhasználóknak, akiknek tanácsot adnak.

2. A PHA projekt használati esetei

A PHA projekt megvalósításának tervezési szakaszában járunk. A tervezés során használati eseteket (use case) készítünk, amelyek olyan egyszerű ábrák, amiket a megrendelő és a szoftverfejlesztők is könnyen megérhetnek.

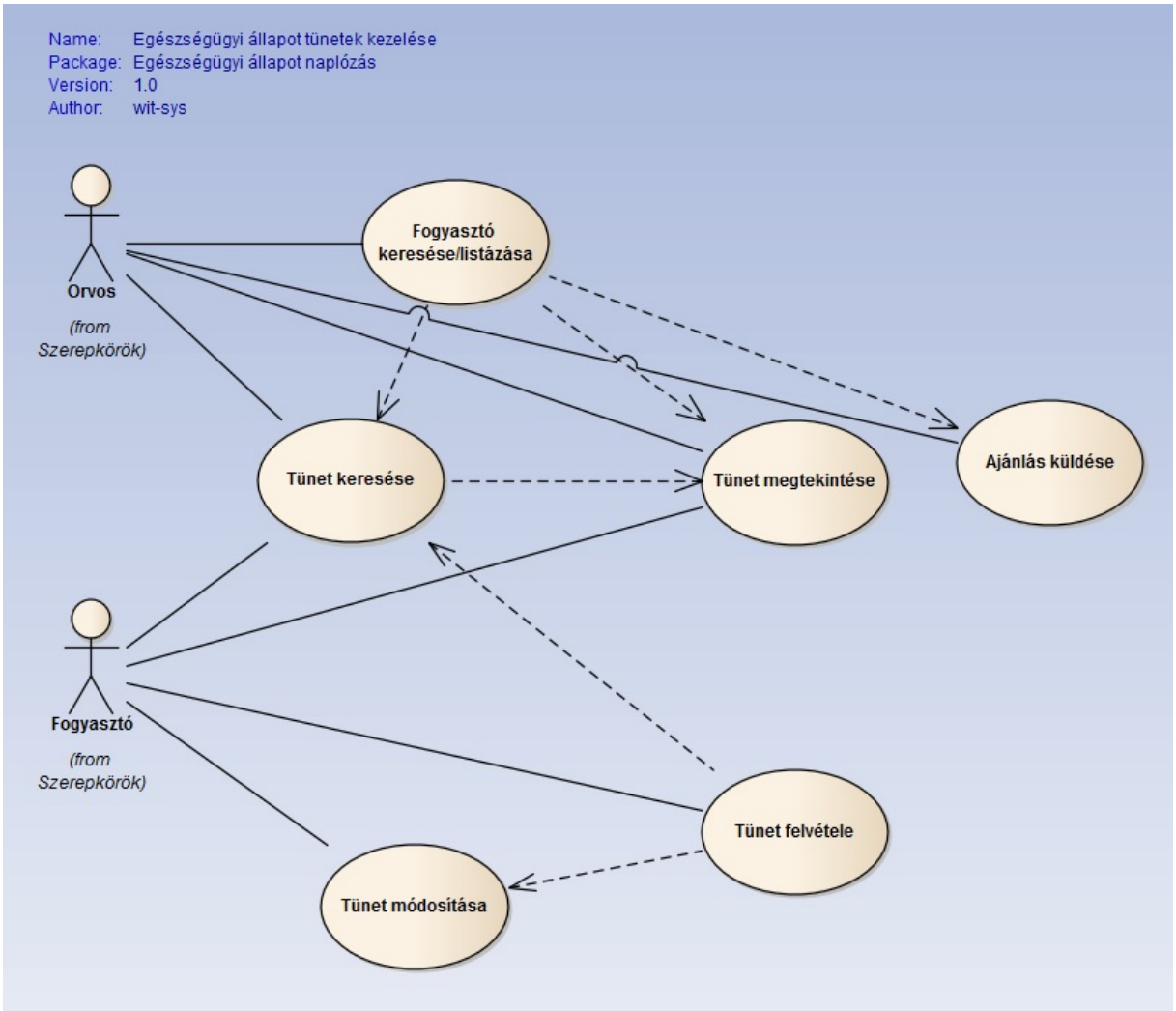
A következő használati esetek készültek el a szakértő (orvos) és a felhasználó kommunikációjáról.

2.1. Egészségügyi állapot naplózása



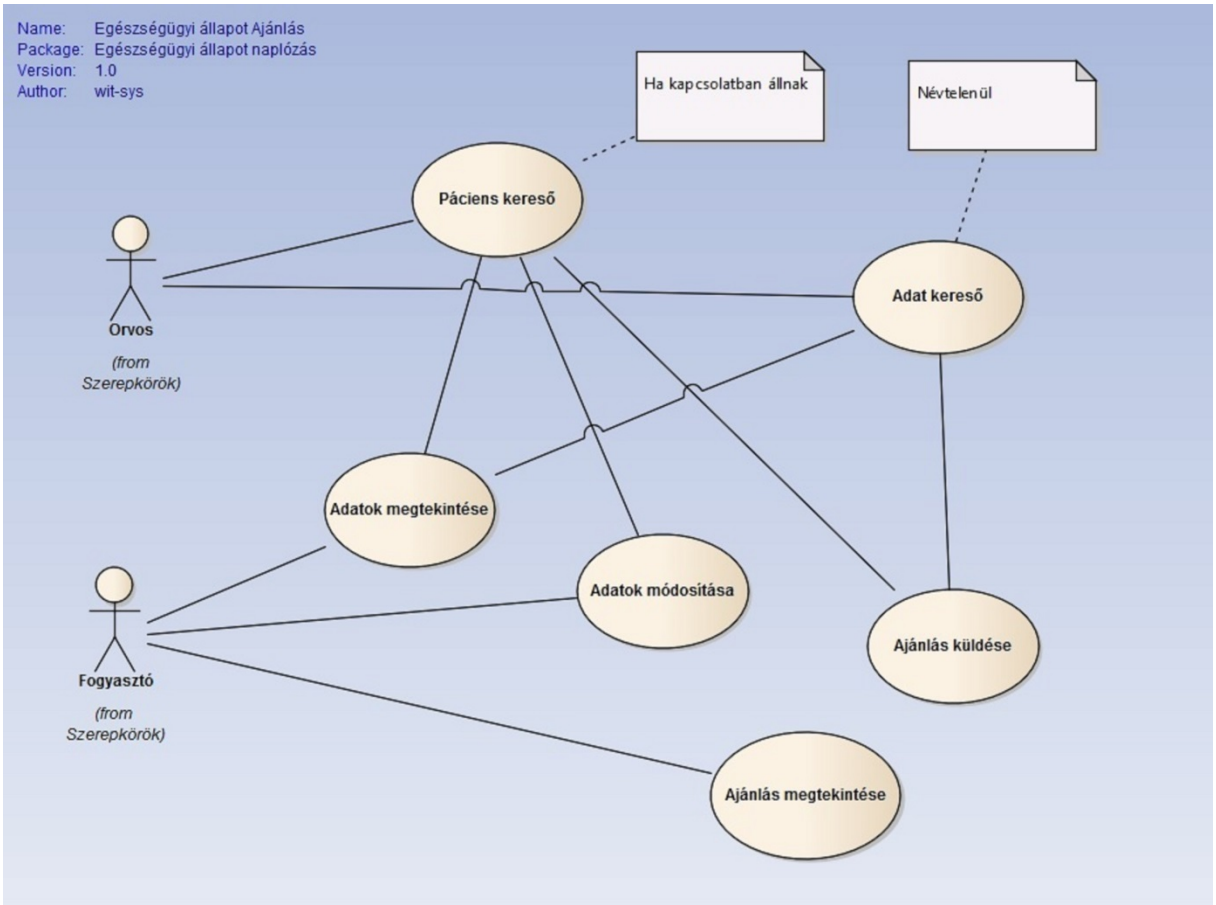
Ez a használati eset mutatja be, hogy a felhasználó (fogyasztó) képes rögzíteni egészségügyi adatait, amelyet az orvos is megtekinthet, de csak ha erre jogot kapott a felhasználótól. Ezen a használati eseten látható többek közt az ajánlás küldése is, amit egy későbbi használati eset fejt ki bővebben.

2.2. Tünetek rögzítése



Ezen a használati eseten látható, hogy a felhasználó képes a tüneteit rögzíteni a rendszerben. Ilyen tünet például a fejfájás, hasmenés, rossz közérzet, stb…. A tünetek alapján a szakérők tudnak szűrni az ajánlások küldése előtt a felhasználók közül a nélkül, hogy a személyes adataikat láthatnák.

2.3. Páciens kereső

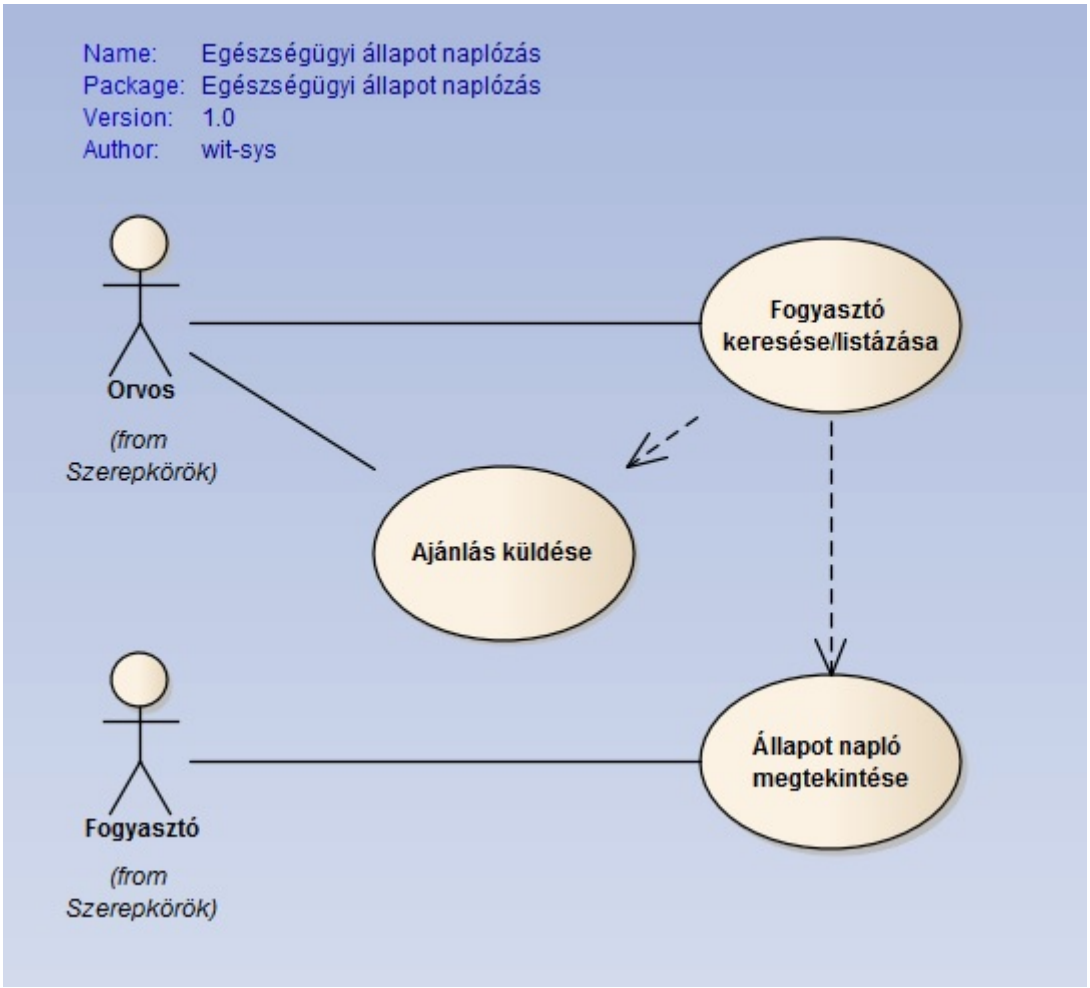


A szekértők (pl. orvosok) egy paciens kereső felülete keresztül kereshetnek olyan felhasználókat, akik megítélésük szerint a segítségükre szorulnak. A paciens keresőn keresztül leszűrhetőek például azok a felhasználók, akik túlsúlyosak, keveset mozognak és magas a vérnyomások. Illetve a szakértőnek számszerűsíteni kell, hogy milyen testsúly index esetén túlsúlyos valaki, mit jelent a kevés mozgás és a magas vérnyomás.

A szűrés után ezeknek a felhasználóknak ajánlást küldhet, ami lehet pl. egy testedzés program, vagy egy diéta, vagy a kettő kombinációja. A rendszer segít betartani ezeket az ajánlásokat, ha a felhasználó elkezdi valamelyiket.

Ugyanakkor a mostani megközelítésben az a legfontosabb, hogy a szakértő úgy küldi az ajánlást, hogy nem tudja, kinek küldi azt.

2.4. Ajánlás küldése



Ha már a szakértő (pl. orvos) elküldte az ajánlást a felhasználónak (fogyasztó), akkor az elfogadhatja az ajánlást és elkezdheti az betartani. Ehhez engedélyt adhat a szakértőnek, hogy követhesse, hogyan változik a felhasználó egészségügyi állapota. Ehhez engedélyt kell adnia a saját állapot naplójának megtekintésére.

3. Fordított névvállalás

A fordított névvállaláson a következőt értjük: A szolgáltatást igénylő nem teszi közé a nevét, ugyanakkor az igény adatai elérhetőek. Ezek alapján az ajánlatót tévők ajánlatot küldhetnek az igényélőnek a rendszeren keresztül, a nélkül, hogy ismernénk, kinek adják az ajánlatot. Ez fordítottja a szokásos anonim tendereknek, ahol az ajánlatkérő ismert, az ajánlatok viszont név nélküliek.

A fordított névvállalásnál a következőket kell figyelembe venni:

- Név nélküliség: A rendszeren keresztül az ajánlat adó ne tudhasson meg semmi többet az igénylőről, mint az igény leírása. Esetünkben ez azt jelenti, hogy a szakértők ne tudhassák meg a nevét és címét, illetve egyéb személyt azonosító adatokat azokról, akiknek ajánlást adnak.

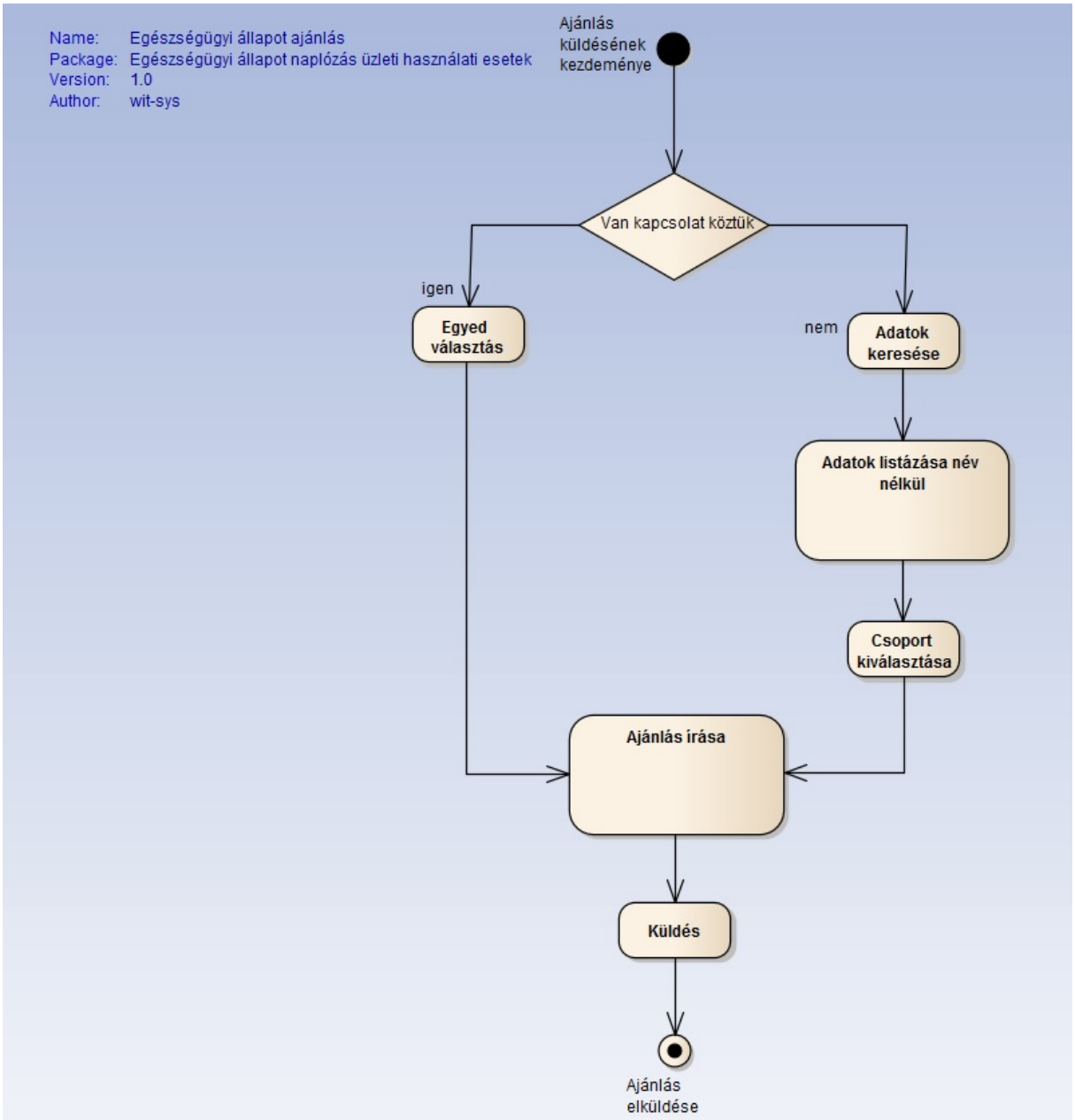
- Csoportos ajánlat küldés: Lehetővé kell tenni, hogy egy ajánlatadó (esetünkben az orvosok és egyéb szakértők) csoportos ajánlatot küldhessenek, azaz egy ajánlat több szolgáltatás igénylőhöz (esetünkben felhasználókhoz) jutathassanak el.
- Átlagolás elkerülése: A név nélküliség biztosításának egyik kézenfekvő lehetősége, hogy a rendszerben tárolt adatokat a felhasználóról átlagoljuk, hogy így elkerülhető legyen, hogy a kimagasló értékek alapján, kizárásos alapon, vagy a nagy hasonlóság alapján, rájöhessen a szakértő, hogy kinek az adatait látja. Ugyanakkor ez nem jó megoldás, mert az átlagostól eltérő értékek értékes információt tartalmazhatnak a szakértők számára.

Megoldási lehetőségek:

- Darabszám eltitkolása: Ne lehessen lekérdezni, hogy az adott feltétel hány felhasználóra érvényes, így a szakértők csak azt látják, hogy van vagy nincs a megadott feltételnek megfelelő felhasználó a rendszerben. Így elkerülhető, hogy a szakértő olyan speciális lekérdezéseket állítson össze, ami már csak egy felhasználóra igaz és így megtudjon a felhasználó titkos adatain kívül mindent az adott felhasználóról. Ez egy nagyon hasznos és egyszerűen kivitelezhető megoldás.
- Hasonló, nem létező személyek generálása: Ha van olyan adatfajta, amiből csak nagyon kevés van, pl. Kik ettek Az arany kakasban, akkor a rendszer generál néhány nem létező személyt, akik szintén rendelkeznek ezekkel az adatokkal, úgy hogy az így generált adatok ne befolyásolják az átlagot. Ezzel szintén azt tudjuk elérni, hogy a szakértő egy személyre tudja leszűkíteni a találati listát. Ez sokkal nehézkesebb megoldás, mint az előző, ugyanakkor ez azt is biztosítja, hogy ha a szakértő képesek lennék a profilok mögé látni, akkor is nagyon nehéz legyen megállapítaniuk, ki valós személy.
- Klaszterekbe rendezés: Az anonimitás eléréséhez profilok alkalmazunk. A felhasználók adatait egy neuronháló klaszterbe szervezi. Az egy klaszterben lévő adatok átlaga egy profilt ad. A tanácsadók a profil adatait látják.

3.1. Ajánlat küldése aktivitási diagram

A rendszer tervezésének mostani szakaszában a következő folyamatábrát (UML terminológia szerint aktivitási diagramot) használjuk:



Ezen az ábrán az a folyamat látható, amikor az orvos ajánlást ad azoknak a csoportoknak, akiknek életmódját kockázatosnak látja. A csoport kiválasztása a mi szempontunkból felesleges, de a szoftver többi része miatt fontos.

3.2. Riasztás, avagy a szakértők és a felhasználók szétválasztása

Rendszerszervezésből jól ismert elv, hogy amit csak szét lehet választani, azt válasszuk is szét (separation of concerns). Ezt az elvet itt is jó lenne alkalmazni a szakértők és a felhasználók szétválasztására, hiszen éppen az a cél, hogy a szakértők a rendszer segítségével ne tudják összekötni az érzékeny egészségügyi és életviteli adatokat a személyes adatokkal.

A megoldás a riasztások bevezetésében látszik. A riasztás egy olyan automatikus üzenet, amit a felhasználó kap, ha az adatai megfelelnek egy előre összeállított feltételnek. És itt van a lényeg. A feltételrendszert előre kell összeállítani a felhasználók konkrét adatainak ismerete nélkül. Ez azt jelenti, hogy a szakértő úgy kell, hogy összeállítson egy riasztást, hogy nem lát felhasználói adatokat,

hanem az általa jól ismert határértéket állítja be, ami felett egy adott betegség kialakulásának kockázata már magas. Maga a riasztás továbbra is egy ajánlás. Ugyanakkor ezt az ajánlást a rendszer küldi a felhasználónak, nem a szakértő, így a szakértők és a felhasználók szétválasztása megtörtént.

További ölet a riasztások összeállítására egy kategórialétrehozó felület. A kategóriák haszna, hogy több adatból jönnek létre, így figyelembe lehet venni pl. a nemet, az életkort a kialakításukban, és olyan intuitív fogalmak leírását teszik lehetővé, mint a túlsúlyos.

Kategória a például jól ismert BMI, azaz a testsúly index, ami figyelembe veszi a nemet és a testmagasságot és a segítségével megadható a túlsúlyosság értékhatára.

Mivel a szakértők által használt források, pl. orvosi könyvek, edzés tervek, nem adnak meg pontos érték határokat az egyes betegségek kockázat növelő körülményeinek leírására, csak ilyen magas absztrakciós szinten lévő fogalmakat használnak, mint „keveset mozog”, „ülő munkát végez”, „zsírosan táplálkozik”, ezért a kategóriák definiálásának lehetősége nagyon fontos, mert a rendszer, szükség szerűen, csak konkrét értékek kezelésére képes.

4. Biztonsági protokollok

Ha egyszer már kialakult a szakértők és a felhasználók kommunikációját leíró protokoll, akkor nagyon fontos, hogy erről a protokollról beláthassuk, hogy biztonságos.

A biztonsági protokollok olyan rövid programok, amelyeknek céljuk a biztonságos kommunikáció. Minden nap használjuk ezeket, gyakran a nélkül, hogy tudatában lennénk ennek. A biztonsági protokollok fontos kérdése a bizalom. Kiben bízhatok meg?

A mi esetünkben az a kérdés, hogy a rendszer felhasználója, megbízhat-e a rendszerünkben, nyugodt szívvel megadhatja-e, hogy mikor mit evett, mikor fájt a feje. Nem fog-e ez a sok kényes információ avatatlan kezekbe kerülni.

Még speciálisabban, az a kérdés, hogy a felhasználó biztos lehet-e abban, hogy az ő személyét beazonosítani képes adatok a rendszeren belül a szakértők kezébe nem kerülhet, azokat nem köthetik össze az ő egészségügyi és életmód adataival.

Szerencsére ez lehetséges, mert a biztonsági protokollok formálisan vizsgálhatók. Nem is olyan régi eredmény, hogy a biztonsági protokollok helyessége elméletileg eldönthető, ha a session-ök száma limitált [Amadio, Charatonik, 2002]. Azaz, érdemes a protokollokat elméletileg vizsgálni!

5. Biztonsági protokoll modellezésére alkalmas leíró nyelvek

A biztonsági protokollok vizsgálatának első lépése, hogy formálisan leírjuk azokat. Az alábbi felsorolás a leíró nyelvek fejlődését mutatja be az évszámok megadásával.

1983 – Dolev-Yao modell [Dolev, Yao, 1983]. Az első leíró nyelv. Itt a protokolloknál szokásos elnevezéseket lehet használni, így egy magas szintű és intuitív leírás adható, de a megértésen túl mást nem szolgál, mert bizonyítási rendszer nem kapcsolódik hozzá.

1989 – BAN Logic [BAN, 1989]. Az első logika, ami protokollok leírására lett kifejlesztve. Mai napig használatos, mert kiforrott leíró eszköz, ugyanakkor nagyobb protokollok vizsgálatára nem ajánlott.

1997 – Spi calculus [Abadi, Gordon, 1997]. Ennek a leíró nyelvnek az újítása, hogy nem csak a protokoll, hanem a protokoll elleni támadás is formálisan leírható, így a segítségével be lehet látni, hogy az adott protokoll az adott támadás ellen biztosítva van, vagy sem.

1997 – Model Checking Methods for Security Protocols (http://seclab.stanford.edu/pcl/mc/mc.html). Mivel a biztonsági protokollok leírhatók véges állapot gépekkel, ezért fejlett modell ellenőrző rendszerek segítségével is vizsgálhatók. Mivel ez a terület nagyon gyorsan fejlődik, és mivel nagyon hatékony és ingyenes modell ellenőrző rendszerek léteznek, ezért ez az egyik legkézenfekvőbb út. Továbbá a modell ellenőrző rendszerek nem csak minőségi kérdések, hanem mennyiségi kérdések megválaszolására is alkalmasak, így akár azt is megtudhatjuk egy protokollról, hogy mennyi az átlagos futási ideje.

2003 – Automated Validation of Internet Security Protocols and Applications (http://www.avispa-project.org/). Ez egy EU által támogatott kutató program. Az eddigi legjobb tulajdonságokat egyesíti és van eszköz támogatottsága is.

2007 – Protocol Composition Logic [ADMR, 2007]. Manapság ez tekinthető a legjobb biztonság protokoll leíró nyelvnek. Mivel a kompozíciót erősen támogatja, ezért a protokoll leírását a részeinek leírására redukálhatjuk, amelyeknek bizonyítása kézzel is lehetséges. Azután az egyes részek helyességéből adódik az egész helyessége. Így nagy, bonyolult protokollok vizsgálatára is alkalmas. Ugyanakkor eszköz nincs támogatottsága.

2011 – Automated VAlidatioN of Trust and Security of Service-oriented Architectures (http://www.avantssar.eu/). Az előző projekt folytatása, szintén EU által támogatott. Jelenleg nem látható, milyen eredményei lesznek.

5.1. Két fő irány: Modell ellenőrzés, Protokoll logikák

Látható, hogy a protokollok leírására két fő lehetőség van. Vagy egy speciális protokoll logikát használunk, vagy egy jóval általánosabb modell ellenőrző rendszert. Mindkét iránynak megvan az előnye és a hátránya. Nézzük ezeket egy kicsit részletesebben.

Modell ellenőrzés:

- Csak véges rendszerek leírására és ellenőrzésére alkalmas.
- Teljesen automatizált.
- Ha lehetséges egy támadás, akkor automatikusa megadja, hogy milyen körülmények között sikeres a támadás.
- A leírás nem feltétlenül könnyen megérthető biztonsági protokoll szakértők szemszögéből.

Protokoll logikák:

- Végtelen állapottér esetén is alkalmazható.
- Nem mindig teljesen automatizálható.
- Ha lehetséges egy támadás, akkor nem adja meg automatikusa, hogy milyen körülmények között sikeres a támadás.
- A leírás sokkal könnyebben érthető, mint modell ellenőrző rendszerek esetén.

Tehát, ha véges állapottérrel reprezentálható a protokoll, és nem fontos, hogy intuitív legyen a leírás, de az automatikus ellenőrzés fontos, akkor használjuk modell ellenőrző rendszereket, mint pl. a PRISM (http://www.prismmodelchecker.org/).

6. Összefoglaló

Ebben a cikkben a PHA projekt egyik fontos tulajdonságával foglalkoztunk, a felhasználók személyét azonosító adatok elzárásával a többi felhasználó, jelesül a szakértők elől. Ennek módszere a fordított névvállalás, ami alatt az értjük, hogy a szakértők név nélkül ismerhetik meg a felhasználók adatait, ami alapján életmódjukra vonatkozó ajánlásokat tehetnek számukra, úgy hogy ők vállalják nevüket, ezzel téve az ajánlást hitelessé.

A probléma megoldására két javaslatot tettünk. Először is érdemes eltitkolni a szakértők elől, hogy egy feltétel rendszernek hány felhasználó felel meg, így elkerülhető, hogy egy túl speciális feltételrendszer beállításával egy adott felhasználóról minden publikus adatot megtudhassanak. Ez azért fontos, mert a sok adatból kikövetkeztethetik, hogy kinek az adatát látják.

A másik javaslat a szakértők és a felhasználók teljes szétválasztása a riasztások bevetésével. A riasztás kialakításához a szakértőnek nem kell a felhasználók semmilyen adatához hozzáférnie, csak az általuk jól ismert kockázatos életmódra utaló határértékeket beállítani, hogy az ezeket elérő felhasználók automatikusan megkapják a riasztást, ami egy életmód ajánlást tartalmaz.

A cikkben foglalkozunk a biztonsági protokollokat leíró eszközökkel is. Ezeket áttekintjük, de a fordított névvállalás leírását egyelőre nem tudtuk elkészíteni. Ezt egy következő cikkben kívánjuk részletesen megtenni.

Irodalomjegyzék

[Abadi, Gordon, 1997] Martin Abadi, Andrew D. Gordon: A calculus for cryptographic protocols: The spi calculus, 4th ACM Conference on Computer and Communications Security, pages 36—47, 1997.

[ADMR, 2007] Anupam Datta , Ante Derek , John C. Mitchell , Arnab Roy: Protocol Composition Logic (PCL), Electronic Notes in Theoretical Computer Science Volume 172, Pages 311–358, 2007.

[Amadio, Charatonik, 2002] Amadio and W. Charatonik. On name generation and set-based analysis in the Dolev-Yao model. In Proc. of the 13th International Conference on Concurrency Theory (CONCUR'02), LNCS, pages 499–514. Springer Verlag, 2002.

[BAN, 1989] M. Burrows, M. Abadi, and R. Needham. A logic of authentication. ACM Operating Systems Review , 23(5):1{13, december 1989.

[Dolev, Yao, 1983] Dolev, D.; Yao, A. C.: "On the security of public key protocols", IEEE trans. on Information Theory, IT-29: 198–208, 1983.

[eFilter1] Biró Csaba, Geda Gábor: Betegségek, allergiák, étel érzékenységek leírása alkalmas XML séma tervezése, Networkshop 2011 konferencia, 13 oldal, 2011.

[eFilter2] Biró Csaba, Juhász Tibor: A hátizsák probléma továbbfejlesztése az egészségügyi profil figyelembevételével diéta tanácsadáshoz, AgriaMédia 2011 konferencia, 387-393, 2011.

[eFilter3] Király Roland: Hiteles adatgyűjtés az eFilter projektben - azonosítási módszerek elemzése, Informatika a Felsőoktatásban 2011 konferencia, 2011.

[eFilter4] Kovásznai Gergely: Developing an Expert System for Diet Recommendation, Conference Proceedings of SACI 2011, 505-509, 2011.

[eFilter5] Kusper Gábor, Márien Szabolcs: Élelmiszer adatbázis szűrése mennyiségi megszorítások alapján logaritmikus indexeléssel, AIK 2011 konferencia, elfogadás alatt.

[eFilter6] Kusper Gábor, Márien Szabolcs, Kovács Emőd, Kovács László: Valós időben választ adó egészségügyi profil, mint több dimenziós megszorítás mátrix, alapján élelmiszert szűrő domain specifikus algoritmus, Networkshop 2011 konferencia, 24 oldal, 2011.

[eFilter7] Kusper Gábor, Kovács Emőd, Márien Szabolcs, Kusper Krisztián, Scheffer Imre, Kiss Balázs, Kovács Péter és Winkler Ernő: Innovatív megoldások az eFilter projektben, IF2011, Informatika a Felsőoktatásban 2011, CD-kiadvány, 22 oldal, 2011.