híradástechnika

VOLUME LIX.

2004/8 Augusztus

Az IPv6 megjelenése

Protokolltervezés

Az információátvitel biztonsága

A Hírközlési és Informatikai Tudományos Egyesület folyóirata

Tartalom

Címlap: Az első hazai gyorsítóberendezés Sopronból indult, a KFKI-ban dolgozott, végül az ELTE adott neki kegyeleti helyet

A szoftverek szerepe

lajtha.gyorgy@ln.matav.hu

Már közel 20 éve annak, hogy megjelentek az első tároltprogramvezérlésű központok. Ezek fejlesztése során a korábbiakhoz képest jelentősen megnőtt a berendezésekben alkalmazott szoftver mennyisége. Az árképzésnél még a fejlesztő mérnökök is, de különösképpen az üzemeltetők, beruházók meglepve tapasztalták, hogy a berendezések árának nagyobb hányada a szoftverköltség. Ez a tendencia tovább erősödött a csomagkapcsolás megjelenésével és az IP alapú hálózatok elterjedésével.

A szoftver meghatározó szerepe a berendezésekben új szemlélet kialakítását követelte meg és megváltozott a gazdasági, beruházási politika. A távközlés valamennyi szereplőjének ennek következtében új stratégiára volt szüksége saját feladata végrehajtásánál. Elsősorban a fejlesztők, tervezők dolgoztak ki új struktúrákat, új átviteli és kapcsolási rendszereket. Itt meghatározó volt, hogy a változások elsősorban a szoftvert befolyásolják és annak módosításával kell az új szolgáltatásokat illeszteni az előzőkhöz. A hálózat struktúrájának átalakításakor és a felhasználók számának növekedése során szintén elsősorban a szoftvert igyekeztek alkalmassá tenni az igények kielégítésére. A hardware, a vas több generációt átélt. Új programok változatlan eszközökre voltak telepíthetők.

A gyártók tudomásul vették, hogy a hardver és a szoftver várható élettartama nem azonos. Azonos hardverrel kellett a változó igényeket néhány évig, sőt esetleg egy évtizedig kiszolgálni, mindig újabb szoftververziókkal. Lassan kialakultak azok a berendezések és ezen berendezésekből azon hálózatok, melyek tökéletesen megfeleltek a szellemi értéket hordozó szoftver nagyobb rugalmasságának. Vannak azonban olyan pillanatok, amikor egy kialakult programrendszer is átalakításra szorul.

Bár a szoftver fejlesztés költségei nem voltak elhanyagolhatók, a teljes távközlési rendszer jelenértéke csökkent. Ez a tendencia a versennyel együtt tarifacsökkenést eredményezett, ami igen meggyorsította a távközlés terjedését, használatát.

A tömeges igénybevételt segítette a csomagkapcsolás bevezetése is. Az irányítási feladatokat átvették a korábban csak speciális célokra használt Internet protokollok. Az utóbbi években egyre gyakrabban hallunk arról, hogy a hálózatot vezérlő Internet protokoll nem képes a felhasználók számának növekedését követni és az új címeket beépíteni a rendszerbe. Nehézséget jelentett ennél a módszernél némely szolgáltatás, különösen a többes-adás és a titkosítás. Kidolgozták ezért a következő bővített protokollt az IPv6-ot. Ennek bevezetése a jelen, illetve a közeljövő problémája. E számunk első két szakmai cikke ezzel a témával foglalkozik, az új generációs IPv6 bevezetésének stratégiáját vázolva.

A protokollok tervezése és vizsgálata szintén egyre fontosabb a hálózatok megbízható működése érdekében. Míg a mechanikai eszközök vizsgálatára évtizedek óta kialakult, jól működő eljárások és műszerek állnak rendelkezésre, addig a protokollok ellenőrzése kezdetben csak próbálkozással volt lehetséges. Viselkedésüket szinte minden várható körülményre egy-egy kísérlettel igyekeztek ellenőrizni. Ez rengeteg időt igényelt és mégsem garantálta, hogy a rendszer a jövőben hibátlanul fog működni. A kutatók ezért igyekeznek megbízható vizsgálati módszert kialakítani. E számunk második blokkja ezen a területen elért eredményeket mutat be.

A protokollfejlesztéssel párhuzamosan a rendszer biztonságára is ügyelni kellett. Ezt a sokoldalú feladatot járták körül következő szerzőink, felvetve a szenzorhálózatok és az anonimitás kérdéseit.

A negyedik csoport két cikkét nem egy kutatási irány, hanem egy személy emléke köti össze. Simonyi Károly több mérnökgeneráció emlékezetében az elméleti villamosságtan című tárgy élvezetes előadásával maradt meg. Már ő gondolt arra, hogy a relativitáselméletet és a Maxwell-egyenleteket összekapcsolja. Ugyanakkor, mint kutató az atomfizikai kutatások hazai megalapozása és eszközhátterének megteremtése érdekében ért el óriási eredményeket. Kiemelkedő volt ezek közül az első hazai Van de Graaff-generátor megépítése, mely ma már műemlék.

A jelen és a múlt tudományos eredményeiről, azok gyakorlati hasznosulásáról korábbi számainkban is gyakran megemlékeztünk. Ezekből is látható, hogy sikereket akkor lehet elérni, ha az újdonságok kellő időben jelennek meg és gyorsan megvalósulnak.

Dr. Lajtha György

Föld! Föld?
Óvatos piaci helyzetkép az infokommunikációs iparról

BŐGEL GYÖRGY

A KFKI Számítástechnikai Rt. stratégiai tanácsadója,
a Közép-Európai Egyetem Üzleti Iskolája tanári karának tagja, a Debreceni Egyetem docense
gybogel@kfki.com

A hajó megtépázott vitorlákkal úszik a még mindig hevesen hullámzó tengeren. A matrózok vitorlát foltoznak, új kormánylapátot ácsolnak, a viharban szerzett sebeiket ápolják. Az árbóckosárban elcsigázott matróz ül, csíkos trikóban, szeme a horizonton. Először csak egy elmosódott foltot lát – nem mer jelezni, hátha csak egy felhő vagy ködfolt lebeg a láthatáron. Ahogy a hajó halad, a kép tisztulni kezd: igen, ezek sziklák, előttük sárga föveny. Most már felkiált: Föld! Föld!

Föld? Teszi fel magában a kérdést a kabinjában a kapitány. Földnek éppenséggel föld. De vajon milyen föld? A világnak erről a tájáról még nem készült térkép. Az a folt a távolban egy új kontinens, egy kicsiny sziget vagy egy veszélyes zátony egyaránt lehet. Mire kell számítani, mire kell készülni? Lesz-e szélmentes kikötő, élelem, víz, le lehet-e telepedni, termékeny-e a talaj? Jut hely mindenkinek, vagy barátságtalan bennszülöttekkel kell hadakozni végeláthatatlanul?

Vihar után

Az infokommunikációs ipar néhány nehéz évet tudhat maga mögött. A kilencvenes évek hosszú fellendülése után a termékei és a szolgáltatásai iránti kereslet visszaesett. A tőzsdei árfolyamok zuhanni kezdtek, a befektetők érdeklődése megcsappant, az újságok címlapjáról lekerültek az iparág kapitányai. A vihar gyorsan tört ki, és amerre elvonult, léket kapott vállalatokat, letépett vitorlákat, elbizonytalanodott embereket hagyott maga után.

A régi iparágakat aligha lepte meg a *recesszió:* ők már többször átéltek ilyesmit. Az informatika viszont fiatal iparág, sokaknak ezért ez volt az első komolyabb megpróbáltatás az életükben. A fiatalság szerencsére tanulékonyságot is jelent. A vihart túlélő cégek nagyjából azt tették, amit hasonló helyzetekben az idősebb tengeri medvék tenni szoktak: a kereslethez vágták vissza a kapacitásaikat, csökkentették a költségeiket, bezárták, átszervezték a veszteséges részlegeiket, profilt tisztítottak, csökkentették az eladósodottságukat, többet törődtek a hatékonysággal, a termelékenységgel. A gazdagabbak pedig még körül is néztek a megtépázott piacon azt kutatva, hogy mit lehetne könnyen és olcsón felvásárolni, miként lehetne nagyobb, erősebb hajót építeni. A hajóraj átrendeződött, a gyengébbek elmerültek, egypár illúzió elveszett, néhány nagy hazugság lelepleződött, mindenki egy kicsit öregebb és tapasztaltabb lett. Mindezek régi jelenségek és receptek.

Ma úgy tűnik, a vihar elvonult. A távolban felbukkant a föld, a vihar idején bevont vagy letépett vitorlákat újra felhúzzák. A vásznakat friss, barátságos szelek dagasztják.

Az infokommunikációs piac ismét növekszik, szereplői közül sokan jobban érzik magukat. A tőzsde optimista, az árfolyamok emelkednek. A lakosság élénk érdeklődést mutat a digitális ipar termékei iránt, nagyobbra cseréli a számítógépét, noteszgéppel sétál WI-FI kávéházat keresve, digitális fényképezőgépet vásárol, modernebbre cseréli a mobiltelefonját, belekóstol a harmadik generációs szolgáltatásokba, széles sávon száguldozik az interneten, bekábelezi a lakást vagy éppenséggel rádiós hálózatot épít, hogy egyetlen családtag se maradjon ki a jóból.

A kisebb vállalatok örömmel tapasztalják, hogy a csökkenő áraknak és a nekik kitalált termékeknek köszönhetően rájuk is tárt karokkal vár az elektronikus gazdaság. A nagyvállalatok az új beruházásokkal egyelőre óvatosak, de mivel a már megépített rendszereiket ki akarják használni, sok integrációs feladatot adnak a szolgáltatóknak. Az elektronikus kereskedelem mutatószámai emelkednek, a nagy ugrások sokszor még a komoly elemzőket is meglepik.

Ismét vannak technikai újdonságok, amelyek megmozgatják az emberek fantáziáját: itt van a „grid” és a „utility computing”, a „software on demand”, a „web services”, itt vannak a hol egymást kiegészítő, hol egymással versengő rádiós technológiák, itt a „smart dust”, vagyis a szimatoló, fülelő, jeleket adó „porszemek”, melyek egyszer talán felváltják a vonalkódokat. Hatalmas keresletet támaszt az iparág termékei iránt Kína, a világgazdasági fellendülés fontos motorja.

„E-biz Strikes Again!” – kürtöli világgá a Business Week egyik májusi címlapja, de még a konzervatív és visszafogott londoni The Economist is optimistán nyilatkozik a digitális ipar jövőjéről. Három fagyos év után 2004-ben négy új internetes vállalkozás jelent meg az amerikai tőzsdén, további 23 pedig már összerakta a szükséges dokumentumokat és bebocsátásra vár. És ami még meglepőbb: ebből a 27 cégből 20 nyereséges – 1998 és 2000 között ez csak a friss tőzsdei vállalkozások négy százalékára volt igaz. Ugyanez a 27 cég 2003-ban 56%-os bevétel- és 490%-os(!) nettó nyereségnövekedést produkált az előző évhez képest.

Föld! Föld! – kiáltja az őrszem az árbockosárból, és tényleg látszanak hegyek, kirajzolódik a parti föveny. Föld? Morfondírozik a kapitány. De milyen föld? Merre megy a hajó?

A piaci jelek szerint az infokommunikációs ipar ismét felszálló ágban van, ami mindenképpen jó hír. A recessziós éveket akár kis balesetnek, átmeneti megtorpanásnak is nevezhetnénk, ami után a dolgok ismét a normális kerékvágásba zökkennek vissza. A kapitalista gazdaság világéletében ciklikus volt, a fellendülésekre hanyatlások következtek, majd újból fellendülés. Egyszerűnek látszik kimondani: a recessziós időszakokat át kell vészelni, ki kell dobálni a homokzsákokat, rendezni kell a sorokat, és aztán, ha a vihar elvonult, fel a vitorlát és mehet minden tovább.

Félő azonban, hogy nem így áll a helyzet. Az iparág történetében az új évezred elején nagy valószínűséggel lezárult egy korszak, az új idők pedig új stratégiákat követelnek. Hajózni lehet, de másképpen, mint a kilencvenes évek „aranykorában”.

Hullámok hátán: a Carlota Perez-modell

Az átalakulások, a ciklikus mozgások, visszatérő minták és tartós trendek megértéséhez *többféle modell* áll rendelkezésünkre. Számunkra most azok a fontosak, melyek az infokommunikációs ipar és a felhasználók kapcsolatáról, vagyis a kereslet és a kínálat viszonyáról mondanak valamit, mennyiségi és tartalmi-minőségi jellemzőkre egyaránt kitérve. Az ilyen modellek egy része makrogazdasági jellegű, más része viszont konkrétabb, egy-egy termék- vagy szolgáltatásfajta piaci mozgásának megértését szolgálja.

A modellek között vannak olyanok, amelyek a nagy *technikai innovációk* (gőzgép, vasút, elektromosság stb.) gazdasági és társadalmi hatásának kibontakozásában keresik a szabályosságokat, a visszatérő mintákat.

Egy manapság gyakran emlegetett kutató, a venezuelai Carlota Perez [8] szerint az innovációs hullámok két nagy korszakra bonthatók, nevezetesen az installáció és az összerendeződés periódusára *(1. ábra)*.

Az *installáció* szakaszában – ahogy a neve is mutatja – kiépül az új technológia által képviselt infrastruktúra. Ha konkrét dolgokról beszélünk: megépül a vasúthálózat, elektromos motorokkal szerelik fel az üzemeket; autógyárak, új országutak mentén benzinkút- és szervizhálózatok nőnek ki gombaként a földből.

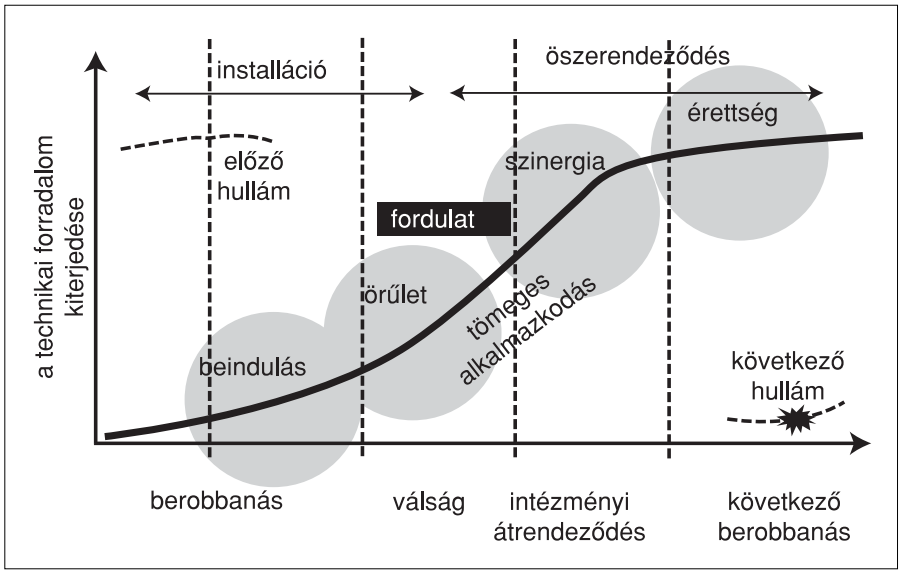
Az installáció korszaka két szakaszra választható szét. Az elsőben az új technológia lappang, keresi a helyét, az általa kínált lehetőségek még nem világosak. A másodikban „berobban” általános érdeklődést kelt: a vállalkozók meglátják benne a „nagy lehetőséget”, a befektetők megnyitják előtte a pénztárcájukat, nagy az izgalom körülötte, a kereslet gyakran meghaladja a kínálatot. Ebben a szakaszban illúziók, csalfa remények is megjelennek, a léggömböknek azonban egy idő után szükségszerűen ki kell pukkanniuk. Az installáció kora ezért általában válsággal végződik.

Ez a válság szerencsére csak átmeneti, és nem az innovációs hullám kifulladását jelzi. Az installáció szakaszának második periódusában irracionális viselkedési formák is megjelennek: csodavárás az új technológiától, eltúlzott, meggondolatlan beruházások, tőzsdei szerencsejáték. Az „őrületnek” azonban megvan a maga fontos funkciója: elősegíti az új technológiára épülő infrastruktúra gyors felépítését. Rohamtempóban fektetik le a síneket, cserélik a gépeket, építik az utakat, húzzák ki a kábeleket, gründolják a szolgáltató vállalkozásokat. A válság bizonyos értelemben rendet csinál: a gyengébbek kiszelektálódnak, a túlméretezett kapacitásokat visszavágják, a tőzsde megnyugszik, az emberek kijózanodnak.

Az installációs szakasznak az új technológiát építő-terjesztő tipikus vállalkozása tág teret, nyílt, friss vadászmezőket lát maga körül, ezért a növekedésre koncentrál, tőkét halmoz föl, igyekszik lecsapni a szűkös erőforrásokra. Eladni akar, kevésbé törődik azzal, hogy mire is használják azt, amit tőle vettek, hiszen már itt is van a következő éhes ügyfél. Mivel az érdeklődés nagy, tőkét könnyen lehet szerezni, a vállalkozások gyorsan szaporodnak.

Az *összerendeződés* szakaszában az új infrastruktúra nagyrészt már kiépült. Az emberek, a vállalatok, a különböző intézmények egyre nagyobb rutinnal használják a technikai újításokat. Egy idő után már egyszerűen természetesnek veszik az új infrastruktúra jelenlétét, azt például, hogy autóba lehet szállni, fel lehet hívni a nagymamát egy másik városban, vagy áram van a konnektorban.

1. ábra Carlota Perez modellje a technikai innovációs ciklusok fázisairól



Természetesnek veszik; csak akkor lepődnek meg néha, amikor a rendszer csődöt mond: áramszünet van, bedugul a légiforgalom, nincs vonal a telefonban. Az innovációs hullám most fejti ki igazán gazdaság- és társadalomátalakító hatását, valamivel kevésbé hangosan, de ugyanakkor mélyebben, mint a megelőző időszakban. Megállíthatatlanul hatol be mindenhová, a gyárakba, az irodákba, az otthonokba, a kultúrába, az államba és a politikába.

Megszületik az a vállalat, amelyik a technikai innovációk okos alkalmazására alapozza a versenyképességét, megszületik az új infrastruktúrát használó fogyasztói életmód, kialakulnak az új eljárások és szokások. Már nem arról van szó, hogy például új vasútvonalakat kell villámgyorsan lefektetni, hanem arról, miként lehet a vasútból ésszerűen működő, szabványos, egységes rendszert csinálni, hogyan lehet a termelésben és a kereskedelemben kihasználni a vasút jelenlétét, hova kell telepíteni a bányákat és az üzemeket, mekkora földrajzi körből lehet munkaerőt toborozni. Nem az a kérdés, hogyan lehetne még több kábelt fektetni a földbe és a tengerekbe, hanem hogy mivel lehet rávenni az embereket a telefonjuk – legyen az hagyományos vagy mobil – gyakoribb használatára, miként könnyítheti meg a tanulást, a tájékozódást vagy a hivatalos ügyek intézését a technológia.

A technikának, az új infrastruktúrának, a gazdaságnak és a társadalomnak ez az összerendeződése hosszabb folyamat. Kevésbé zajos az előző korszak tarka és lelkes világánál, következményei viszont tartósabbak, megalapozottabbak. Történik ez mindaddig, amíg az adott technikai innovációs hullám ki nem fullad, és át nem veszi a helyét valami más.

E kor tipikus vállalata konszolidáltabb piacon dolgozik, hiszen az installációs szakasz végén bekövetkező válság megtizedeli, átrendezi a sorokat. A növekedés lassul, a gründolási láz lelohad. A felhasználók, a vevők hamar felismerik, hogy most nekik áll a zászló. Óvatosak, gyanakvóak, zajos kampányokkal kevéssé lehet rájuk hatni. Építkeznek, egyre kreatívabban használják a technikát, de mindezt megfontoltan, költségeket és hasznokat mérlegelve teszik. Nem a technológia birtoklása, hanem a *használata* érdekli őket. Aki el akar adni nekik valamit, annak az alkalmazást kell segítenie, a hasznot kell garantálnia.

Vevői és eladói oldalon egyaránt a hatékonyság, a termelékenység a jelszó: a technika vevője hatékonyabb, versenyképesebb akar lenni, eladója pedig a konszolidálódó, beérő piacon csak akkor tud nyereségesen dolgozni, ha vigyáz a saját hatékonyságára, kordában tartja a költségeit. A kapcsolatokat meg kell becsülni, hiszen egy elvesztett vevő helyébe nagyon nehéz másikat szerezni.

Az eseményeket látva logikusan adódik a következtetés: az *infokommunikációs innovációs ciklus* a kilencvenes években átment a maga installációs korszakán, átélte a végén jelentkező válságot, és a hajó most az összerendeződés, az alkalmazkodás lassabban hömpölygő vizei felé tart. Az összerendeződés, az alkalmazkodás egyik legfontosabb „terméke”, az úgynevezett „integrált, valós idejű, kiterjesztett elektronikus vállalat” [1] ebben a második időszakban épül fel. Falait az alapként szolgáló, nagyrészt már kiépült infrastruktúrára rakják.

Az infokommunikációs technológia lépésről lépésre tölti ki a teret: először egyes tevékenységeket automatizálnak vele, utána teljes funkciókat, folyamatokat; ez után a szigetrendszerek integrációja következik, majd a több vállalatot átfogó ellátási láncoké [6]. A folyamat megállíthatatlannak tűnik, hosszú távú következményei kiszámíthatatlanok.

Az összerendeződés időszaka más stratégiákat, viselkedési módokat és módszereket kíván, mint az installációs korszaké. A hangsúlyok eltolódását, a stratégiák átalakulását jól példázza az *informatikai szolgáltatások* 520 milliárd dolláros éves forgalmú iparága, mely az elmúlt két évben csak 3-3%-os növekedést tudott felmutatni, szemben a kilencvenes évek fantasztikus tempójával.

Érzékelvén a piaci korlátokat, az iparág kisebb-nagyobb képviselői igen találékonynak mutatkoznak saját hatékonyságuk növelésében, költségeik csökkentésében, egyszersmind azt is bemutatva, mire képes a technológia. Olcsóbb munkaerőt keresnek, egyes tevékenységeiket olyan országokba telepítik, mint a lehetőségre gyorsan és ügyesen lecsapó India – a kiszervezésnek ezt az új hullámát a tevékenységek valós idejű kontrolljának technikai lehetősége élteti.

Az adatközpontokat működtető, sok száz ügyfelet kiszolgáló Inflow Inc. egyenként mintegy kétezer négyzetméteres, zümmögő gépekkel megrakott épületeiben egy időben két-három alkalmazott lézeng, egyébként minden automatizálva van. Az Accenture szoftverekből és szolgáltatásokból álló csomagokat rak össze különböző iparágak vállalatai számára, amiket aztán könnyebben és gyorsabban lehet testre szabni.

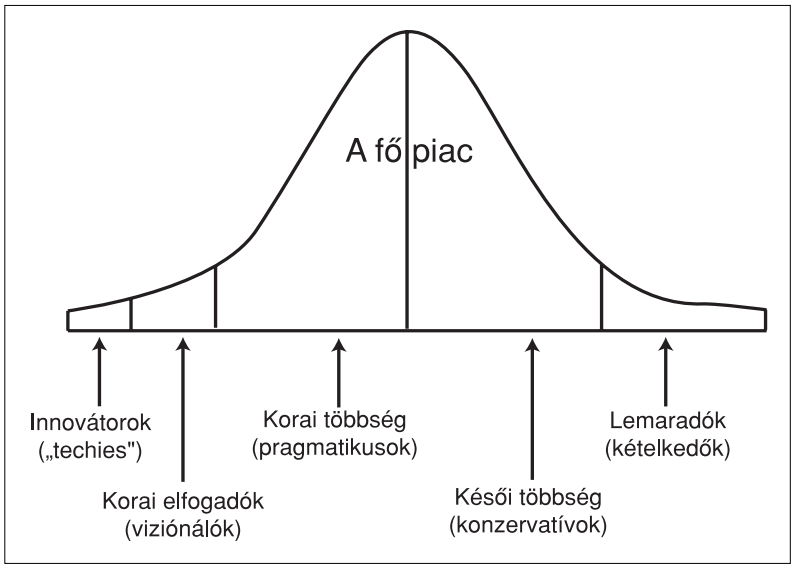
A Perez-modell hatása a világra

Az indiai informatikai ipar egyik koronagyémántja, a Wipro Technologies automatizálja a szoftverkészítési folyamatokat, és egy olyan programmal is büszkélkedhet, amely hat európai nyelvről 99%-os pontossággal fordít le szövegeket angolra.

Testvére, az Infosys Technologies web-szolgáltatási technológiákra építve szabályos könyvtárat rakott össze újrafelhasználható szoftvermodulokból: fejlesztői, amikor új feladatot kapnak, ezeket emelik le a polcokról és varrják össze őket az adott igénynek megfelelően.

A holland Getronics asztali gépek diagnózisát és támogatását automatizálja, aminek köszönhetően felére tudta csökkenteni a szükséges létszámot.

Az IBM-nél egy automatizálási akciónak köszönhetően ma órák alatt végeznek el szervereken olyan szoftverinstallálási munkákat, amikhez régebben öt-tíz nap is kellett.



2. ábra Geoffrey Moore elfogadási modellje

Bár a felhasznált technika modern, a példákból látható, hogy az alkalmazott módszerek közgazdasági szempontból régiek. Alkalmazottai termelékenységét ma is az tudja növelni, költségeit az tudja csökkenteni, aki ki tudja használni a sorozatnagyságból származó gazdasági előnyöket, szabványosítani tud, kész elemekből dolgozik, olcsóbb forrásokat használ, az egyik tevékenységénél szerzett előnyöket át tudja vinni egy másikra, gyorsan tanul, és így tovább. Ezeket az „alapmódszereket" használják a tömeggyártásra szakosodott kínai vállalatok és a nagy nevű amerikai és európai szolgáltató cégek egyaránt, természetesen másmás módon és tartalommal.

Az informatikai szolgáltatóknak úgy kell csökkenteniük a költségeiket, automatizálni a folyamataikat, hogy közben ne okozzanak problémát az ügyfeleiknek. Az érdekek egyébként találkoznak: a hatékonyabb, olcsóbb szolgáltató hatékonyabbá és olcsóbbá, végső soron versenyképesebbé tudja tenni az ügyfeleit. Ehhez nem elegendő a technikához érteni, hiszen a technika hasznosulása szervezeti, emberi változásokat feltételez.

Az oktatásban, az alkalmazottak fejlesztésénél ma nem az a kérdés, hogy „technika vagy üzlet": olyan emberekre van szükség, akik mindkét területen otthonosan mozognak. Az informatikai és távközlési cégek nem véletlenül igyekeznek a gyártás és az alapszolgáltatások biztosítása felől a magas szintű üzleti tanácsadás felé felfelé kapaszkodni az értékláncon

A célok, a szemléletmód, az elvárások változása a *szerződéses feltételekre* is kihat. A vállalati informatikai tanácsadásban sokáig a munkaórák alapján való számlázás volt divatban, a szolgáltatónak tehát az volt az érdeke, hogy egy munkához minél több emberre legyen szükség. Ma viszont más a helyzet. A neves piacelemző cég, az IDC szerint a mai tanácsadási szerződések mindössze 20%-ánál találhatunk hagyományos óradíjas megoldást, szemben a négy évvel ezelőtti 85%-kal. A díjazás ma általában teljesítmény-alapú. A szolgáltatók akkor kapják meg a pénzüket, ha sikerült növelni a hatékonyságot, ha a terveknek megfelelően nőnek a bevételek, csökkennek a költségek – hogy ehhez hány embernek kellett dolgoznia, az az ügyfelet nem érdekli.

A Carlota Perez által leírt modell szerint az innovációs folyamatban az őrület, a válság és a kijózanodás törvényszerűen követik egymást. A Gartner Group közismert hypegörbéje (a „hype" szót felhajtásnak, cirkusznak lehet fordítani) valami hasonlót üzen, csak nem makrogazdasági és társadalmi szinten, hanem az egyes termékek, termékcsaládok viszonylatában. A technikai újdonságok eleinte nagy feltűnést keltenek, amit gyártók, marketingesek, újságok, tanácsadók és konferenciaszervezők együttesen gerjesztenek. A csinnadratta után jön a törvényszerű kiábrándulás: hát ez mégsem az a csodagyógyszer, ami mindent meggyógyít. A kiábrándulást realizmus követi: tényleg nem csodagyógyszer, de ennél vagy annál a betegségnél valóban használ – így végül az újdonság megtalálja a maga helyét a világban. Carlota Perez korábban bemutatott modellje azt példázza, mi van akkor, ha egy innovációs forradalom hatására egy egész iparág indul el a hype-görbén.

A modell megújulása

Geoffrey Moore [5] modellje is tanulságos, és több jelenség magyarázatára alkalmas az infokommunikációs piacon. A Chasm Group alapítója és elnöke szerint a piac fokozatosan fogadja be az új technológiákat *(2. ábra).* Az egyes befogadó csoportok nemcsak méretükben, hanem igényeikben, elvárásaikban, szokásaikban is különböznek, más érdekli őket, másra „kattannak". Lehet, hogy valamivel meg tudod hódítani az egyiket, de nem kizárt, hogy ugyanazzal az eljárással kudarcot vallasz a következőnél. Aki ezt nem veszi figyelembe és nem vált időben, saját sikerei csapdájába esik.

Az újdonságokra először azok kis létszámú csoportja figyel fel, akiket maga a technológia érdekel, és nem az, hogy mit lehet vele csinálni. *Technokratákról,* lelkes, kíváncsi emberekről van szó, akiknek az asztala és a zsebe mindenféle ketyerékkel van tele, de ritkán vannak döntési pozícióban. Az újdonság addig érdekli őket, amíg meg nem ismerik, utána más felé fordulnak, várják a következőt. A befogadás folyamatában utánuk azok jönnek, akik az újdonságokban meglátják a nagy *stratégiai lehetőséget:* íme, itt van valami, amivel ki lehet törni a mezőnyből. ők már nem technikában, hanem üzletben gondolkodnak, merészek, mernek kockáztatni – de sajnos kevesen vannak.

Utánuk viszont népes csoport következik: a *korai többségnek* nevezett pragmatikusoké. ők nem forra-

dalmárok, nem szeretnek sokat kockáztatni. Megvárják, hogy a technológia beérjen, megjelenjenek a meggyőző pozitív referenciák. Inkább a szemüknek hisznek és nem a „nagy dumáknak”. Tanulni és befektetni hajlandók, de nem akarnak mindenáron elsők lenni: a préri tele van lelőtt pionírokkal – mondják. Az alkalmazástól nem várnak radikális változásokat, nagy ugrásokat: a rövidebb de biztosabb lépéseket kedvelik. Megtervezik a várható hasznot, óvatosan bánnak a költségekkel, jól megválogatják a szállítóikat. Sokan vannak, közülük kerülhetnek ki az első komoly referenciák.

Ezekre a referenciákra nagy szükség van, mert a korai többség megnyerése után a *késői többség* következik. Tipikus képviselői az érett, kiforrott, olcsó megoldásokat kedvelik. A nyilvánvaló előnyök, a könnyű alkalmazás győzik meg őket. Türelmesen megvárják, hogy az új technológia termékei és szolgáltatásai tömegcikkekké váljanak, és akkor indulnak bevásárolni. Tartanak a technológiától, kicsit talán félnek is tőle – nem akarják megtanulni, azt kívánják, hogy a technológia tanulja meg őket. Ha csalódnak, gyorsan visszavonulnak, és hosszabb időre elmehet a kedvük az egésztől. Az egyszerű, könnyen megérthető megoldásokat kedvelik, amikhez ragaszkodnak is, ha beválnak. Nem szeretnének a pincében generátort építeni: azt akarják, hogy az áram a konnektorból jöjjön, egyszerűen, olcsón és megbízhatóan.

Ha valaki még mindig hódítani akar, a késői többség után a *lemaradók* csoportját veheti célba. Nehéz dolga lesz: e kör tipikus képviselői mindent megkérdőjeleznek, előszeretettel hivatkoznak a kudarcokra (informatikai projekteknél nem nehéz ilyeneket találni). Felhívják a figyelmet az ígéretek és a valóság közötti sokszor valóban mély szakadékra. Gyakran kiáltják: „A király meztelen!” Idegesítő társaságról van szó, de meglátásaikból, kételyeikből és kérdéseikből sokat lehet tanulni.

Geoffrey Moore itt leírt elfogadási modellje mellé értelemszerűen odakívánkozik a marketing klasszikus *életciklus-modellje*. Egyszerű, gyakran megtapasztalt dolgot mond ki: egy termék vagy termékcsalád életében törvényszerűen követik egymást a bevezetés, a növekedés, az érettség és a hanyatlás fázisai. A két modell között nem nehéz felfedezni a párhuzamot. A bevezetett termék először a technika megszállottjainak érdeklődését kelti fel. A növekedést kezdetben az újító, kockáztató forradalmárok, majd a korai többség gerjesztik. A beérett, kiforrott piac a késői többségé, és végül talán még néhány lemaradót is meg lehet csípni.

Hol tart az infokommunikációs ipar mint egész ebben az elfogadási-életciklus modellben?

Több jel is arra mutat, hogy valahol a késői többség meghódításánál, az érettség fázisában. A garázskorszaknak, a műhelyekben összerakott, igen nehezen kezelhető gépek időszakának vége. A nagy stratégiai ugrásoknak is tanúi lehettünk: a forradalmárok egy része elhullott a viharban, többeket viszont valóban az élre repített a modern technológia: lásd például e-Bay, Dell, Amazon. A korai többség már kiépítette a maga belső infrastruktúráját, megvette és installálta a rendszereit, és, mint már leírtuk, a hatékonyság növelésén fáradozik. Most a késői többség meghódítása van soron, érett piachoz illő stratégiával és harcmodorral.

A tömegcikk lázadása

Az érett piacokra a *tömegcikkesedés* a jellemző, és pont ez kell a késői többségnek. Az infokommunikációs ipar válság utáni fellendülésének a tömegcikkesedés az alapja és a motorja, bár ez nem mindenkinek jó hír. E nélkül nem gerjedhettek volna be olyan keresleti motorok, mint például a lakosságé, a kisvállalkozásoké, a kiszervezett szolgáltatásoké.

Egy tömegcikkesedő terméknek szabványosnak, olcsónak, könnyen cserélhetőnek, könnyen megtanulhatónak, mindenfélével kompatibilisnek, összekapcsolhatónak kell lennie. Ezek pedig az infokommunikációs ipar alapvető haladási irányai és jelszavai.

Az iparág vezértermékе, az asztali számítógép jól példázza a tömegcikkesedés folyamatát. A PC viszonylag rövid idő alatt szabványos, könnyen installálható és használható termékké vált. A legtöbb ugyanazzal az operációs rendszerrel, ugyanazzal a mikroprocesszorral és ugyanazokkal a szoftverekkel van felszerelve. Mindennel összekapcsolhatók, legfőképpen egymással, ami az internet korában alapkövetelmény. Az áruk összezsugorodott, ma már nem tekinthető akadálynak. Áruházakban, plázákban kaphatók, csak be kell őket tenni a kosárba.

Ma ugyanez történik a szerverekkel, a munkaállomásokkal, a hálózati és tárolóeszközökkel: közülük is az olcsó, könnyen munkába állítható és bővíthető változatok a népszerűek. A közkedvelt internetes keresőt működtető, éppen a tőzsdére igyekvő Google cég a hardverállományát polcról levett gépekre, idősebb mikroprocesszorokra alapozza, és olcsó vagy éppenséggel ingyenes, nyitott forráskódú szoftvereket használ hozzájuk. Az újságokban sorra jelennek meg a hírek arról, hogy mennyi pénzt takarított meg például az Amazon vagy a General Electric alacsony árú informatikai tömegcikkek vásárlásával. A tömegcikkesedésre bazírozó Dell egyre-másra jelenteti meg alacsony árfekvésű termékváltozatait, miközben jóval kevesebbet költ kutatásra és fejlesztésre, mint például a Sun.

Az infokommunikációs piac tömegcikkesedésének egyik tipikus tünete a „túlfejlesztés”: a termék jóval többre képes, mint amit az átlagos fogyasztó vár tőle. Ez a magyarázata annak, hogy még olyan vezető cégek is, mint az említett Google, GEés Amazon lemondhatnak az innovációk állandó követéséről, és megelégedhetnek korábbi generációs változatokkal. A „túlfejlesztés” jelensége nélkül a Dell sem arathatna ekkora sikereket, költség-játszmává változtatva a korábbi technikai innovációs meccset.

A „utility computing” [7] és a „software on demand” koncepciója jól jelzi a tömegcikkesedés elképzelhető

irányait. Ezen – igen komoly vállalatok által is felvállalt – elképzelések szerint az informatikával ugyanaz fog történni, mint a vízzel és az árammal. Ma már senki sem működtet otthon vízművet és generátort ott, ahol víz van a csapban és áram a konnektorban.

A jövő felhasználója – hangzik az érvelés –, nem vásárol és telepít magának alkalmazásokat és rendszereket, hanem, ha szüksége van valamire (például egy ügyfélkapcsolat-menedzsment alkalmazásra), az interneten keresztül egyszerűen bérbe veszi azt egy szolgáltató „közműtől". Amikor használni kezdi, elindul a taxióra, amikor kikapcsolja, leáll. Nem kell törődnie karbantartással, fejlesztéssel – az a szolgáltató dolga. Egészen más gazdasági modell ez, mint a nagy beruházásokkal és fix költségekkel járó megvásárlás és installálás. A példák szaporodnak: a tőzsdei aspiráns Salesforce.com, a Taleo és a Right-Now Technologies az interneten kínálja vállalatoknak a szoftvereit, nagyjából 65 USD felhasználónkénti havi áron.

A tömegcikkesedés az infokommunikációs iparág képviselői számára több kellemetlenséggel jár: nehezebb a megkülönböztetés, éleződik a verseny, zsugorodnak a nyereséghányadok, ugyanakkora eredményért jóval többet kell dolgozni. (Vessünk egy pillantást a PC piacra: a mennyiségi fellendülés a gyártóknak és a kereskedőknek csak kis forgalomnövekedést hozott, hiszen az árak összeestek.) A folyamat ennek ellenére megállíthatatlan, önmagát gerjeszti, a logikája – szabványosítani kell, kész elemekből kell dolgozni, kompatíbilisnek kell lenni, óvakodni kell a monopóliumoktól, meg kell osztani, széles körben kell teríteni, ki kell szervezni, szabványos dolgokat szabványosan kell használni stb. – mélyen bele van égetve az infokommunikációs piac viselkedésébe vásárlói és eladói oldalon egyaránt [11].

A tömegcikkesedésre, a költségalapú versenyre adott tipikus válasz a *gyárszerű működés* szorgalmazása, amely sokfelé megfigyelhető gyártóknál és szolgáltatóknál egyaránt.

Vegyük például a szoftverfejlesztést. Az ötvenes évek végén jó ha húszezer szoftveres szakember volt a világban. A számuk ma becslések szerint mintegy kilencmillió. A szoftverírás valamikor, amikor még gépi kódban kellett dolgozni, nagyon bonyolult, nagyon nehéz tevékenység volt. Ma viszont számos eszköz könnyíti meg a fejlesztők munkáját. Ahogy a vállalatok szoftverigénye szabványosodik, ahogy a szoftver modularizálódik, a fejlesztése (legalábbis annak nagy része) úgy válik egyre inkább gyártási rutinná. Ebben a minőségében értelemszerűen oda igyekszik települni, ahol a világban ezt a gyártási tevékenységet olcsón és szervezetten le lehet bonyolítani. Erre a logikára épül az indiai vállalatok (lásd például Infosys, Wipro, Tata, Satyam) globális kiszolgálási modellje: a projektek során az ügyfélnél kell elvégezni a speciális helyismeretet kívánó munkákat, a „gyártást" viszont az olcsó és jól szervezett, modulokból építkező hátországba kell telepíteni…

Föld! Föld! – kiált ismét a matróz az árbockosárban. Most jó lenne sokkal pontosabb térképekkel rendelkezni, tisztábban látni, gondolja a gondolataiból kizökkentett kapitány. Aztán felmegy a hídra, és parancsokat kezd osztogatni. Majd meglátjuk, mi lesz. Navigare necesse est –, hajózni ma is kell...

Irodalom

[1] Bőgel György–Forgács András (2001):
Vége az ERP világnak?
Kontrolling, október

[2] Carr, N. (2004):
Does IT Matter?
Harvard Business School Press, Boston

[3] Gates, B. (1995):
The Road Ahead.
Viking, New York

[4] Kocsis Éva–Szabó Katalin (2000):
A posztmodern vállalat.
Oktatási Minisztérium, Budapest

[5] Moore, G. (2002):
Crossing the Chasm.
Harper Business, New York

[6] Murphy, T. (2002):
Achieving Business Value from Technology.
John Wiley & Sons, Hoboken, New Jersey

[7] Ördög Péter:
Utility computing – Informatikai közművek.
Diplomadolgozat, Debreceni Egyetem, Közgazdaságtudományi Kar, 2004.

[8] Perez, C. (2002):
Technological Revolutions and Financial Capital.
Edward Elgar, Cheltemham, U.K.

[9] Porter, M. (2001):
Strategy and the Internet.
Harvard Business Review, március-április

[10] Salamonné Huszty Anna (2000):
Jövőkép- és stratégiaalkotás.
Kossuth Könyvkiadó, Budapest

[11] Shapiro, C.–Varian, H. (1999):
Information Rules.
Harvard Business School Press, Boston

Az új generációs Internet alapjai

BENDE ZSÓFIA, CZIGÁNY ÁDÁM, NAGY KRISZTINA, LUKOVSZKI CSABA

Budapesti Műszaki és Gazdaságtudományi Egyetem, Távközlési és Médiainformatikai Tanszék
csaba.lukovszki@tmit.bme.hu

Kulcsszavak: címzés, címtár, mobilitás, biztonság, nemzetközi projektek

Információs társadalmunk egyik alappillére az Internet, melynek működését eddig az Internet Protokoll 4-es verziója szabályozta. Az Internet robbanásszerű elterjedése és az ennek következtében felmerülő új igények azonban szükségessé tették az IPv4 bővítését, fejlesztését. A megoldást egy új verzió jelenti. Cikkünk célja a protokoll 6-os verziójának bemutatása, a megalkotásához vezető igények ismertetése, a létrehozott új szolgáltatások alapjainak felvázolása. Végül, de nem utolsósorban feltárjuk az IPv6 jelenlegi helyzetét, nemzetközi és nemzeti szinten egyaránt.

A következő generációs Internet protokoll, más néven az IPv6 az Internet alapját jelentő 4-es verzió, vagyis az IPv4 (röviden IP) továbbfejlesztése. Az IPv4 1983-ban mutatkozott be és várakozáson felüli sikert tudhat magáénak. Azonban már egy évtizede felmerült az igény az új elvárásoknak is megfelelni képes, továbbfejlesztett verzióra. Néhány évtized múlva valószínűleg már csak internetes múzeumokban találkozhatunk az Internet hőskorát megalapozó 4-es verzióval.

Az első IP megújítását célzó protokollok már a 80-as évek végén megszülettek. Számos javaslat kidolgozása után az IPv6 megalkotásához a protokollokban rejlő újítások egybevetése vezetett. Az IPv6 specifikációk alapjait a 90-es évek elején az Internet számos újítását összehangolva, az Internet Engineering Task Force (IETF) fektette le.

Jelenleg a szakma szerint nincs alternatíva, mely áttörné az Internet fejlődésének gátjait. Az IPv6 térhódítása csupán idő kérdése!

Új igények, változások, újdonságok

Egyre gyakrabban hallunk mutimédia-tartalom továbbításról, biztonságos virtuális magánhálózatokról, mobil irodáról. A felmerült igények kielégítésére az IPv4-et is alkalmassá tették, viszont nagy előny, hogy az IPv6 ezeket szabvány szinten teszi lehetővé. További szempontként említhetjük új szolgáltatások hatékony bevezetésének lehetőségét is. Például többesküldés (lásd később) segítségével hatékonyan küldhet szét egy cég kizárólag az alkalmazottai számára üzeneteket, vagy juttathat el multimédiás csomagokat előfizetői számára.

A meglévő alkalmazások (például videokonferencia, webrádió) elterjedését is elősegítik az IPv6-ban alkalmazott szolgáltatások. Az üzleti területeken kívül a kutatási területeken is előrelépést jelent az IPv6, gondoljunk az elosztott hálózatokra vagy a GRID technológiára.

Az IPv4 évtizedes használata nemcsak új igények megjelenését eredményezte, de megtapasztalhattuk, hogy az egyes IP-ben megvalósított funkciók elhagyhatóak. Így kimaradt az IPv6 fejrészéből az ellenőrző összeg, melynek funkcióját magasabb rétegek veszik át. Ugyanígy kimaradtak a tördeléssel összefüggő mezők, a csomagméretről az IPv6-ot használó végpontoknak kell megegyezniük.

Összességében elmondható, hogy a célok között szerepel a végpontok közötti Internet paradigmájának visszaállítása. Így a peer-to-peer alkalmazások, vagy a végpontok közötti biztonsági megoldások és a címfordítók mellőzése mind az alapkoncepció részét képezik.

A címtér korlátainak ledöntése

Jelenleg a Föld lakosságának körülbelül 10%-a rendelkezik Internet eléréssel. Amennyiben célul tűzzük ki ezen arány növelését és figyelembe vesszük a népesség-növekedést, továbbá az önálló címmel rendelkező eszközök térhódítását, akkor világossá válik, hogy a jelenlegi IP cím mennyiség kevés. Az IPv4-es rendszerben problémát okoz az osztály alapú címzés, mely az egyes címtartományok allokációját csak meghatározott kvantumokban teszi lehetővé. A kiosztás elvi helytelensége miatt ez egy meglehetősen rossz választás volt.

A címtér szűkösségének problémája az Egyesült Államokban kevésbé jelentős, így az IPv6 elterjedése a világ többi részén (pl. Ázsiában, Kínában) hamarabb várható. Ezt könnyen megérthetjük, ha figyelembe vesszük, hogy például Kína 1,3 milliárd lakosa ellenére mindössze 22 millió IPv4-es címmel rendelkezik. A felhasználók száma mára már megközelíti ezt az értéket, viszont 2007-re 62,5 millió előfizetőt jósolnak. Japánban és Koreában is hasonló a helyzet, ezért ezek a kormányok óriási pénzeket költenek az IPv6 bevezetésére. Az IPv4-es címek 70%-a az Egyesült Államokhoz tartozik.

Napjainkban a kevés cím problémáját dinamikus címkiosztással is próbálják orvosolni, vagyis egy hálózatra feljelentkező gép nem minden esetben ugyanazt a cí-

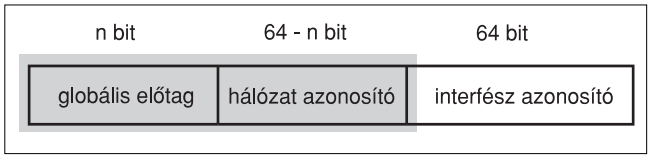
met kapja meg. A címek rendezetlensége miatt egy forgalomirányítónak akár több százezer bejegyzést is tárolni kell, ami nagyobb hardverkövetelményt támaszt az útvonalválasztókkal szemben. Erre nyújt megoldást a hierarchikus címkiosztás.

Az IPv6-os címek

Az IP címek kiosztása delegációs rendszerben történik, melynek legfőbb szerve az IANA (Internet Assigned Numbers Authority). A négy területi regisztrátor szervezet közül Európában a RIPE NCC (Réseaux IP Européens Network Coordination Centre) látja el ezt a feladatot. Ennek a tagjai többnyire Internet-szolgáltatók, illetve a szolgáltatók és regionális szervezet közé beékelődő nemzeti regisztrátorok is. Ezen a láncon keresztül kaphat bármely felhasználó IP címet, vagyis csatlakozási pontot a világ legnagyobb hálózatához. Az IPv6 egyik legjelentősebb újítása a 128 bites címek bevezetése. Ez sok nagyságrendnyi változást jelent az IPv4 által szolgáltatott 32 bites címtartományhoz képest.

Az IPv6 háromféle címzési módot különböztet meg. Egyesküldés (unicast) esetén egyetlen interfész a címzett. Többesküldés (multicast) esetén a címhez tartozó összes interfészhez megérkezik a csomag. Lehetséges, hogy egy címhez tartozó csoport tagjai közül csak valamilyen metrika szerinti legközelebbi interfész kapja meg a csomagot, ekkor használjuk a legközelebbinek való küldést (anycast). Az IPv4-ből jól ismert üzenetszórás az IPv6-ból teljesen kimaradt, helyét a többesküldés veszi át, melyhez számos előre definiált csoportot specifikáltak. [1]

A cím felépítése jól láthatóan hierarchikus *(1. ábra)*.



1. ábra IPv6 címek felépítése

Az *interfész azonosító* feladata, hogy egy IPv6-os kapcsolódási pontot azonosítson az adott hálózatban. Ezt legtöbbször a második rétegbeli hozzáférési közeg címének (pl. Ethernet hálózat esetén a MAC cím) segítségével történik. A MAC cím egyedisége elvileg garantált, gyakorlatilag több gyártó is figyelmen kívül hagyta ezt.

A középső, jelen esetben 16 bitnyi címrész azonosítja az adott hálózatot (SLA, Site Level Aggregation).

A korábbiakban említett címtípusok megkülönböztetésére és a hálózat megadására használjuk a globális előtagot. Ezt a hierarchikus kiosztás és feldolgozás miatt partícionálták. A korábban elfogadott ajánlás szerint az első 3 címtípus bitet (001) követő 13 bit tartozik a legmagasabb szintű szétosztáshoz (TLA, Top-Level Aggregation), melyet globálisan az IANA végez, így minden TLA egy-egy térséget azonosít. Ez a publikus gerinchálózatok szintje. Adott térségben lévő nagy szolgáltatók, vagy nemzeteknek a címek tovább oszthatóak (NLA, Next Level Aggregation), erre a célra a következő 32 bit használható.

Ez a megoldás magában hordozza a strukturált és átlátható címkiosztás lehetőségét, viszont mivel nem látható pontosan előre, hogy a különböző szinteken valójában mekkora tartományokat kell kijelölni a leghatékonyabb lefedéséhez, a kiosztás nagyon pazarlóvá válhat, ám a 128 bitnyi teljes hosszból eredő variációk száma így is igen nagy.

Az egyesküldési címeken belül megkülönböztetünk globálisan, adminisztrációs tartományra értelmezett és linken egyedi címeket. Míg a globálisan értelmezett címek egyedileg azonosítanak egy hosztot az Interneten, addig a adminisztrációs tartományra értelmezett címek egyazon tartományon belüli címzésre használhatóak globális előtag igénye nélkül. A linken egyedi címek csak adott linken belül érvényesek. Elsősorban autokonfigurációs és szomszédság felderítési célokat szolgálnak.

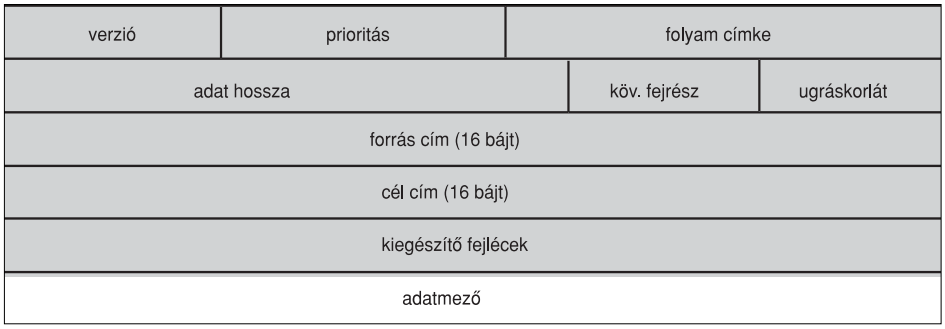
Az IPv6 datagram felépítése

Az IPv6-os címzés ismertetése után térjünk át az IPv6 gyakorlati alkalmazásaira, különös tekintettel az újdonságokra, ezek közül is elsőként az Internet Protokoll egyik alappillérét jelentő szállítási egység vizsgálatára, az IPv6 adatcsomagra.

Az IPv6 alap fejléce fix hosszúságú (ellentétben az IPv4-gyel) és ehhez kapcsolódhatnak még opcionálisan kiegészítő fejrészek. Így az útvonalválasztók számára a feldolgozás egyszerűbbé és gyorsabbá válik.

Minden, az IPv6 alapvető kapcsolatfelépítéshez szükséges adat szerepel *(2. ábra)* a fejlécben úgy, mint a forrás és cél azonosítását szolgáló címmezők, a folyamot meghatározó mező, az adatmező hossza és az ugráskorlát, mellyel a csomag élettartamát lehet szabályozni. Az ezeken felüli opcionális funkciókat a kiegészítő fejlécekben tárolták.

2. ábra
Az IPv6 datagram felépítése



Mobilitás

A 21. század informatikai igényeiben az elsők között szerepel a mobilitás. Ezt a tendenciát az IP világnak is figyelembe kell venni és a most létrejövő piacképes rendszereknek mindenképpen illeszkedniük kell ehhez. Nem meglepő tehát, hogy már az IP 4-es verziójánál is foglalkoztak a témával. Felmerült azonban egy jelentős probléma, amely alapjában véget vethetett volna a mobil IP történetének. Ugyanis ha egy hoszt IP címe megváltozik, akkor ez együtt jár azzal, hogy a felsőbb rétegbeli alkalmazások, melyek eddig ezt az címet használták azonosítóként, megszakadnak[8].

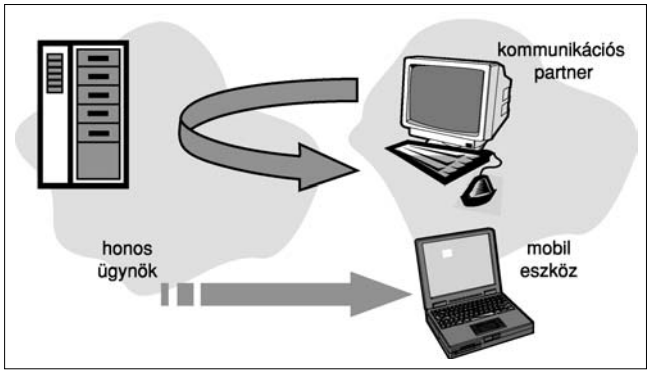
Mindezeket figyelembe véve dolgozták ki a Mobil IPv6-ot [3]. Az eddig kidolgozott protokollok alapján a mobil hoszt számára lehetővé válik az egyes hozzáférési pontok közötti mozgás úgy, hogy közben a hozzá más hozzáférési pontokról érkező csomagokat is megkapja. Ezt a mechanizmust az IP és a rajta működő protokollok az alkalmazások elől elrejtik. Itt már nincs szükség idegen ügynökökre, és nem igényel külön támogatást az aktuális helyi útválasztótól sem. De talán a legjelentősebb eltérés az útvonal optimalizálás bevezetése.

A mobil IP megvalósításának alapötlete igen egyszerű. A hoszt kétféle IPv6-os címmel rendelkezik. Az egyik, az úgynevezett honos cím, amely a honos hálózatbeli állandó cím. Amikor egy másik hozzáférési pontra kapcsolódik, azon egy felügyeleti cím fogja azonosítani. Ez utóbbiból akár több is lehet, de mindig ki kell jelölni az elsődlegest közülük. Ezt a mozgás szerint változó címet a mobil hoszt mindig ismerteti a honos hálózatával, amelyben az aktuális összerendeléseket a honos ügynök tárolja. Ez végzi a távol lévő hoszthoz beérkező csomagok továbbítását is, amennyiben az a honos címre érkezett.

A mobil hoszt természetesen közölheti aktuális IP címét a kommunikációs partnerével. A továbbiakban a kapcsolatfelépítés bemutatása következik, amelynek kétféle módja is van.

Egyik lehetséges megvalósítás, amikor a mobil hoszt nem közli aktuális címét partnerével, így a csomagok először a honos ügynökhöz kerülnek, majd innen jutnak el a másik félhez és vissza *(3. ábra)*. Ez a megoldás, melyet kétirányú alagutazásnak is neveznek, nem igényel külön IPv6 támogatást.

3. ábra Kétirányú alagút használata



Az útvonal optimalizálásnál *(4. ábra)* már szükség van az aktuális címre. Első lépésben a mobil csomópont ismerteti az aktuális cím-összerendelést, így ezután már közvetlenül küldhetők a csomagok a két végpont között, nincs szükség a honos ügynök közreműködésére.

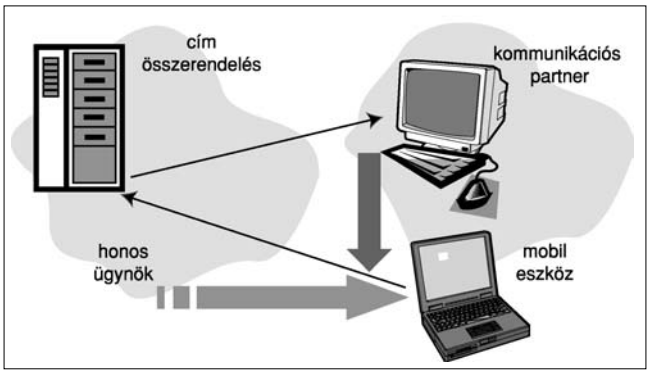
Biztonság

Az Internet gyors növekedésének és elterjedésének eredményeképpen az egyik legfontosabb megoldandó feladattá vált a biztonság problémája. Jelentőségét nem lehet és nem is szabad kétségbe vonni vagy lebecsülni, mára szinte elsődleges kérdéssé vált. A biztonság fogalma alá tartozó szolgáltatások sokfélék lehetnek.

Érdemes néhány szót szentelni a legfontosabb biztonsági követelményeknek. A köztudatban biztonság alatt gyakran a *titkosságot* értik, vagyis annak biztosítását, hogy a címzetten kívül más ne tudja értelmezni az elküldött információt. De természetesen nem ez az egyedüli követelmény és elmondható az is, hogy talán nem is minden esetben a legfontosabb. A titkossághoz talán legközelebb áll az *integritásvédelem*, amely biztosítja, hogy ne változtathassák meg illetéktelenek az elküldött csomag tartalmát. A *hitelesítés*, mely számos internetes biztonsági kérdést megold, célja annak ellenőrzése, hogy az összeköttetésben résztvevő felek valóban azok, akiknek mondják magukat. Végül a *visszajátszás elleni védelem* foglalkozik azzal a problémával, hogy egy csomagot ne lehessen a későbbiekben újra felhasználni.

Az Internet Protokoll 4-es verziója nem volt ezekre megfelelően felkészítve, a biztonsági kérdéseket főként alkalmazás szinten valósították meg, ami azt vonta maga után, hogy egyes funkciókat több alkalmazásba is beépítették. A probléma megoldására alkották meg az IP Security [24] protokollt, ami hálózati rétegbeli biztonságot nyújtott. Az IPv6 újat nyújtott abból a szempontból is, hogy olyan alapszintű biztonsági követelmények, mint a fejléc hitelesítése, a protokoll kötelezően megvalósítandó részei lettek. Ez nem azt jelenti, hogy a kommunikációnak ezután csakis és kizárólag hitelesítve és titkosan kell végbemennie, hanem csupán annyit rögzít, hogy az IPv6-ot megvalósító hosztoknak kötelezően rendelkezésre állnak a hitelesítést és titkosítást végző algoritmusok.

4. ábra Útvonal optimalizálás címösszerendeléssel



A protokoll rugalmas, többféle üzemmóddal is rendelkezik. A kívánt biztonsági szolgáltatások két kiegészítő fejrész segítségével valósíthatók meg [5]. A Hitelesítési fejrész az adatok hitelesítési és integritásvédelmi funkcióit látja el, az elküldött adatok titkosításáért pedig a Beágyazott Biztonsági fejrész felel .

Automatikus konfiguráció

Az Internet Protokoll 6-os verziójának újabb jelentős eredménye az automatikus konfiguráció. Ez képessé teszi a hosztokat arra, hogy saját maguk hozhassanak létre egy lokális címet azon az alhálózaton, amelyre éppen kapcsolódtak. Ezzel érthető módon a hálózati adminisztrációs feladatok is jelentősen lecsökkenthetők. Szolgáltatói oldalról lehetővé válik a hálózathoz tartozó hosztok címeinek egyszerű és gyors cseréje. Az automatikus konfigurációnak két változata van .

Az *állapotmentes automatikus konfiguráció* esetén a hoszt saját maga allokál egy címet [9]. Ennek egy lehetséges módját az IPv6-os cím ismertetése során már bemutattuk. Mielőtt azonban a címet használhatná, szükség van annak ellenőrzésére, hogy egyedi-e. Ezt a metódust nevezik duplikált címdetektálásnak (DAD).

A másik megoldásban rendelkezésre áll egy kitüntetett szerver, például a DHCP (Dinamikus Hoszt Konfigurációs Protokoll) szerver, amely közreműködik a hoszt konfigurációjában, így biztosítva, hogy ugyanaz a hoszt minden esetben ugyanazt a címet kapja a hálózattól. Ekkor *állapot alapú automatikus konfigurációról* beszélünk [10].

Nem lehet azonban figyelmen kívül hagyni, hogy az autokonfiguráció összességében időigényes művelet. Így mobil környezetben nem előnyös a címek duplikálásának vizsgálata. Ennek gyors és egyszerű megoldása, majd annak szabványosítása még várat magára.

Nemzetközi és nemzeti törekvések

Az Internet Protokoll 6-os verziója mára már szinte az egész világon elfogadottá vált és nem egy térségben figyelhetjük meg, hogy az Interneten kialakuló mind nagyobb verseny miként mozdítja elő az IPv6 regionális telepítését, a szolgáltatások alkalmazásának vizsgálatát, valamint a protokoll felhasználási lehetőségeinek folyamatos kutatását. Számos különböző projektet hoztak létre más és más célkitűzésekkel, azonban mégis elmondhatjuk mindegyik egy cél köré csoportosult: elősegíteni az IPv6 széleskörű alkalmazását.

Az *IPv6 Task Force* legfőbb feladata az IPv6 továbbfejlesztése. Ez egy világméretű összefogás, több regionális központtal, melyek Európában, Észak-Amerikában, Brazíliában, Kínában, Japánban, Dél-Koreában, Indiában és Iránban találhatóak. Tényleges eredményként könyvelhetjük el a *6BONE*-t [13], amely egy nemzetközi, kísérleti, virtuális számítógép hálózat. A 6BONE nem egy külön, e célra létrehozott infrastruktúrán üzemel, hanem egy IPv4 alapú Internet hálózaton alakították ki az adatátviteli csatornáit. A 6BONE kitűnő eszköznek bizonyult az IPv6 új útvonal választási stratégiáinak és algoritmusainak kipróbálására és az IPv6 szoftverek és berendezések ellenőrzésére. Mivel mára az IPv6 megérett a használatra, a 6BONE teszthálózatot és az összerendelt cím allokációkat fokozatosan megszüntetik.

Az 1990-es évek vége felé, és az új évezred elején számos kezdeményezés, projekt indult útjára, melyek közül már több be is fejeződött. Ilyen például a *6WINIT* projekt [14], melynek segítségével bevezetésre került az új vezeték nélküli mobil Internet Európában. Egy másik projekt, a *6INIT* [15] olyan technológiákat valósított meg, mellyel felügyelni lehet az információ feldolgozást, továbbá olyan technológiákat, melyek segítik a kommunikációt, a szélessávú hozzáférést, ezek együttműködését is beleértve.

Európa

Európa egyre inkább arra törekszik, hogy egységet alkosson nemcsak az államigazgatás és a gazdaság, hanem a technológiai fejlődés területén is. Az európai országok számára az IPv6 használatának egységes kidolgozása, az Internet piaci verseny mellett, a felzárkózást is biztosítja az információs sztráda alkalmazásában élen járó Amerikához és Ázsiához. Az Európai Unió ennek támogatására hozta létre az európai IST (Information Society Technologies) szervezetet. Mindezek előkövetelménye az egységes szolgáltatási alap megteremtése, melyet napjainkban is számos pilot projekt támogat.

Az egyik legismertebb a *6NET* [16], amely egy három éves EU projekt. Feladatai többek között egy nemzetközi IPv6 pilot hálózat telepítése és működtetése fix és mobil komponensekkel annak érdekében, hogy elfogadtassa az IPv6 fejlesztés eredményeit; migrációs stratégiák tesztelése; új IPv6 szolgáltatások, alkalmazások bevezetése, vizsgálata; címkiosztás értékelése stb. Magyar vonatkozása a dolognak, hogy 2002 óta a 6NET partnere a később bemutatatott HUNGARNET is.

Az *Euro6IX* [17] a mai napig a legnagyobb kutatás, melyet az európai IST indított el. Célja, hogy megtervezze és telepítse az első pán-európai nem kereskedelmi IPv6 hálózatot; ezen az infrastruktúrán IPv6-alapú alkalmazásokat, szolgáltatásokat fejlesszen és teszteljen; elérhetővé tegye a hálózatot egy speciális felhasználói csoport számára tesztelés céljából; elterjessze, összeköttetést és koordinációt biztosítson standard szervezetek (például IETF, RIPE) számára. Nagy jelentőségű egy hálózati szigetek összeköttetésére szolgáló gerinchálózati IPv6 létrehozása, melyet a *GÉANT* [18] projekt valósít meg, amely ezen felül még útvonalválasztással is foglalkozik.

Magyarország

Ugyanezen trendeket figyelhetjük meg hazánkban is. A NIIF [19] (Nemzeti Információs Infrastruktúra Fejlesztési Program) a magyarországi kutatói hálózat fejlesztésének és működésének programja. A program a teljes magyarországi kutatási, felsőoktatási és közgyűj-

teményi közösség számára biztosít integrált országos számítógép-hálózati infrastruktúrát, valamint erre épülő szolgáltatásokat, élvonalbeli alkalmazási környezetet, valamint tartalom-generálási, és tartalom-elérési hátteret. A NIIF IP gerinchálózatát HBONE-nak nevezzük. A HBONE a hazai akadémiai közösség számítógép hálózata. Az IPv6 elterjedéséből származó változások Magyarországot sem kerülték el.

A NIIF IPv6 törekvéseinek mérföldköveiből:
– az NIIF 6NET partnerré vált (2002. szeptember),
– IPv6 hálózat működik (2002. decembere óta),
– HUNGARNET IPv6 címkiosztás elindulása (2002. szeptember),
– natív IPv6 kapcsolat épült ki Bécsbe,
– GÉANT IPv6 pilot szolgáltatáshoz kapcsolódtunk.

A hazai eredmények közé tartozik többek között a hálózat menedzsment létrehozása, a teljesítmény mérése és az alkalmazási kísérletek.

A jelenlegi szolgáltatások közül említésre méltó a NIIF IPv6 címallokációs és regisztrációs szolgáltatása. Az NIIF IPv6 cím allokációs és regisztrálási jogokkal rendelkezik az NIIF/HUNGARNET tagintézmények számára. Ennek keretében vállalja, hogy azon tagintézményei számára, akiknek ilyen szolgáltatásra szükségük van, másodlagos, vagy akár elsődleges és másodlagos reverse IPv6 DNS szervert üzemeltet.

A jövő

Végezetül nézzük meg, milyen események, előrelépések várhatóak a közeljövőben. 2004. októberében Mandelieu-ben, Franciaországban rendezik meg az ötödik ETSI IPv6 „Plugtest” találkozót. Szintén a 2004-es évben kerültek, illetve kerülnek sorra Malajziában, Kínában, Németországban, Svájcban, Észak-Amerikában és Koreában az IPv6-os események folytatását alakító csúcstalálkozók.

Összefoglalva a leírtakat kijelenthetjük, hogy az Internet hatos számot viselő protokollja nem csupán egy újabb ígéretes, de a várakozásokat soha be nem váltó kutatási terv, hanem az elkövetkező évtizedek Információs Szupersztrádáját alapvetően meghatározó fejlesztés. Beszéljünk akár Információs Társadalomról vagy a fejlődő térségek felzárkóztatásáról, a mobilitás térhódításáról vagy új generációs hálózatbiztonsági megoldásokról, mindezek mögött az IPv6-ot fogjuk találni.

Irodalom

[1] IPv6 Cluster, „Moving IPv6 in Europe”, Edition of the 6Link European IPv6 Research and Develpoment Series, May 2003, www.ist-ipv6.org/pdf/ISTClusterBooklet2003.pdf

[2] „IP Address Services”, Internet Assigned Numbers Authority, www.iana.org/ipaddress/ip-addresses.htm

[3] R. Hinden, S. Deering, “IP Version 6 Addressing Architecture”. Request For Comments: 3513, IETF Network Working Group, April 2003

[4] R. Hinden, S. Deering, „Internet Protocol, Version 6 (IPv6) Specification”, Request For Comments: 2460, IETF Network Working Group, December 1998

[5] S. Kent, R. Atkinson, „Security Architecture for the Internet Protocol”, Request For Comments 2401, IETF Network Working Group, November 1998

[6] T. Aura, J. Arkko, “MIPv6 BU Attacks and Defenses” Internet Draft, IETF Mobile IP Working Group, February 2002

[7] D. Johnson, C. Perkins, J. Arkko, „Mobility Support in IPv6” Internet Draft, IETF Mobile IP Working Group, June 30,2003

[8] „IPv6 Stateless Address Autoconfiguration”, Request For Comments 2462, Network Working Group, December 1998

[9] R. Droms, J. Bound, B. Volz, B. Volz, T. Lemon, C. Perkins, M. Carney, „Dynamic Host Configuration Protocol for IPv6 (DHCPv6)”, Request For Comments 3315, IETF Network Working Group, July 2003

[10] Silvano Gai, “Internetworking IPv6 with CISCO routers”, McGraw Hill Text, March 27, 1998

[11] Information Society Technologies honlapja: www.cordis.lu/ist/

[12] IPv6 Task Force honlapja: www.ipv6tf.org/europe.php

[13] 6BONE honlapja: www.6bone.net/

[14] 6WINIT honlapja: www.6winit.org/

[15] 6INIT honlapja: www.6init.org

[16] 6NET honlapja: www.6net.org/

[17] Euro6IX honlapja: www.euro6ix.org/

[18] GÉANT honlapja: www.join.uni-muenster.de/geantv6/

[19] NIIF honlapja: www.iif.hu/

[20] 6LINK honlapja: www.6link.org/

[21] 6POWER honlapja: www.6power.org/

[22] IPv6 Forum: www.ipv6forum.com/

[23] Mohácsi János, Szigeti Szabolcs, Máray Tamás, “Az IPv6 hálózati protokollok”, http://tracy.ipv6.fsz.bme.hu/mydocs/networkshop97/

[24] J. Arkko, V. Devarapalli, F. Dupont, “Using IPsec to Protect Mobile IPv6 Signaling between Mobile Nodes and Home Agents”, IETF Network Working Group, June 2003

Áttérés az újgenerációs Internet használatára

BENYOVSZKY BALÁZS, MEZŐ BALÁZS, PALLOS B. RICHÁRD, LUKOVSZKI CSABA

Budapesti Műszaki és Gazdaságtudományi Egyetem, Távközlési és Médiainformatikai Tanszék
csaba.lukovszki@tmit.bme.hu

Kulcsszavak: IPv6, áttérési technikák, protokollfordítók

Az újgenerációs Internet ötletének megszületése és annak megvalósítása között eltelt idő közel sem volt olyan hosszú, mint amennyi idő szükséges lesz a protokoll elterjedéséhez. A legfontosabb feladat az, hogy az IPv6 megjelenése az Internet világában ne okozzon törést a világhálóban. Átjárható legyen mind az IPv4-et és IPv6-ot használó felhasználók számára is. Ez úgy lehetséges, ha minél hatékonyabb áttérési technikák segítik a két protokoll egyidejű működését. Az IPv6 számos előnyös tulajdonsága serkenti az áttérés folyamatát. Azonban vannak olyan tényezők is, amelyek lassítják, és késleltetik az IPv6 világméretű elterjedését.

Az Újgenerációs Internet megszületésének elsődleges és legfontosabb célja az volt, hogy megoldást nyújtson az egyre fogyatkozó IP címek problémájára. Ennek tökéletesen eleget tesz az új 128 bites cím, és a hierarchikus címzési rendszer. Az újabb felhasználók rövid időn belül rákényszerülnek az új címek használatára. Mikor a szolgáltatók már kifogynak a megszokott IPv4-es címekből, kénytelenek lesznek nyitni az IPv6 felé. Vannak azonban olyan előnyei az új protokollnak, ami esetleg arra ösztönzi a szolgáltatókat, felhasználókat, hogy ne várják meg ezt az időt, és hamarabb használják ki e hasznos tulajdonságokat.

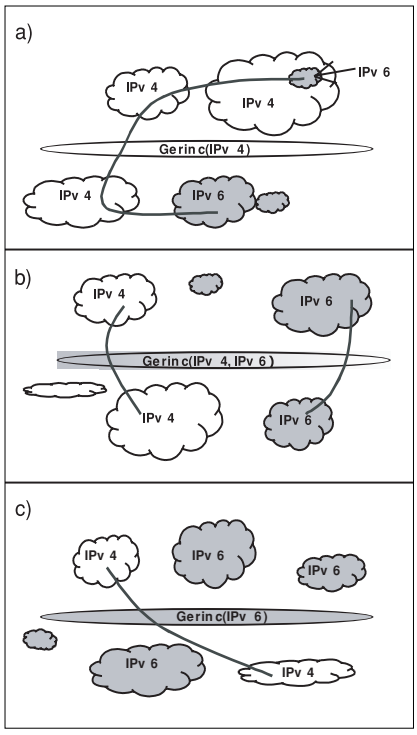
Egy komoly európai ösztönzésnek vehetjük azt a közelmúltban megjelenő hírt, mely szerint az Európai Unió területén 2007-2008-ig az IP 6-os verziójára akarnak áttérni. Ezzel kapcsolatban, már felülről irányuló ösztönzés érzékelhető.

Mindezek ellenére mégis azt tapasztalhatjuk, hogy az áttérés az IPv6 használatára nem olyan viharos sebességű, mint azt korábban sejthettük. Ennek oka lehet számos olyan IPv4-en alkalmazott kényszermegoldás, mely az új protokoll bevezetését nagymértékben késlelteti. Ilyen technika a rohamosan csökkenő IP címek problémájára ideiglenes megoldást nyújtó hálózati címfordító, ismert nevén a NAT (Network Address Translator). Ennek segítségével magánhálózati címeket oszthatunk alhálózati eszközeinknek, melyek csak kevés globális címen osztoznak az Interneten. Kívülről azonban emiatt nem tudjuk megcímezni az egyes eszközöket. A megoldás késlelteti az IP címek kimerülését, és ezáltal lassítja az áttérés folyamatát.

Az IPv6-os protokoll szélesebb körű bevezetése azonban nemcsak ezen okok miatt várat még magára. Számtalan előkészületi feladatot kell még elvégezni a világméretű elterjedése előtt. Létezik már egy IPv6-os gerinchálózat melyet 6Bone névre kereszteltek el, és amelyre a rákapcsolódás lehetősége számunkra is adott.

Azonban az IPv6-os áttérés nem tud azonnal megvalósulni. Számos olyan áttéréssel kapcsolatos vonatkozása van, melyeknek meg kell feleltetni az újonnan kapcsolódó eszközeinket. Az új protokollnak felülről kompatíbilisnek kell lennie a régivel, valamint az áttérés során a hálózat berendezéseinek egyaránt támogatniuk kell mindkét verziójú protokollt.

1. ábra Áttérési tendenciák



Nagyobb probléma viszont, hogy a hozzáférési és gerinchálózati szolgáltatók nincsenek felkészülve az IPv6-os átvitel kezelésére. Másrészt az alkalmazásoknak is támogatniuk kell az IPv6-os forgalom generálását. Mivel az áttérésnek nincs kitűzött időpontja, a fent említett tényezők mind az áttérési időszak elnyújtásához fognak vezetni.

Az áttérés tendenciái

Az IPv6-ra való áttérés menete három nagyobb időszakra bontható *(1. ábra)*, melyekben különböző megoldások használata célszerű az IPv4-es és IPv6-os protokollok egyidejű működtetéséhez.

Az IPv6 elterjedésének kezdeti szakaszában *(1/a. ábra)* jellemzően a mostani technológia által nyújtott infrastruktúrát kell használni. Ebben a szakaszban a meglévő gerinchálóz-

zatot kell úgy felhasználnunk, hogy a kisszámú, IPv6-alapú hálózatot össze tudjuk kapcsolni. Ezen, különálló IPv6 képességekkel rendelkező hálózatokat hívjuk IPv6-os szigeteknek. A szigetek összekapcsolásának megvalósítását tűzte ki célul a 6Bone projekt is. Az átállás e korai fázisában jellemzően az IPv6-ra már átállított hálózati eszközök egymás közötti kommunikációját kell megoldani IPv6 képességekkel nem rendelkező hozzáférési hálózatokon.

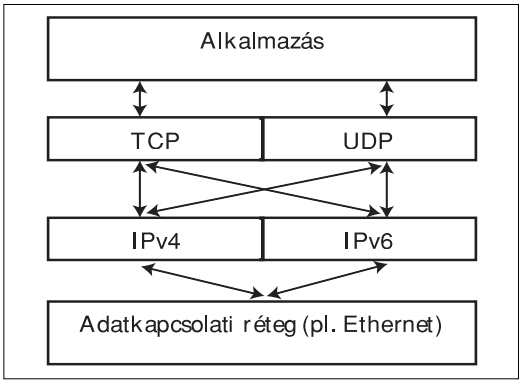
Az IPv6 terjedésével várhatóan egyre többen kapcsolódnak majd be az új protokoll használatába *(1/b. ábra)*. Emiatt a későbbiekben szükségessé válik, hogy a különböző verziójú IP protokollt támogató gépek képesek legyenek egymással kommunikálni. A végpontok számára további követelmény, hogy ez a kapcsolatfelépítés átlátszóan, a hálózat felhasználói számára észrevétlenül menjen végbe. Az IPv6 terjedésének utolsó szakaszában már jellemzően az IPv6 protokoll alapján működő eszközök lesznek túlsúlyban, miközben a gerinchálózatok is már az új protokollt fogják használni *(1/c. ábra)*. A gerinchálózatokon a még megmaradt IPv4-es eszközök üzemeltetése szintén valamilyen megoldást igényel.

Az IPv6-ra való áttérés hosszú folyamatnak ígérkezik. Emiatt szükséges, hogy az átállás közben a már működő IPv4 feletti alkalmazások zökkenőmentesen legyenek képesek illeszkedni az új protokollhoz. Ezen igény miatt születtek az áttérési technikák, melyek a két protokoll együttműködését hivatottak biztosítani.

Áttérési technikák alapjai

Az áttérési technikák alapjainak három nagy csoportját kell megemlítenünk. Ezek kombinálásával alakítható ki az adott lehetőségekhez legjobban illeszkedő megoldás.

Az első és legfontosabb technikája az áttérésnek az IPv6 csomagok átvételének megvalósítása egy IPv4-es eszközön. Ilyenkor a már meglévő IPv4-es protokoll verem „mellé” egy IPv6-ost is létesítünk, így az eszköz *kettős protokoll veremmel* [1] fog rendelkezni. A megoldás lényege, hogy a hálózati eszközök így képesek mind az IPv6-os, mind pedig a IPv4-es protokoll feldolgozására. Az eszköz felismeri a bejövő IP csomagokat, és a megfelelő veremnek továbbítja. Az alkalmazások a két protokollt egyidejűleg használhatják *(2. ábra)*.



2. ábra Kettős protokoll verem

Számítógépek számára ez a megoldás csupán szoftverfrissítést igényel, egyéb beágyazott rendszereket futtató hálózati eszközök esetében viszont sokszor komolyabb költségeket jelenthet. Egy hálózati útválasztó esetén például szükséges az útválasztó protokollok frissítése is. Ilyen frissített protokoll a RIPng, az OSPFv6 vagy a BGP4+.

Figyelembe kell venni azonban a többletterhelést is. Mivel a kettős protokoll veremmel rendelkező eszközöknek IPv4-es és IPv6-os címre is szükségük van, ezért ezen eszközöknek nagyobb többletterheléssel méretezendők a címek karbantartása miatt. Egy hálózati útválasztóban például meg kell oldanunk, hogy az eddigi 32 bites IP címekre méretezett útválasztó táblák mellett képes legyen az eszköz a 128 bites IP címekkel rendelkező IPv6-os hálózatok szerkezetét is tárolni.

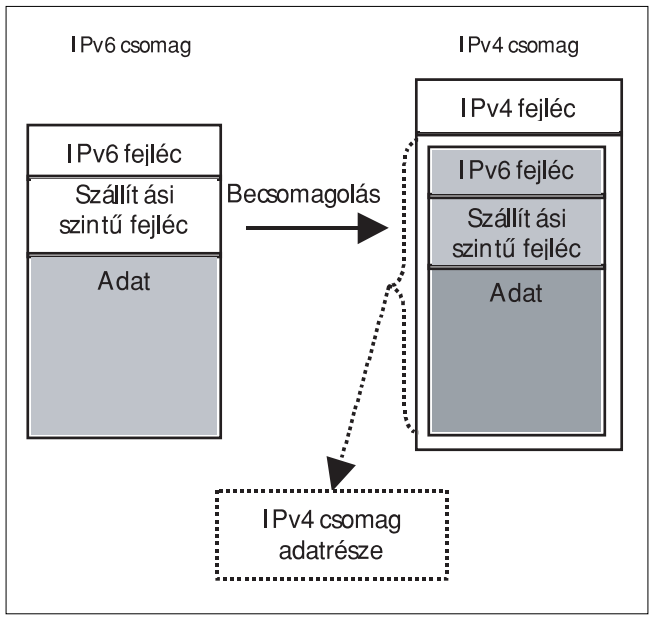
Ugyan az új technológia funkciói ezen esetben kis költséggel kihasználhatóvá válnak, de ez a megoldás semmit sem javít az IPv4-es címek elfogyásán. Szintén nem oldja meg a csak IPv6-ra, illetve csak IPv4-re felkészített csomópontok közötti kommunikációt.

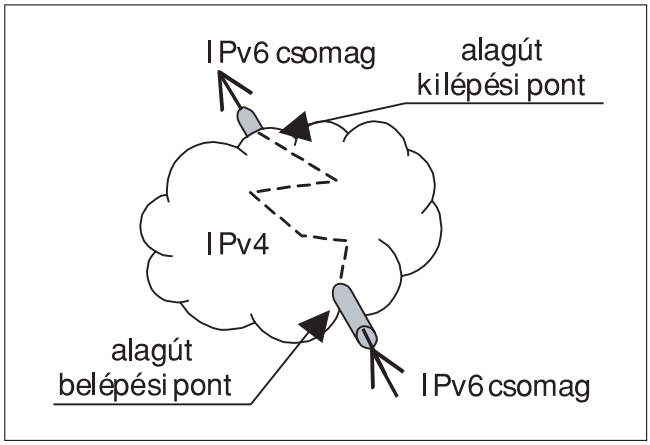
A kettős protokoll verem megvalósítása a legtöbb áttérési megoldás működéséhez alapkövetelmény.

A második áttérési módszer az *alagút technika* [1], amely nagyobb távolságok áthidalására született arra az esetre, ha közöttük nem áll rendelkezésre IPv6-os hálózati összeköttetés. Alkalmazásával lehetséges távoli elszigetelt IPv6-os hálózatok összekapcsolása valamilyen hordozó protokoll felett. A megoldás lényege, hogy az IPv6-os csomagokat egy azonos szintű protokoll csomagjaiba ágyazzuk *(3. ábra)*.

A beágyazás az IPv6 kezelésére alkalmatlan hálózatbelépési ponton történik meg. Ezen hálózatban az útválasztók nem tekintenek bele a csomag tartalmába, melynek hasznos terhében „utazik” a teljes IPv6-os csomag *(4. ábra)*.

3. ábra IPv6-os protokoll IPv4-be ágyazása





4. ábra Az alagút technika

A köztes hálózat a saját útválasztásának megfelelően (az ábrán szaggatott vonallal) továbbítja a csomagokat az alagút végpontja felé, a belső IPv6-os tartalommal nem foglalkozik.

Az alagút kilépési pontja az IPv6-os csomag célhálózatának pereme, mely szükségszerűen egy dupla protokoll veremmel rendelkező hálózati eszköz. A csomópont felismeri a beágyazott IPv6-os csomagot, és elvégzi a kicsomagolást a hordozó protokoll „burkából". Ezután a csomagot már IPv6-os csomagként továbbítjuk. A legelső ilyen megvalósítás a 6over4 technika volt [2].

Az alagút alapú technikáknak két nagy hátránya van. Mivel a meglévő protokollokat egy azonos szintű protokollba kell becsomagolni, ezért ez egy plusz fejlécet jelent minden csomagnak. Ekkor egyrészt növekedik a hálózati terhelés, másrészről a megnövekedett csomagméret miatt az alagútban szükség lehet a csomagok tördelésére.

Az alagút technikát használó megoldások két nagy csoportja a manuálisan beállított, és az automatikusan konfigurálódó alagutak. Az *automatikus alagutak* alkalmazásánál speciális, IPv4 kompatíbilis IPv6 címeket használunk. Itt az IPv6-os cím megegyezik az IPv4-es címmel, megfelelő számú nulla bitet eléírva. Ilyen IPv4 kompatíbilis IPv6-os címet használnak a kapcsolattartó eszközök az alagút belépési pontjának címzésére. Miután az IPv6-os hálózat eljuttatta a küldött csomagot az alagút belépési pontjának, az egy IPv4-es csomagba helyezi az IPv6-os csomagot. Így a belépési pont az IPv4-es protokollon indítja útjára a csomagot. Az útválasztás értelemszerűen az IPv4-es topológia szerint történik. A megoldás előnye a konfiguráció automatizálása, mellyel több távoli hálózatot is elérhetünk külön alagút adminisztráció nélkül. Hátránya, hogy az IPv6 128 bites címtere az IPv4 kompatíbilis címek miatt nem használható ki, ezért ebben az esetben meg kell elégednünk a jelenlegi protokoll 32 bites címeivel.

A *manuálisan beállított alagutak*nál adminisztrálni kell a távoli hálózat felé menő alagút be- és kilépési pontját, valamint az útválasztási táblákat. Kevés külső szigettel való összeköttetés esetén ez egyszerűen használható megoldás.

Az előzőekben tárgyalt módszerek nem teszik lehetővé a csak IPv4-et és a csak IPv6-ot támogató eszközök együttműködését. Az áttérési technikák harmadik típusa, a *protokoll fordítók*, e probléma megoldására születtek. Működésükből és a két protokoll közötti sok különbségből adódóan nem alkalmasak hosszú távon megoldani a csak IPv4 és a csak IPv6 képességű eszközök együttműködését. Sok esetben csak kellő körültekintéssel alkalmazhatóak, ennek ellenére a legtöbb esetben kielégítő megoldást nyújtanak.

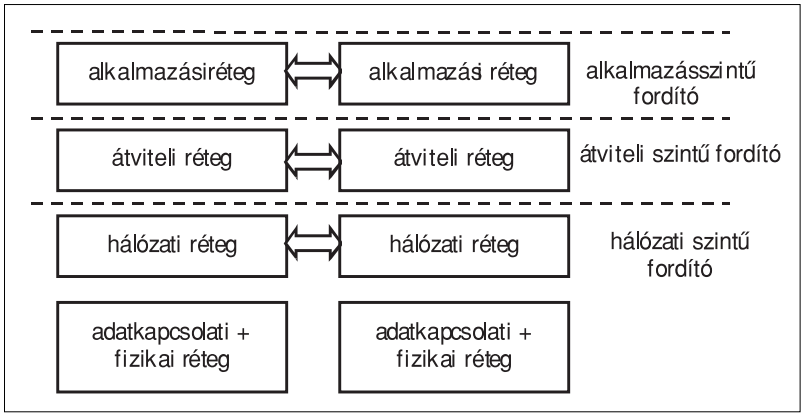
A protokoll fordítás műveletét e technikák a TCP/IP hivatkozási modell különböző rétegeiben végezhetik el. A modell rétegei alapján különböztetjük meg az alkalmazás szintű, az átviteli szintű és a hálózati szintű protokollfordítókat *(5. ábra)*.

Legalacsonyabb szinten a *hálózati rétegbeli protokollfordító* dolgozik. Ezen az absztrakciós szinten a fordítóprogram csupán a régi és az új protokoll fejléceit képes egymásba átfordítani. Működése egyszerű, mivel nem vizsgálja a csomagok tartalmát, csupán a fejlécet. Ebből adódóan a beágyazott IP címet tartalmazó protokollokat (mint például az FTP vagy a DNS) nem képes hibátlanul fordítani. Egyszerű konfigurálhatósága és a fordítás gyorsasága miatt használják.

Az *átviteli rétegben megvalósított fordítókat* kiszolgálókon lehet alkalmazni. Működésük lényeges eleme, hogy két külön kapcsolatot építenek fel a fordítandó kapcsolat esetében, így hasonló koncepcióban működnek, mint a proxy tűzfalak. A kezdeményező fél a szerveren keresztül próbálja elérni a célt. A szerver ekkor felépít egy kapcsolatot a kezdeményezővel az egyik (tegyük fel, IPv6-os) protokollon. Ezután kiépít egy kapcsolatot a cél felé, de ezt már a másik (IPv4-es) protokollon teszi. Ezek után a két fél között továbbítja a csomagokat, és a két protokollt egymásba átfordítja.

Működésének feltétele egy speciális DNS fordítóprogram, mely az IPv4-es címeket fordítja megfelelően az IPv6-os eszközök számára. Emiatt csak olyan IPv6-os végpont képes kapcsolatot létesíteni az IPv4 felé, mely a DNS fordítón keresztül keresi az IPv4-es cél cí-

5. ábra A fordítók típusai



mét. A visszakapott cím egy előtagból és a cél IPv4-es címéből áll. A speciális előtagot a hálózat a fordítást végző kiszolgáló felé irányítja, így halad azon keresztül a két fél közötti adat. A kiszolgáló a megfelelő címkonverziókat elvégezve valósítja meg a fordítást.

Az *alkalmazási rétegben megvalósított fordító*k lényege, hogy a fordító a teljes csomagtartalom alapján végez fordítást. Legtöbbször két inkompatíbilis hálózat közötti átjárónak alkalmazzák, emiatt tartalmaznia kell mindkét hálózat protokolljának implementációját. Mivel a teljes csomagtartalom alapján végez protokollkonverziót, ezért jóval nagyobb a számításigénye, mint az alacsonyabb rétegben működő fordítóknak. A használni kívánt alkalmazások számára külön kell a szükséges protokollokat a fordítóban megvalósítani. Emiatt ez a technika megoldja a beágyazott IP címeket tartalmazó protokollok fordításának nehézségét is.

Magán felhasználók

A kidolgozott áttérési technikáknak köszönhetően az otthoni felhasználók számára lesz a legkönnyebben megvalósítható az áttérés. Jelenleg ha egy magán felhasználó csatlakozni szeretne az IPv6-os hálózathoz, akkor erre a legegyszerűbb lehetősége valamilyen IPv6-os alagút alapú megoldás segítségével van.

A felhasználók szempontjából egyik legkényelmesebb megoldás az *alagút ügynök* [3] használata. Ehhez egy dedikált szervernek kell működnie azon szolgáltatónál, mely az IPv6-os hozzáférést szolgáltatja. Az ügynök feladata, hogy a felhasználó azonosítása után létrehozza az alagutat, szükség esetén módosítsa, használat után pedig lebontsa azt. Bejelentkezéskor elvégzi az alagút szerverekben szükséges beállításokat és elkészíti a DNS bejegyzéseket is az IPv6-hoz csatlakozó felhasználó számára, továbbá beállítja a használt alagút végpontjainak címeit. Működéséhez meglévő IPv4-es infrastruktúra szükséges, előnyeit és hátrányait az alagút alapú megoldásoktól örökli.

Hasonló, ám időben jóval előrébb tekintő megoldás *a kettős protokoll verem alapú áttérési technika* (DSTM) [4]. Erre az áttérés késői szakaszában lesz szükség, amikor a hálózatok jórészt az IPv6-os protokollal működnek.

A DSTM egy szerver-kliens alapú modell, melyben a szervert a hozzáférési szolgáltató biztosítja. A kliens ekkor már főként az IPv6-os protokollt fogja alkalmazni, de szükség lesz rá, hogy kettős protokoll veremmel rendelkezzen. Amennyiben a kliens IPv6-os végponttal szeretne kapcsolatot teremteni, azt gond nélkül megteheti. Ha viszont csak IPv4 támogatással rendelkező címet próbálna elérni, szüksége lesz egy IPv6 feletti IPv4-es alagútra a távoli végpontig. A kliensnek az alagút elkészítéséhez szüksége van IPv4-es címre is, mellyel alapesetben nem rendelkezik. Ekkor jön a képbe a DSTM szerver, mely a kommunikáció időtartamára egy IPv4-es címet szolgáltat az alagút végpontnak és elvégzi annak beállításait.

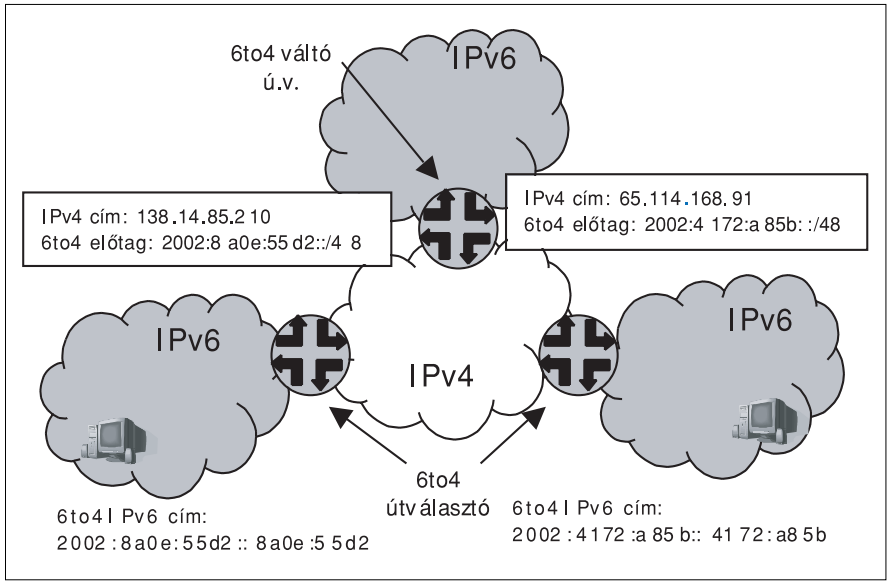
Hozzáférési szolgáltatók

Egy hálózaton belüli elszigetelt IPv6-os alhálózatok összekötésének leggyorsabb módszere a kézzel adminisztrált alagutak alkalmazása. Ezeket egyszeri beállítás után stabilan lehet használni IPv4-es hálózat áthidalására. Több ilyen elszigetelt hálózat esetén viszont a *6to4* [5] alagút megoldás alkalmazása célszerű, mely képes automatikus alagutakat építeni, így elkerülhető a bonyolult alagútmenedzselés. Működésekor az IPv6-os alhálózat egy speciális IP cím előtaggal [6] hirdeti a hálózathoz tartozó, távoli hálózatba vezető alagutak belépési pontjainak címeit *(6. ábra)*. Ezek szükségszerűen dupla protokoll veremmel rendelkező alagút végpontok, melyeknek van IPv4-es címe. Ezen IPv4 címeket az alagút végpontok a közös IPv4-es hálózaton egymás között hirdetik. Így az IPv4-es hálózathoz kapcsolódó IPv6-os alhálózatok határaihoz eljutnak ezen információk.

A meghirdetett címek alapján az IPv6-os szigetek egymás között már képesek alagutakat felépíteni. A megoldás érdekessége, hogy a címek hirdetésére elegendőek lehetnek a már régóta működő IPv4 feletti útválasztó protokollok.

Az eddigi alagút alapú megoldásokhoz szükség volt globális IPv4-es címre, mely az alagút egyik végpontját jelentette. Amennyiben szükségünk lenne privát címtartományból, például hálózati címfordítóval (NAT) védett hálózatból elérni távoli IPv6-os hálózatokat, a *Te-*

6. ábra A 6to4 működése



redo [7] megoldást kell szemügyre vennünk. Ez az egyetlen megoldás mely képes NAT mögül is megfelelően működni.

Az *állapotmentes IP/ICMP fordító* [8] (SIIT) a hálózati rétegben működő protokollfordító. Feladata, hogy a fordítást végző eszköz számára állapotmentesen, azaz belső állapotok tárolása nélkül valósítsa meg a protokollok fejlécei közötti váltást. Megoldja az ICMP csomagok fordítását és tördeli az IPv4-es csomagokat, hogy az IPv6-os hálózat maximális csomagméretének megfeleljenek. Külön IPv6 címeket használ azon IPv4-es eszközök számára, melyek értelmezni tudják az újabb protokollt. Ezen címek az IPv4 lefordított címek. Az IPv6-ot nem támogató eszközöket az IPv4 összerendelt címekkel azonosítja.

Működése során a fordító az IPv4 és IPv6-os fejléceket alakítja át egymásba. Alkalmazása azon esetekben lehetséges, ahol a teljes alhálózat támogatja már az IPv6-ot, és szükséges a külső IPv4-es címek használata. A kapcsolatokhoz az IPv6-os végpontoknak IPv4 lefordított címekre van szükségük, melyeket egy speciális DHCP kiszolgáló oszt ki számukra.

Az SIIT fordítókat az IPv6-os hálózat határán kell elhelyezni. Állapotmentes működésük miatt nem szükséges, hogy egy kiszemelt kapcsolatnak minden csomagja egy eszközön haladjon keresztül, ezért a határoló pontokon párhuzamosan több SIIT eszközt is lehet használni a terhelésmegosztásra.

Szintén hálózati rétegbeli mechanizmus a *hálózati címfordító és protokollfordító* (NAT-PT), mely a NAT megoldás egy kiterjesztése [9]. Működése a felhasználó számára hasonlóan átlátszó protokollfordítást tesz lehetővé, mint az SIIT megoldás. Fontos kiemelni viszont, hogy a NAT-PT szintén az IPv6 sziget határán helyezkedik el, de mivel a NAT-PT állapottartó, ezért szükséges, hogy minden a szigetről kifele kezdeményezett hálózati kapcsolat rajta, vagy vele együttműködő útválasztókon keresztül haladjon. A NAT-PT működésekor minden, az IPv6-os szigetet azonos cél felé elhagyó csomaghoz, dinamikusan rendel egy IPv4 címet az általa használt IPv4 címtérből. Így a NAT-PT az IPv6-os szigetről kapcsolatot kezdeményező fél számára átlátszóan továbbítja a csomagokat az IPv4-es cél felé.

A NAT-PT előnyös tulajdonsága, hogy az IPv6-os szigeten nincs szükség kettős protokollveremre. A kiszolgáló beállítása egyszerű, az IPv6-os alhálózat számára átlátszó.

Az átviteli rétegben működő, szervereken megvalósítható mechanizmus a *Transport Relay Translator* [10] (TRT). A TRT feladata, hogy a rajta keresztülhaladó csomagokat elfogja, és átfordítsa IPv4-ről IPv6-ra és viszsza ugyanígy. Egy kapcsolat felépülése után fontos, hogy a teljes kapcsolat a TRT kiszolgálón haladjon keresztül. Működéséhez szükséges egy speciális DNS kiszolgáló, mely az IPv6-os sziget által megcímzett távoli hálózaton elhelyezkedő IPv4-es cél címét egy megfelelő előtaggal kiegészítve IPv6-os címként oldja fel. Az előtag feladata, hogy a hálózat a TRT kiszolgáló felé továbbítsa a távoli, IPv4-es félnek küldött csomagot, melyet a kiszolgáló lefordít és továbbküld a cél felé.

A TRT megvalósítása a *SOCKS64* [11] technika, mely az átviteli rétegben működik, ehhez kettős protokoll veremmel rendelkező kiszolgáló, valamint a kliensek hálózati programkönyvtárainak módosítása szükséges, mivel a kliens és a szerver egy speciális SOCKS protokollon kommunikálnak. Az eredeti SOCKS implementáció [12] tűzfallal izolált hálózatokon keresztül nyújtott átjárót a belső hálózat IPv4-es és a külső hálózat szintén IPv4-es hálózati eszközei között. A SOCKS64 megvalósítás már lehetővé teszi IPv4-es és IPv6-os csomópontok kommunikációját mind homogén (azonos protokollok között pl. IPv4-IPv4), mind heterogén (különböző protokollok esetén pl. IPv6-IPv4) kapcsolat esetén.

Teljes protokoll konverziót valósítanak meg az *alkalmazás szintű átjáró*k [13] (ALG). Az átjárók két inkompatíbilis hálózat határán helyezkednek el, és szükségszerűen rendelkeznek mindkét hálózat protokollvermével. A két hálózat közötti kommunikáció teljes protokoll konverzióját elvégzik. Mivel ehhez szükséges a teljes csomagtartalom vizsgálata, ezért az alkalmazás szintű átjárók működése jelentősen nagyobb terheléssel jár, mint az alacsonyabb szinten megvalósított áttérési megoldásoké.

Az IPv4 és IPv6 közötti különbségek miatt a fordítás során bizonyos funkciók elveszhetnek, de ezektől eltekintve az ALG megoldja az összes problémát, mely a többi technikánál jelentkezhet.

Összefoglalás

A szolgáltatóknak a fent említett technikák kiválasztásánál figyelembe kell venni a korábban alkalmazott átviteli technológiákat. Munkájukat nagyban megkönnyíti, hogy az Internetes társadalom már eddig is nagy erőfeszítéseket tett arra, hogy megvizsgálja az IPv6 ATM-mel és MPLS technológiával való együttműködését.

Az áttérési technikák hivatottak tehát az IPv4 és IPv6-os protokollok együttműködését megvalósítani. De mint érzékelhető, az áttérés komplex feladat, melyre nincs egyértelmű megoldás. Minden esetben az adott környezethez és infrastruktúrához legjobban megfelelő technikát érdemes alkalmazni.

Irodalom

[1] R. Gilligan, E. Nordmark,
"Transition Mechanisms for IPv6 Hosts and Routers",
Request for Comments 1933,
Network Working Group, April 1996
[2] B. Carpenter, C. Jung,
"Transmission of IPv6 over IPv4 Domains
without Explicit Tunnels",

Request for Comments 2529,
Network Working Group, March 1999
[3] A. Durand, P. Fasano, I. Guardini, D. Lento,
"IPv6 Tunnel Broker",
Request for Comments 3053,
Network Working Group, January 2001
[4] Jim Bound,
"Dual Stack Transition Mechanism ",
INTERNET DRAFT, July 2003
[5] B. Carpenter, K. Moore,
"Connection of IPv6 Domains via IPv4 Clouds",
Request for Comments 3056,
Network Working Group, February 2001
[6] C. Huitema,
"An Anycast Prefix for 6to4 Relay Routers",
Request for Comments 3068,
Network Working Group, June 2001
[7] C. Huitema,
"Teredo: Tunneling IPv6 over UDP through NATs",
Internet Draft, February 5, 2004
[8] E. Nordmark,
"Stateless IP/ICMP Translation Algorithm (SIIT)",
Request for Comments 2765,
Network Working Group, February 2000
[9] G. Tsirtsis, P. Srisuresh,
"Network Address Translation –
Protocol Translation (NAT-PT)",
Request for Comments 2766,
Network Working Group, February 2000
[10] J. Hagino, K. Yamamoto,
"An IPv6-to-IPv4 Transport Relay Translator",
Request for Comments 3142,
Network Working Group, June 2001
[11] H. Kitamura,
"A SOCKS-based IPv6/IPv4 Gateway Mechanism",
Request for Comments 3089,
Network Working Group, April 2001
[12] M. Leech, M. Ganis, Y. Lee, R. Kuris,
D. Koblas, L. Jones,
"SOCKS Protocol Version 5",
Request for Comments 1928,
Network Working Group, March 1996
[13] K. Yamamoto, M. Sumikawa,
"Overview of Transition Techniques for
IPv6-only to Talk to IPv4-only Communication",
Internet Draft, March, 2000

Hírek

20 éves a Cisco Systems. Az 1984-ben alapított cég neve az elmúlt két évtizedben egybefonódott az Internet történetével. Len Bosack és Sandy Lerner, a Stanford Egyetem kutatói két évtizede alapították meg a Cisco Systems céget, amely nevét San Francisco városáról kapta. Bosack és Lerner a különálló hálózatok összekötésének lehetőségeit vizsgálta a Stanford Egyetem két épülete között. Ahhoz azonban, hogy a hálózatokat ténylegesen összekapcsolhassák, egy olyan új technológiára volt szükség, amely képes kezelni a különböző helyi hálózati protokollokat. Ez az elképzelés vezetett a többprotokollos útválasztó megszületéséhez.
Az alapítás óta eltelt két évtized alatt a Cisco a hálózati gazdaság előfutárából jelentős nemzetközi nagyvállalattá vált. A Cisco Systems IP alapú hálózati megoldásai biztosítják mind az Interneten, mind a legtöbb nagyvállalat, felsőoktatási és kormányzati intézmény számára az adatkommunikációs kapcsolatot. A világhálón közlekedő információk döntő részét a cég rendszerei szállítják.

A Cisco Systems Cisco Carrier Routing System (CRS-1) nevű terméke a világ legnagyobb kapacitású internetes útválasztójaként bekerült a Guinness-rekordok könyvébe.
A 92 terabites összteljesítményű útválasztórendszer az eddigieknél mintegy kétszer nagyobb adatforgalmat tesz lehetővé. Az Egyesült Államok kongresszusi könyvtárának teljes gyűjteménye 4,6 másodperc alatt letölthetővé válik. Ugyanennek az anyagnak a letöltése egy másodpercenként 56 kilobites sebességet biztosító behívásos modemmel körülbelül 82 évig tartana. A számok magukért beszélnek.
A valós idejű hangátviteli és csevegőszolgáltatásnak köszönhetően egyszerre akár egymilliárdan is játszhatják ugyanazt az online játékot. Az Egyesült Államok összes házttartása (105 480 101 otthon) 872 kbit/s sebességű nagy sávszélességű kapcsolathoz juthat. A hálózaton keresztüli egyéni videolejátszással egyszerre 15 millióan élvezhetik a 6 Mbit/s sebességű, kitűnő minőségű videoprogramokat. Egyszerre megközelítőleg 12 415 felhasználó töltheti le ugyanazt a 7,4 GB-os filmet, és ez mindössze 1 másodpercet vesz igénybe. A CRS-1 az internetes szolgáltatások és multimédiás alkalmazások eddig nem tapasztalt mértékű elterjedése előtt nyitja meg az utat.

Az ASN.1 nyelv a protokolltervezésben

POÓS KRISZTIÁN, PAPP ANDRÁS

Veszprémi Egyetem, Műszaki Informatikai Kar, Információs Rendszerek Tanszék

poos.krisztian@irt.vein.hu, papp.andras@irt.vein.hu

Reviewed

Kulcsszavak: kódolási eljárások, formális leíró technikák, mobil adatátvitel

Az ASN.1 nyelv különböző alkalmazások közötti üzenetek leírására szolgál, mint ilyen, magas szintű üzenetleírási formákkal rendelkezik, megkímélve ezzel a protokolltervezőket attól, hogy bit vagy bájt szinten kelljen foglalkozniuk a kommunikációban résztvevő üzenetek felépítésével. Kezdetben e-mail üzenetek leírására használták. Azóta az ASN.1 olyan alkalmazások széles körében is használatossá vált, mint például a hálózat-felügyelet, a biztonságos e-mail, mobil telekommunikáció, légiirányítás, vagy VoIP. Cikkünkben ezt a nyelvet és sokrétű alkalmazhatóságát mutatjuk be.

1. A formális leíró technikák és az ASN.1 kapcsolata

Az ASN.1 nyelv alkalmas adattípusok formális leírására, szabályhalmazokat definiál, amelyekkel bármely adattípus átalakítható egy továbbítható bitfolyammá. A nyelvet (ITU-T X.680 [6], X.691 [8]) alkalmazva a tervezésben időt nyerünk és csökkenthetjük a hibalehetőségeket. A kódolás feladata a modulokkal leírt adatspecifikáció olyan formára hozása, hogy egyértelműen azonosítható legyen a vételi oldalon. Ehhez az ajánlások három szabályhalmazt definiálnak, a típusok és a típusból származtatott értékek reprezentációit, az opcionális mezőt valamint az azonos típusú mezőt. A kódolási szabályokat az X.690-es ajánlás [7] tartalmazza, elnevezésük rendre a következő: BER, CER, DER. Ennek kiegészítése az X.691 és X.693 [9], ami a PER és XER kódolási szabályokat adja a specifikációhoz.

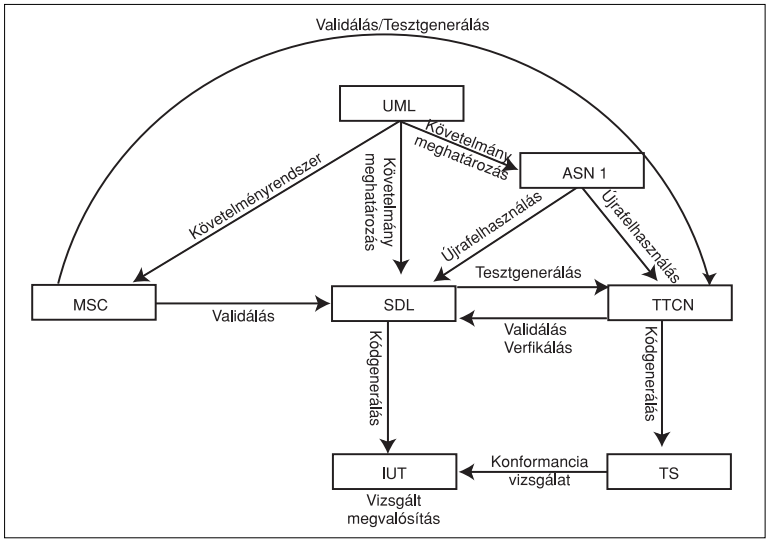
Célszerű egy rendszer viselkedését SDL-ben úgy leírni, hogy az üzenetváltáshoz ASN.1-es adattípusokat használjunk, mert a TTCN nyelv ismeri az ASN.1 adatdefiníciókat, és ez a későbbi a tesztelés során hasznos lehet. Az SDL processzek [3] változókat manipulálnak, amik értékekkel rendelkeznek, melyeket a megfelelő kifejezések kiértékelése adja. Egy változónak csak egy, adott adattípusú értéke lehet. Az adattípust literálok és operátorok összessége együttesen jellemzi. A literálok olyan nevek, amelyek az egyes értékeket jelölik, az operátorok pedig olyan függvények, amelyeket a literálok és a változók fölött alkalmazunk kifejezések szerkesztéséhez. Az operátorok szemantikáját az SDL-ben axiómákkal adjuk meg.

Egy absztrakt adattípus az adatobjektum funkcionális jellemzőit adja meg, tehát a művelet eredményét az adatobjektummal definiálja és azt, hogy megszorítások nélkül miként lehet az adatobjektum által képviselt értékeket megkapni. Az absztrakt adattípus egy vagy több típust (eng. *sort)* definiál, amelyek ismert értékkészlettel és az ezen értelmezett műveletekkel jellemezhetők. Általánosan az absztrakt adattípus egy vagy több osztályt tartalmaz, amelyekre definiálnia kell az operátorokat, amelyek operandusai lehetnek a különböző osztályok, valamint az egyenleteket amelyek eredménye mindig egyetlen osztályt ad.

Tehát az absztrakt adattípus tulajdonképpen osztályok, operátorok és egyenletek összességéből áll.

Az SDL-ben az absztrakt adattípus nincs megnevezve, – csak impliciten létezik – és ennek részeit definiáljuk a különböző adattípus deklarációkkal. Ezt parciális típusdefiníciónak nevezzük. A rendszerspecifikációs fa bármely pontjában egyetlen absztrakt adattípus definíció létezik, amelyet a fa gyökerétől a kérdéses pontig parciális típusdefiníciók alkotnak. Természetesen az absztrakt adattípusok is rendelkeznek öröklődéssel, de nem teljes hierarchia szinten, hanem csak a közvetlen ősöktől van öröklés. A fa egy csomópontjában csak a csomópont direkt őseiben szereplő parciális típusok alkalmazhatóak.

1. ábra Az ASN.1 helye az FDT-k között



Az ASN.1 szabvány [6] megkülönböztet kis- és nagybetűket. Az adattípus kezdőbetűje nagy, a típusból definiált értéké pedig kicsi, valamint a struktúra egy mezője is kisbetűvel kezdődik. A definíció jele '::=', amely egyben a típus és az értékdefiníció jele is.

2. Az ASN.1 története, felhasználhatósága

A számítástechnikai fejlődés kezdetekor a hardvergyártó cégekre nem volt jellemező, hogy a legyártott chipek, processzorok kompatibilisek legyenek egymással. Több cég párhuzamosan fejlesztett és termelt, így változatos architektúrákat készítettek a piac számára, melyeket természetesen specifikusan lehetett csoportosítani.

Ma is sokféle architektúra létezik (x86, Ultrasparc, PowerPC, stb), azonban az informatika hajnalán még több rendszerrel lehetett találkozni. Ilyen különbség például az, hogy nagyon sok rendszer ASCII kódolást, az IBM mainframe-jei EBCDIC kódolást használnak, valamint a PC-k 2-es komplemensű, 16 és 32 bites memória-szavakat, a mainframe-ek 60 bites, egyes komplemensű aritmetikát használnak. Hasonlóképpen felfedezhetünk ábrázolási eltéréseket egy Token-Ring és egy Ethernet hálózat között is. Mindegyik esetben az adatokat más módon kezeli a két különféle architektúra, így szükség van valamilyen közvetítő módszerre, amellyel a kétféle rendszer között adatcserét tudunk végrehajtani.

Amennyiben egy architektúrával dolgozunk, még akkor is felmerülhet adatábrázolási különbség, mert attól függetlenül, hogy az általunk használt eszközök egyazon architektúrára épülnek, még többféle operációs rendszert, és ezen belül sokféle programozási nyelvet használhatunk. Példának vegyünk alapul egy adatstrukúra definiálást egy C és egy Pascal kódrészlettel, *(2. ábra)*.

```
typedef struct header {
   int      mezo1;
   char[8]  mezo2;
   boolean  mezo3;
} header
```

```
Type header = Record
   mezo1 : Integer;
   mezo2 : String[8];
   mezo3 : Boolean;
end;
```

2. ábra C és Pascal kódrészlet

Látható, hogy az adatábrázolás más módon történik a két nyelven. Például egy név tárolására az egyiknél karakterek sorozatát, míg a másiknál string típust használunk.

Ahhoz, hogy egyik architektúráról a másikra, vagy egyik programnyelvről a másik számára érthetővé tegyük a kódot vagy az adatot, valamilyen konverziós eszközre van szükségünk. Definiáljuk ehhez a szükséges szintaxisokat.

Konkrét szintaxisnak nevezzük a küldeni kívánt adatreprezentációkat, egy adott programozási nyelvben. Azért szintaxis, mert figyelembe veszi az adott nyelv lexikai és nyelvtani szabályait, és azért konkrét, mert az alkalmazások kezelik és eleget tesz a gépek architekturális feltételeinek.

Hogy megszabaduljunk a konkrét szintaxisok változatosságától, a továbbítani kívánt adatokat úgy kell leírni, hogy ne legyenek tekintettel a használt programnyelvekre. Ettől függetlenül azonban a leírásnak figyelembe kell vennie egy bizonyos nyelv mind lexikai, mind grammatikai szabályait, azonban mindig függetlennek kell maradnia a programozási nyelvektől és soha nem telepíthető közvetlenül a gépbe. Az ilyen leírást nevezzük *absztrakt szintaxisnak,* Abstract Syntax Notationnek (ASN) pedig a nyelvet, mellyel az absztrakt szintaxis leírható.

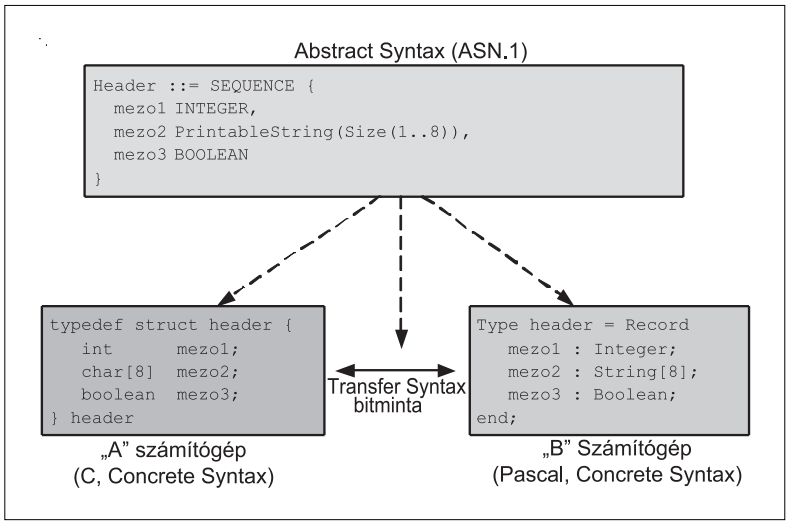
A programozási nyelvektől való függetlenség miatt az absztrakt szintaxisnak legalább olyan erősnek kell lennie, mint bármely nyelv adattípusának, ami tulajdonképpen egy rekurzív jelölés, amely lehetővé teszi komplex adattípusok létrehozását alap adattípusokból (string, int, char stb.) és típuskonstruktorokból (struct, union stb.)

A számítási eszközök általi kezelés és értelmezés alatti bármely félreérthetőség elkerülése végett az absztrakt szintaxisnak formálisnak kell lennie. Az absztrakt szintaxis precízen definiálja az adatot, azonban nincs szemantikai funkciója.

Már csak egyféle szintaxist kell megvizsgálnunk, mégpedig az *átviteli szintaxist.* Ez tulajdonképpen egy félreérthetetlen oktett string halmaz, mely az absztrakt szintaxis egy értékét reprezentálja az átvitel során. Természetesen ez az átviteli szintaxis teljesen az absztrakt szintaxistól függ, csak annyit határoz meg, hogy az adatokat hogyan kell továbbítani az absztrakt szintaxis alapján. Valójában az átviteli szintaxis strukturálja és irányítja a bájtokat, melyeket a másik gépnek küldünk. Az absztrakt szintaxistól eltérően ez egy fizikai mennyiség, és ebből fakadóan számításba kell vennie a bájtok elrendezését, a bitek súlyát stb.

A különböző átviteli szintaxisokat össze lehet kapcsolni egy egyszerű absztrakt szintaxissal. Ez főleg akkor érdekes, amikor megnő az átviendő adat mennyisége, és sokkal bonyolultabb kódolás szükséges: ilyen és ehhez hasonló esetekben lehetőség van az átviteli szintaxis megváltoztatására anélkül, hogy hozzányúlnánk az absztrakt szintaxishoz. Egy egyszerű ASN.1 adatleírásból automatikusan annyi konkrét szintaxist és annyi eljárást tudunk származtatni – ami létrehozza az átviteli szintaxist a kódolókba és dekódolókba –, amennyit csak akarunk.

Az ASN.1 fordító feladata az automatikus generálás végrehajtása, melyet a *3. ábrán* látható szaggatott vonalak mentén haladva végez el. A folyamat során tetemes fáradozástól kíméli meg a felhasználót, miközben lehetővé teszi tetszőleges számú számítógép összekapcsolását. A fordítóba implementálni kell néhány kódolási szabályt, melyek leírják a kapcsolatot az absztrakt és a átviteli szintaxis között.



3. ábra A fenti példára vetített szintaxis hármas

3. ASN.1 az OSI rétegekben

Az ASN.1 [2] felhasználásának bemutatása után térjünk rá használatára az OSI modell rétegeiben. A hét réteg közül csak a két legfelsőre (megjelenítési, alkalmazási réteg) térünk ki, mert csak ezekben jelenik meg az ASN.1.

A megjelenítési réteg az OSI [4] rétegmodell hatodik rétege, és legfőbb feladata biztosítani az adatok kódolását, dekódolását. Ahogy az előző fejezetben láthattuk, az adatábrázolás az architektúrától és a nyelvtől is függhet, ezért egy általános ábrázolás szükséges az adatcsere lebonyolításához. A megjelenítési réteg biztosítja, hogy az adat ebben a formában kerüljön továbbításra, viszont nem törődik az információ jelentésével. Ez gyakorlatilag az, hogy a két rendszernek az adattovábbítás előtt meg kell állapodnia a használni kívánt kódolási szabályban (BER, CER, DER, PER, XER stb.).

Így a megjelenítési réteg az alkalmazási réteg számára biztosított szolgáltatásai a következőek:

– egyezkedés az átviteli szintaxisról
– átviteli szintaxisok egy gyűjteményének ismerete
– fordítás, a konkrét szintaxis kódolási szabályainak használatával az átviteli szintaxisra és vissza
– az egyezkedés során meghatározott átviteli szintaxis összekapcsolása az alkalmazásban elfogadott absztrakt szintaxissal.
– hozzáférés a viszony réteg szolgáltatásaihoz

Az alkalmazási réteg, mint a legfelső (hetedik) réteg feladata az alkalmazások hozzáférése az OSI rétegekhez, továbbá olyan szolgáltatások biztosítása, melyek közvetlenül elérhetőek az alkalmazásból. Egy alkalmazás minden kapcsolati eleme egy-egy alkalmazás-entitás, melyek alkalmazási protokollokat és megjelenítési szolgáltatásokat használnak az információ megosztásához.

Minden egyes alkalmazás adatstruktúrája ASN.1-ben specifikált APDV-ként továbbítódik. Valamennyi esetben, amikor egy alkalmazás adatot kíván küldeni, biztosítja a megfelelő APDV-t; és annak ASN.1 nevét a megjelenítési réteg számára. A megjelenítési réteg ismeri az ASN.1 definícióra vonatkozó adatkomponensek típusát és méretét, valamint kódolásuk, illetve dekódolásuk menetét a továbbításhoz. A túloldalon a megjelenítési réteg analizálja a várt adatstruktúra ASN.1 azonosítóját, miután már tudja, hogy hány bit tartozik az első komponenshez, hány a másodikhoz, etc... Ezzel az információval a megjelenítési réteg végrehajthatja a szükséges konverziókat, hogy biztosítani tudja az adatot a fogadó gép belső felépítésének figyelembe vételével.

Az OSI alkalmazások által használt ASN.1 reprezentáció egyedüli, mióta az ITU javasolta, hogy az összes adatcsere az alkalmazási és a megjelenítési réteg között ASN.1 absztrakt szintaxissal legyen megadva. Az alkalmazási réteg számára azért is szükséges egy ilyen erős és strukturált jelölés, mint az ASN.1, mert itt már nem lehetséges a bitek bájtokban való gyűjtése, mint az alacsonyabb rétegekben. Ezenkívül nem várható el az alkalmazás-fejlesztőktől sem, hogy tökéletesen tudatában legyenek a problémáknak, melyekkel csak akkor találkoznak, ha az üzeneteket bitekké kódolják.

Rövidítések

APDV	Application Protocol Data Value
ASCII	American Standard Code for Information Interchange
ASN.1	Abstract Syntax Notation 1
BER	Basic Encoding Rules
CER	Canonical Encoding Rules
DER	Distinguished Encoding Rules
EBCDIC	Extended Binary Coded Decimal Interchange Code
EDGE	Enhanced Data rates for Global Evolution
FDT	Formal Description Techniques
GGSN	Gateway GPRS Support Node
GSN	GPRS Support Node
GTP	GPRS Tunnelling Protocol
ISO	International Standards Organization
ITU-T	ITU, Telecommunication Standardization Sector
MMS	Multimedia Messaging Service
MS	Mobile Station
OSI	Open System Interconnection
PER	Packed Encoding Rules
SDL	Specification and Description Language
SGSN	Serving GPRS Support Node
TCP	Transmission Control Protocol
TTCN	Testing and Test Control Notation
UDP	User Datagram Protocol
UML	Unified Modelling Language
VoIP	Voice over IP
WAP	Wireless Application Protocol
XER	XML Encoding Rules

4. Az ASN.1 szintaxis és jelölésrendszere

Az ASN.1 fő jellemzője, hogy az adatok típusokba vannak sorolva. A típus egy olyan nem üres halmaz, melyet továbbítás előtt kódolhatunk. Az ASN.1 típusoknak [1] a továbbítás miatt speciálisnak kell lenniük, és biztosítaniuk kell a megfelelő funkcionalitásokat.

A főbb ASN.1 típusok a következők: BOOLEAN, NULL, INTEGER, REAL, ENUMERATED, BIT STRING, OCTET STRING, *...String [6], CHOICE, SEQUENCE, SET, SEQUENCE OF, SET OF. Ezen típusok használatával összetett típusokat is készíthetünk.

Amikor egy típust definiálunk, valamilyen nevet kell adnunk neki, hogy hivatkozhassunk rá. A név nagybetűvel kezdődik. Minden ASN.1 hivatkozást a '::=' szimbólum segítségével hozunk létre:

```
Hazas ::= BOOLEAN
```

Az ASN.1 sorok végén nincs pontosvessző.

A SET, SEQUENCE és CHOICE összetett típusok egyes elemei mind egyedi azonosítóval rendelkeznek, mely kisbetűvel kezdődik. Ezen azonosítók segítségével a specifikáció sokkal átláthatóbbá válik és könnyebben olvasható, kezelhető lesz, azonban az adatátvitel során ezek az azonosítók nem továbbítódnak. Így abból a célból, hogy a fogadó gép informálva legyen az értékek típusáról, és hogy az adatot megfelelően tudjuk dekódolni, a továbbító gép kódolója hozzárendel az azonosítóhoz egy 'tag'-et (cédulát). A kódoló alapértelmezés szerint egy 'universal' nevű 'tag'-et használ. Van azonban, amikor az alapértelmezett eset nem elegendő a félreérthetőségek elkerüléséhez, ilyenkor szükséges a „cédulák" határozott jelölése a létrehozandó típusokban a komponensek előtt. A 'tag' (cédula) egy szám szögletes zárójelben, a típus előtt:

```
Koordinatak ::= SET {
          x [ 1]  INTEGER,
          y [ 2]  INTEGER,
          z [ 3]  INTEGER OPTIONAL
}
```

Az ASN.1 megengedi a rekurzív típusok létrehozását is, amennyiben van olyan eleme a rekurzív típusnak, amely véges értékeket tartalmaz:

```
Lottoszam ::= INTEGER (1..49)
Lottohuzas ::= SEQUENCE SIZE (6) OF Lottoszam
```

Amennyiben már megírtuk az egyes ASN.1 jelöléseinket, csak össze kell gyűjtenünk azokat és egy közös specifikációban egyesítenünk, mely leírja az adatátvitel szabályait.

Ez szabálycsoport tulajdonképpen egy protokoll specifikációjának tekinthető. Egy adott specifikáció egy vagy több ASN.1 modult tartalmazhat, ahol minden egyes modul egybefogja a típusokat, értékeket, osztályokat. A modulnevek nagybetűvel kezdődnek, és BEGIN és END kulcsszavak közé fogják a modulban definiált típusokat:

```
Module DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
...
END
```

Az AUTOMATIC TAGS azt jelenti, hogy a specifikáció készítőjének nem kell foglalkozni a szögletes zárójelekbe helyezett 'tag'-ekkel, mert azok automatikusan létrejönnek a fordító által.

5. Az ASN.1 kódolási szabályai

Basic Encoding Rules (BER)

A BER [1] kódolás formátuma minden esetben egy TLV hármas, ahol az egyes elemek jelentése:

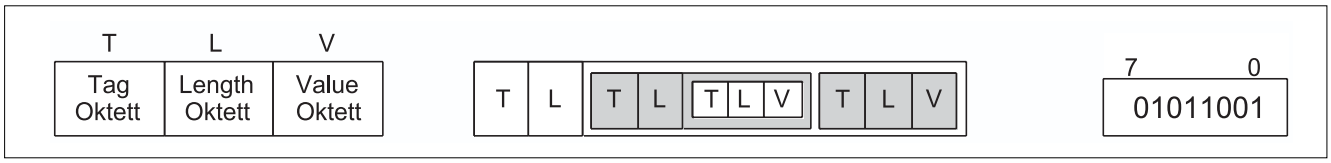
T – type/tag, L – length, V – value.

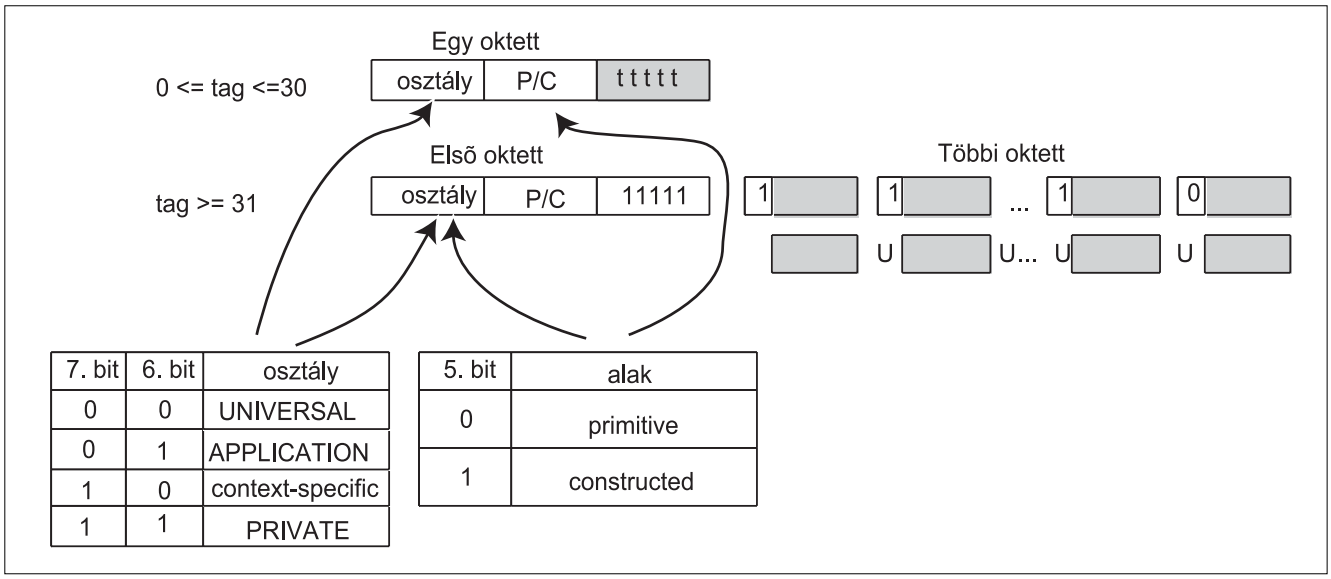
Mindegyik mező oktettek sorozata. Maga a V érték lehet egy új TLV hármas is. A BER kódolás 'Big Endian' kódolás, ugyanis a legmagasabb helyi értékű bit a bal oldalon található. A 'tag' oktettek (általában egy oktett elegendő) megfelelnek az értéktípus kódolt címkéjének. Ha a 'tag' mező hossza kisebb, mint 30, akkor az osztályok és számok kódolt hossza egy oktett lesz. Ha a 'tag' hosszabb 30-nál, akkor a szám a 6-0. sorszámú – ne felejtsük el, hogy itt a bitek sorszámozása 'Big Endian' módszerrel történik, melyet a *4. ábrán* is láthatunk – bitek összefűzéséből épül fel minden oktettben, kivéve az elsőt, ahol az alsó 5 bit mindegyike 1-es értékű lesz. Az utolsó oktettet kivéve mindegyikben a 7. számú bit értéke mindig 1. Az első T mező 5. számú bitje határozza meg, hogy a V csak értéket (primitive) vagy másik TVL hármast (constructed) tartalmaz.

Az L mező tartalmazza az aktuálisan kódolt érték (V) hosszát. Amennyiben az első T mező 5. bitje 'primitive' kódolási formát jelez, az L mezőt határozott alakban kódolja, ellenben ha az 5. bit 'constructed' formát jelez, az L mező kódolási formátumát a küldő fél választhatja meg, hogy határozott vagy határozatlan formában történjen.

A határozott alak lehet rövid (ha az L mező 127-nél kisebb), és lehet hosszú, a küldő döntésétől függően. Ez a szabadság megengedi, hogy a protokoll réteg egy bizonyos számú oktetten kódolja az összes L mezőt, két gép közötti specifikus kommunikációnál.

4. ábra Balról jobbra: A TLV szekvencia primitive és constructed esetben és a 'Big Endian' bitsorrend





5. ábra A T mező két lehetséges formátuma

A hosszú formában az L rész első oktettje a length mező hosszát reprezentálja.

A határozatlan forma kódolása olyan esetekben szükséges, amikor nem a teljes tartalmi rész ismert a küldő számára, így annak hossza nem állapítható meg a kódolás előtt. Ezenfelül másik előnye a határozatlan formának, hogy megvéd minket az értékek kétszeres vizsgálatától – mely először a hossz megállapításánál, majd a tényleges adatkódolásnál történik –, így hatékonyabb kódolókat készíthetünk. Ha az érték határozatlan alakban van kódolva, két zéró oktett zárja le a kódolt adatot. Ez a két utolsó oktett valójában egy TLV hármas, amely egy [UNIVERSAL 0]-val címkézett 0 hosszúságú értéket jelképez.

A BER kódolás architektúra független, hiszen erre az architektúrákban a bitsorrend adott és a kódolási szabályok könnyen konvertálhatóak.

Canonical and Distinguished Encoding Rules (CER/DER)

Az olyan kódolási szabályt, amely semmilyen szabadsági fokot nem hagy, kanonikus kódolási szabálynak nevezzük. A BER-ből két kanonikus kódolási szabályt származtattak, a CER-t [1] és a DER-t [2], amelyek tulajdonképpen a BER specializációi. Ez azt jelenti, hogy egy CER-rel vagy DER-rel kódolt szöveget egy BER dekódolóval tudunk dekódolni. Természetesen ez a másik irányba nem működik.

A két kódolási szabály egy érdekes tulajdonságot, az absztrakt értékek és kódolásuk közötti kétirányúságot adja kezünkbe, aminek segítségével bármely ASN.1 absztrakt értékhez egy oktett stringet tudunk rendelni, és fordítva. Bármely oktett stringhez létezik egy hozzátartozó absztrakt érték.

Ezzel a tulajdonsággal a fogadó alkalmazás összehasonlíthatja a fogadott oktett stringet egy megadott oktett stringgel anélkül, hogy tudná az értéket, amihez az valójában hozzárendelhető.

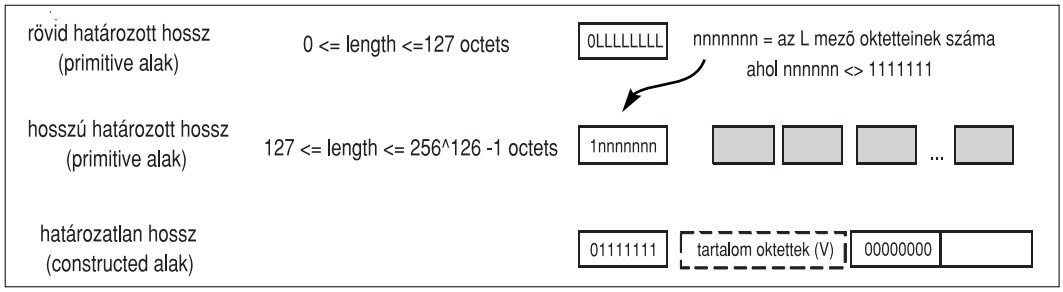
A kulcsfontosságú különbség a két szabály között az, hogy a CER a 'constructed' alaknál határozatlan, míg a DER határozott alakot használ. Emiatt a CER kódolást olyan alkalmazásoknál használják, amelyeknek nagy mennyiségű adatot kell továbbítani.

XML Encoding Rules (XER)

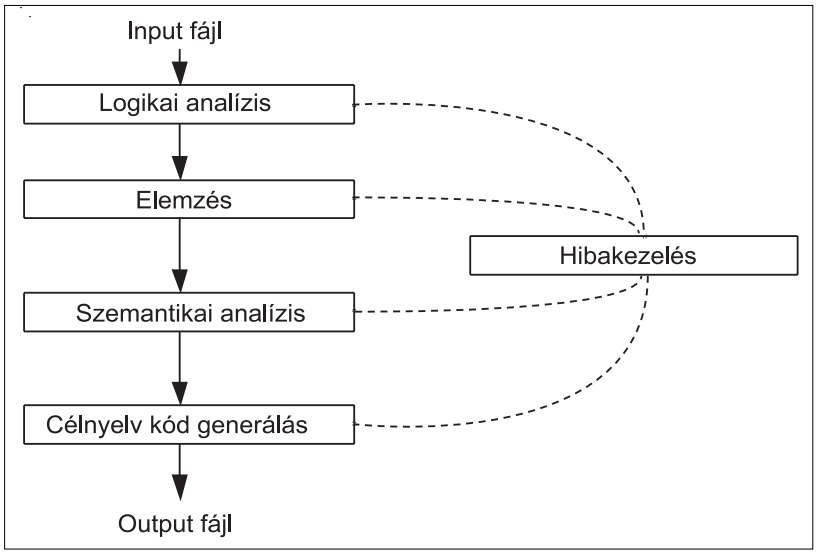
A XER kódolási szabály [1] lényege, hogy az ASN.1 értékeket XML nyelvre kódolja át. Az alapvető ötlet, hogy határoljuk az ASN.1 elemeket a következő XML címkékkel: <MARK> ... </MARK>.

Ez azt jelenti, hogy egy típus értékei a következőképpen kódolhatók:

```
PDU ::= SEQUENCE {  komponens1 SEQUENCE OF T,
                    komponens2 U }

<KOMPONENS1>...</KOMPONENS1>
<KOMPONENS1>...</KOMPONENS1>
<KOMPONENS1>...</KOMPONENS1>
<KOMPONENS2>...</KOMPONENS2>
<KOMPONENS2>...</KOMPONENS2>
```



6. ábra Az L mező lehetséges formái



7. ábra
A fordítás lépései, egy idealizált fordító felépítése

6. Az ASN.1 fordító

Általánosságban a fordító egy olyan számítási eszköz, amely beolvas egy programot mely első nyelven, a forrásnyelven íródott, és lefordítja azt egy második nyelvre, amely a célnyelv, és tulajdonságait tekintve már adott gép architektúrájának megfelelő ábrázolási módot követ. Természetesen minden egyes architektúrához külön, az ahhoz készített fordító szükséges. A mi esetünkben a forrásnyelv az ASN.1, a célnyelv pedig lehet C, C++ és Java, a program pedig egy specifikáció, mely néhány modulból épül fel.

Egy idealizált fordító négy rétegre bontható, melyek mindegyike csak akkor működik, ha a fölötte lévő réteg hibátlanul fejezte be működését. Az elemzési és lexikai hibákat a forráskódban lévő meg nem engedett karakterek, vagy grammatikai struktúrák gerjesztik, míg a szemantikai hibákat az inkoherens specifikációk idézik elő (például egy INTEGER hozzárendelése BOOLEAN-ként deklarált értékhez).

Amennyiben a kód nem tartalmaz sem szintaktikai, sem szemantikai hibákat, a fordító általában a következő fájlokat generálja:

– egy fájlt a konkrét szintaxissal, amely az ASN.1 specifikációban szereplő adattípusok fordítása a megfelelő célnyelvre,

– egy vagy több fájlt, amely tartalmaz egy kódoló és egy dekódoló eljárást az ASN.1 specifikációból minden egyes típusra, amely megvalósítja a kódolási szabályokat, valamint generálják az átviteli szintaxist.

7. A GTP és útprotokollja

Egy teljes ASN.1 specifikáció létrehozását a GTP és annak útprotokollja segítségével próbálunk meg bemutatni, azonban ehhez szükség van a GTP protokoll, és az azt magában foglaló GPRS rövid ismertetésére is. A specifikáció felépítése során a kiindulópont a GPRS hálózat megismerése, ami után már fel tudjuk építeni a függelékben található ASN.1 specifikációt.

A cikkünkben bemutatott példát úgy próbáltuk meg kiválasztani, hogy mindenképen egy viszonylag új technológiát vizsgáljunk meg és ez a mobil távközlés területén használatos rendszer legyen. Így került a GPRS rendszerre és a GTP protokollra a választás.

Mivel a GPRS nemcsak a ma még jóval elterjedtebb GSM, hanem a közeljövőben egyre inkább teret hódító EDGE hálózatokon is használható, érdemes ennek fokozott figyelmet szentelnünk. Napjainkban a legelterjedtebb alkalmazások kapcsán is előtérben van ez a szolgáltatás, hiszen GPRS-t használhatunk WAP és Internet oldalak böngészésekor, vagy MMS üzenetek küldése során. Éppen ezért a GPRS mérföldkőnek számít a GSM hálózatok fejlődésében, a ma rendelkezésre álló hálózati infrastruktúrán, – és ez az az ok, ami miatt ezzel a rendszerrel és protokollal foglalkozunk.

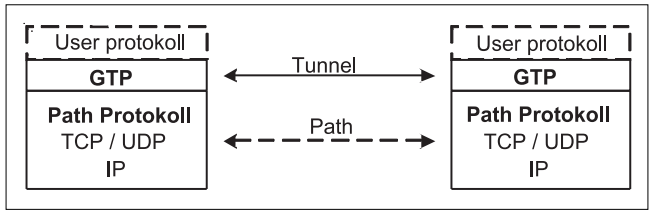
Egy GPRS hálózatban több GSN található, amelyek IP protokollon keresztül tartják egymással a kapcsolatot. Ezek lehetnek GGSN-ek vagy SGSN-ek. Az SGSN kommunikál a mobil készülékkel, a GGSN az átjáró az Internetre. Így, amennyiben egy mobil készülékről például egy www. oldalt böngészünk, az adatok útvonala rendre a következő lesz:

MS ⇒ SGSN ⇒ GGSN ⇒ webszerver.

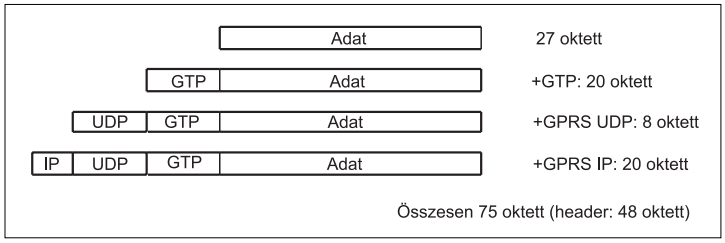
Ez a virtuális adatútvonal.

A GTP alagút [10] ebben a hálózatban tulajdonképpen két GSN között található meg. A két GSN (általában a GGSN és az SGSN) egy virtuális kapcsolaton keresztül, a GTP alagúton tartja egymással a kapcsolatot, amely a *8. ábrán* is látható.

8. ábra A GTP és a GTP útprotokoll szerkezete



9. ábra
Adatcsomaghoz a GTP útprotokoll által hozzáadott fejrészek



A GTP az adatcsomagokat a *9. ábrán* látható módon továbbítja, először egy GTP fejrészt tesz az adat elé, majd ezt egy TCP/UDP [5], végül pedig egy IP [5] fejrésszel egészíti ki. A másik oldalon rendre lebontja ezeket a fejrészeket, és megkapja a szükséges adatot, melyet tovább küldhet a mobil készülék vagy az Internet irányába. Ezt a folyamatot végzi GTP útprotokoll, melynek adatszerkezete a függelék részben az ASN.1 specifikációban található meg, ahol látható a fejrészek elhelyezkedése, illetve az, amint az egyes fejrészek után következő csomagrész újabb fejrészeket tartalmaz, ahogy a GTP csomag egy TCP vagy egy UDP csomagot.

8. Összefoglalás

A formális leíró technikák meglehetősen fontos szerepet töltenek be a protokoll-tervezésben. Ezért nagyon fontos az, hogy biztosítsuk a gyors átjárhatóságot az egyes FDT-k között. Itt kerül a képbe az ASN.1, amely legfőbb tulajdonságának – a hardverfüggetlenségnek – köszönhetően tökéletesen alkalmas erre a feladatra. Úgy is mondhatnánk, hogy az összekötő kapocs szerepét tölti be a formális technikák, az UML, az SDL és a TTCN között.

Az ASN.1 segítségével teljes protokoll-specifikációkat tervezhetünk és magas szintű alkalmazásspecifikációkat írhatunk le. Mindenképpen szükségünk lesz erre a protokolltervezés egyes lépései között, azonban két legfőbb tulajdonsága – olyan érthető, mint amilyen absztrakt – megszabadít minket minden korláttól, mely akadályozhatja tervezői tevékenységünket.

Irodalom

[1] Olivier Dubuission:
ASN.1 – Communication between heterogeneous systems
Morgan Kaufmann Publishers, 2000.

[2] Prof. John Larmouth:
ASN.1 Complete
Morgan Kaufmann Publishers, 1999.

[3] J. Ellsberger, D. Hogrefe, A. Sarma:
SDL Formal Object-oriented Languages for Communicating Systems,
Prentice Hal Europe, 1997.

[4] OSI – A Model for Computer Communications Standards
U. Black, Prentice-Hall, 1994.

[5] Andrew S. Tanenbaum:
Számítógép-hálózatok
Panem-Prentice Hall, 1999.

SERIES X: DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS,
OSI networking and system aspects – Abstract Syntax Notation One (ASN.1)

[6] Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation
ITU-T, Rec. X.680, 07/2002.

[7] Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)
ITU-T, Rec. X.690, 06/1999.

[8] Information technology – ASN.1 encoding rules: Specification of Packed Encoding Rules (PER)
ITU-T, Rec. X.691, 06/1999.

[9] Information technology – ASN.1 encoding rules: XML Encoding Rules (XER)
ITU-T, Rec. X.693, 12/2001.

[10] Jyke Jokinen:
GPRS & UMTS Protocols: GTP details,
Tampere University of Technology,
Department of Information Technology,
Advanced Topics in Telecommunications, 2000.
www.cs.tut.fi/kurssit/8309700/reports/gtp-report.pdf

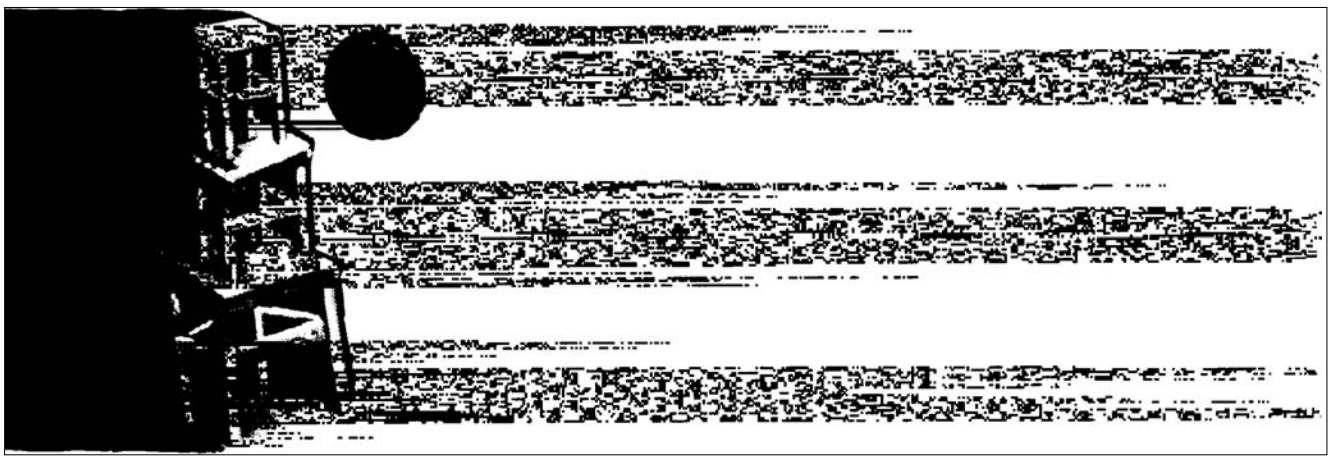
Függelék

A GTP útprotokoll ASN.1 specifikációja:

```
Module-packets DEFINITIONS
AUTOMATIC TAGS ::=
  BEGIN

Ip ::= SEQUENCE {
  ip-header     SEQUENCE {
    ip-version  INTEGER,
    ip-header-length  INTEGER,
    type-of-service   SEQUENCE {
          precedence  BIT STRING,
          delay       BIT STRING,
          throughput  BIT STRING,
          reliability BIT STRING,
          unused      BIT STRING
    },
    full-length       INTEGER,
    identification  OCTET STRING,
    unused      BIT STRING,
    dont-fragment     BIT STRING,
    more-fragment     BIT STRING,
    fragment-offset   BIT STRING,

    life-time   INTEGER,
    protocol-type     BIT STRING,
    header-checksum   BIT STRING,

    source-address OCTET STRING,
    destination-address OCTET STRING,

    options     OCTET STRING
    },
```

```
   ip-packet SEQUENCE {
   }
}

Tcp ::= SEQUENCE {
   tcp-header     SEQUENCE {
      source-port OCTET STRING,
      destination-port  OCTET STRING,
      sequence-number   OCTET STRING,
      acknowledgement-number  OCTET
                              STRING,
      tcp-header-length INTEGER,
      unused       BIT STRING,
      urg-bit      BIT STRING,
      ack-bit      BIT STRING,
      rst-bit      BIT STRING,
      psh-bit      BIT STRING,
      syn-bit      BIT STRING,
      fin-bit      BIT STRING,
      window-size        INTEGER,
      checksum     OCTET STRING,
      urgent       OCTET STRING,

      options      OCTET STRING
      },
   tcp-packet     SEQUENCE {
      ip-packet    Ip
   }
}

Udp ::= SEQUENCE {
   udp-header     SEQUENCE {
source-port       OCTET STRING,
destination-port OCTET STRING,
udp-segment-length INTEGER,
udp-checksum      OCTET STRING
},
   udp-packet     SEQUENCE {
ip-packet  Ip
   }
}

END
```

```
Protocol DEFINITIONS AUTOMATIC
TAGS ::=
     BEGIN
IMPORTS Ip, Tcp, Udp
FROM Module-packets;

PDU ::= CHOICE {
  gtp-header SEQUENCE {
     gtp-version        INTEGER
                        DEFAULT 0,
     pt                 INTEGER
                        DEFAULT 1,
     spare              BIT STRING
                        DEFAULT
                        '111'B,
     snn                INTEGER,
     message-type       OCTET
                        STRING,
     length             OCTET
                        STRING,
     sequence-number    OCTET
                        STRING,
     flow-label  OCTET STRING,
     sndcp-n-pdullc-number   OCTET
                        STRING,
     spare1             BIT STRING
           DEFAULT '11111111'B,
     spare2             BIT STRING
           DEFAULT '11111111'B,
     spare3             BIT STRING
           DEFAULT '11111111'B,
     tid          OCTET STRING}
  },
   gtp-packet CHOICE {
     tcp-packet  Tcp,
     udp-packet  Udp
   }
}

END
```



Automatikus tesztgenerálás formális protokollspecifikáció alapján

VINCZE GÁBOR

Budapesti Műszaki és Gazdaságtudományi Egyetem, Távközlési és Médiainformatikai Tanszék

vincze@alpha.ttt.bme.hu

Reviewed

Kulcsszavak: konformancia tesztelés, tesztgenerálás, mutáció-analízis, evolúciós algoritmusok, bakteriális algoritmus

A cikkben egy eljárást mutatunk be automatikus tesztgenerálásra a protokoll formális SDL specifikációja alapján. A protokolltesztelés a fejlesztési folyamat fontos része, ám a tesztkészletek kialakítása időigényes feladat. Ennek a fázisnak az automatizálása csökkenti a bevezetési időt, és egy komoly hibaforrást szüntet meg. Megmutatjuk, hogyan használható a mutációanalízis egy állapottér-bejáró algoritmusból eredő tesztesetek, és a tesztkritériumok megfeleltetésének. Ezek után evolúciós algoritmusokat alkalmazunk egy optimális részhalmaz kiválasztására ebből a kezdeti tesztkészlet-halmazból. Ezeket az eljárásokat felhasználva egy teljes tesztgenerációs folyamatot építünk fel, amellyel egy protokoll formális specifikácójából tesztkészleteket kapunk.

1. Bevezetés

Ahogy a távközlési cégeknek egyre több szolgáltatást kellett nyújtaniuk, miközben hálózataik integrálására törekedtek, úgy nőtt a távközlési protokollok komplexitása. Ezzel egyidejűleg ezeknek a hálózatoknak egyre növekvő megbízhatósági követelményeknek kellett megfelelniük. Ezzel a komplexitás-növekedéssel a protokollok specifikációjához szükséges erőfeszítés súlyos teherré vált, és a megbízhatóság, valamint a gyártók termékeinek együttműködése iránti igény átfogóbb tesztelést tett szükségessé. Ezek a problémák hívták életre a formális specifikációs eljárásokat, valamint a formális tesztelési eljárásokat, amelyekkel ellenőrizni lehet, hogy egy alkalmazás a specifikációnak megfelelően működik-e.

A távközlési világban legelterjedtebben használt formális nyelvek a Specifikációs és Leíró Nyelv (Specification and Description Language, SDL [1]) a rendszerek specifikálására, amely a rendszert párhuzamosan működő kommunikáló véges automatákkal modellezi, és a Fa és Táblás Kombinált Jelölésmód (Tree and Tabular Combined Notation, TTCN [2]) a rendszerek fekete doboz jellegű ellenőrzésére.

Ma már a rendelkezésre állnak nagymértékben integrált és széles körben elterjedt fejlesztőeszközök [3], hogy segítsék a fejlesztőket a specifikációs és a vizsgálati folyamat során. Ennek ellenére a formális tesztkészletek előállítása még mindig jelentős munkát igényel, és az emberi tényező továbbra is a legdrágább, és legtöbb hiba forrása. Mivel a tesztkészleteket sokszor több százszor vagy ezerszer kell lefuttatni, a futási idő és a hardverkövetelmények szintén kulcsfontosságúak.

Ebben a cikkben bemutatunk egy módszert az automatikus tesztgenerálásra a rendszer SDL leírásából. Ennek a tesztgenerációs folyamatnak négy fő lépése van:

1) formális specifikálás SDL nyelven
2) teszteset-halmaz előállítása egy állapottér-bejáró algoritmussal
3) mutáció-analízis
4) egy optimális teszteset-részhalmaz kiválasztása

Először bemutatjuk a mutáció-analízis eljárást; ezek után megmutatjuk, hogyan alkalmazunk evolúciós algoritmusokat egy optimális teszteset-részhalmaz kiválasztására, majd végül bemutatjuk a teljes tesztgenerációs folyamatot az INRES protokoll példáján.

2. Mutáció-analízis

2.1. Áttekintés

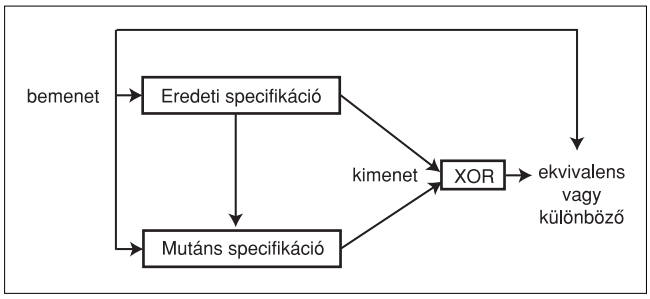
A mutáció-analízis egy fehér doboz módszer tesztesetek kialakítására, azaz a rendszer belső logikájának ismeretén alapul. A hagyományos mutáció-analízist a programkódokban található hibák felderítésére dolgozták ki, ám a mi esetünkben programok helyett specifikációkra alkalmazzuk, a megfelelő fekete doboz tesztesetek kiválasztásához.

Egy mutáció-analízis rendszerben definiálni kell egy mutációs operátor készletet [4], ahol minden operátor egy atomi szintaktikai változást testesít meg. Ezeknek az operátoroknak az alkalmazása két okból praktikus. Egyrészt lehetővé teszik a hibatípusok formális leírását, másrészt lehetővé teszik a mutánsok automatikus generálását. Az operátorokat szisztematikusan alkalmazva a specifikációra egy mutánskészletet generálhatunk.

Egy mutáció-analízis rendszer 3 komponensből áll:

– az eredeti rendszer,
– a mutáns rendszer – az eredeti rendszerhez képest egy apró szintaktikai változást tartalmaz. A mutánsokat a mutációs operátorok alkalmazásával kapjuk, ahol minden operátor egy apró szintaktikai változást testesít meg,

– orákulum – egy ember, vagy a legtöbb esetben egy gép, amely megkülönbözteti az eredeti rendszert a mutánstól a környezettel való interakciói alapján.



1. ábra Mutáció-analízis

Abból a feltételezésből indulunk ki, hogy a véges automatát megalkotó olyan specifikációt készít, amely közel áll az elvárásokhoz, és ezért azok a tesztesetek, amelyek felfedik a specifikáció szintaktikai változásait hasznosak. Csak elsőrendű hibákat idézünk elő, tehát egyszerre csak egy mutációt alkalmazunk, mert azok a tesztesetek, amelyek az egyszerű változásokat detektálják, az egyszerű változások sorozataként előállított komplex változásokat is detektálják [5].

A tesztesetek akkor különböztetik meg a mutánst az eredetitől, ha az más kimenetet ad. De az operátorok által generált mutánsok egy része szemantikailag ekvivalens lehet az eredeti rendszerrel, azaz a mutáns és az eredeti rendszer pontosan ugyanazt a kimenetet adná minden lehetséges bemenetre. Ezeket a mutánsokat *ekvivalensnek* nevezzük. Az olyan rendszereket, amelyek ugyanazt a kimenetet adják minden bemenetre, mint az eredeti rendszer, de szemantikusan nem ekvivalensek azzal, *pszeudo-ekvivalenseknek* nevezzük (az ekvivalens mutánsok a pszeudo-ekvivalens mutánsok egy részhalmaza). A teszteseteknél minden ekvivalenst figyelmen kívül kellene hagyjunk, és minden nem-ekvivalenst figyelembe kellene vennünk. Ez komoly problémát okoz a mutáció analízis rendszereknél, mivel általában nem lehetséges az ekvivalensek automatikus identifikálása, és az ekvivalensek és nem-ekvivalensek megkülönböztetése emberi közreműködést igényel.

2.2. Mutációs operátorok

A mutációs operátorok kialakításánál nagyon fontos szempont, hogy amennyiben lehetséges, ne adjanak egyetlen pszeudo-ekvivalenst se, és természetesen minimalizálják az ekvivalensek számát. Az operátorok kialakításának alapelvei:

– az operátorok atomi hibákat hivatottak modellezni;
– csak elsőrendű,
– csak szintaktikailag helyes;
– és csak szemantikusan helyes mutánsokat szeretnénk generálni;
– az operátorok véges, és a lehető legkisebb számú mutánst generálják.

Öt operátor osztályt van definiálva [4] a kommunikáló kiterjesztett véges automatákhoz, attól függően, hogy az automata mely részét módosítják: állapot-, bemenet-, kimenet-, cselekvés- és predikátum-módosító operátorok.

Minden osztálynál három típusú operátort adhatunk meg, attól függően, hogy milyen jellegű hibát reprezentálnak: növelő, csökkentő és cserélő operátorok.

2.3. Teszteset – tesztkritérium megfeleltetés

A következő algoritmus segítségével egy véges méretű, strukturálatlan, és nagymértékben redundáns tesztkészlet (amelyet például egy a rendszerspecifikáció állapotterét bejáró állapottér-bejáró algoritmussal kaphatunk) minden egyes tesztesetéhez hozzárendelhetünk egy tesztkritérium-halmazt. Ha mutációs operátorokat alkalmazunk a nem megfelelő bemenetek megfigyelésére, ennek a kezdeti tesztkészletnek szintén tartalmaznia kell nem megfelelő teszteseteket.

Legyen C egy kétdimenziós, boole-algebrai értékeket tartalmazó mátrix.

0) Generáljunk egy teszteset-halmazt;
1) Alkalmazzuk egy mutációs operátort a véges automatára, hogy létrehozzuk az *i.* mutánst;
2) Futtassuk le az összes tesztesetet a mutáns specifikáción, és figyeljük meg az inkonzisztenciákat: amennyiben a teszteset az eredeti specifikációtól eltérő eredményt ad, a teszteset detektálja az adott mutánst
3) Hozzuk létre a *Ci* oszlopvektort (C mátrix *i.* oszlopát)
   – legyen *Ci[j] = 0*
     ha a *j.* teszteset nem detektálja az *i.* mutánst;
   – legyen *Ci[j] = 1*
     ha a *j.* teszteset detektálja az *i.* mutánst;
4) Ismételjük a 2-4. lépéseket, ahol *i* 1-től N-ig vesz fel értékeket, amíg létre nem hoztuk az összes lehetséges mutánst;
5) Nyerjük ki a C kritériummátrixot, ahol a sorok az eredeti halmaz teszteseteit ábrázolják, az oszlopok pedig a mutánsokat.

3. Tesztszelekció evolúciós algoritmusokkal

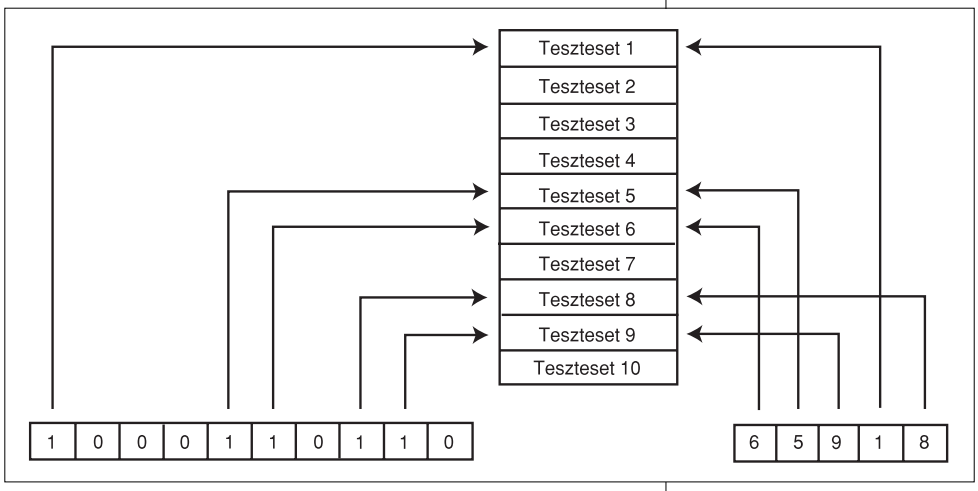
A szelekciós folyamat célja, hogy a tesztesetek egy optimális részhalmazát kapjuk a már meglévő strukturálatlan, és nagymértékben redundáns halmazból. Erre a célra három különböző „puha” algoritmust alkalmaztunk: a Genetikus Algoritmust (GA), a Pszeudo-Bakteriális Genetikus Algoritmust (PBGA), és a Bakteriális Evolúciós Algoritmust (BEA).

Az evolúciós algoritmusokra azért esett a választás, mert jó eredményeket adnak elfogadható időn belül, képesek az igen bonyolult esetek kezelésére is, és könnyen integrálhatóak a tesztgenerációs folyamatba [6].

3.1. Általános megfontolások

Egyedek: Egy *egyed* a probléma egy lehetséges megoldása, a mi esetünkben egy optimalizált tesztkészlet. Két lehetőségünk volt az egyedek ábrázolásá-

ra: egy fix hosszúságú, N bitből álló sorozat, ahol N az eredeti halmaz összes tesztkészletének száma, és egy bit értéke 1, ha az adott teszteset szerepel a tesztkészletben. Ezeket az egyedeket *bitsorozat* egyedeknek neveztük el. A másik megoldás egy változó méretű, 1 és N közötti értékeket tartalmazó halmaz, amelyben minden elem az eredeti halmaz egy tesztesetét ábrázolja. Ezeket az egyedeket *mutató-halmaz* egyedeknek neveztük el. Az utóbbi esetben természetesen lehetséges, hogy egy tesztkészlet többször tartalmazza ugyanazt a tesztesetet, ám ezek az egyedek magasabb futtatási költséggel rendelkeznek bármiféle egyéb érték nélkül, így hamar kiesnek a szelekció során. Az algoritmustól függően egyik vagy mindkét ábrázolási módot alkalmaztuk.



2. ábra Bitsorozat és mutató-halmaz egyedek

Teszteset költsége: a vizsgálati költség az adott teszteset futtatási költségét reprezentálja, ami jelenthet végrehajtási időt, vagy hardverkövetelményeket. Legyen T = {t1,t2,…,tn} a t1,t2,…,tn teszteseteket tartalmazó készlet, és R = {r1,r2,…,rk} az általa lefedett tesztkövetelmények halmaza.

Ekkor minden teszteset-halmazhoz hozzárendeljük a c: T→R pozitív függvényt.

Egy adott T tesztkészlet futtatási költségét ekkor az alábbi függvény adja:

$$c(T) = \sum_{t \in T} c(t) \qquad (1)$$

Az egyéni tesztesetek futtatási költsége lehet tetszőlegesen kijelölt, vagy a mutáció-analízis fázis során megmért érték.

Itt azt feltételezzük, hogy minden tesztkövetelmény ellenőrzése bizonyos erőforrásigénnyel rendelkezik, valamint a teszteset inicializálása is erőforrásokat igényel. Így egy teszteset költségét az alábbiak szerint kapjuk meg:

$$c(t) = c1 + c2 * L \qquad (2)$$

ahol *c1* az inicializációs költség, *c2* az egyes tesztkövetelmények ellenőrzéséhez rendelhető költség, L pedig az ellenőrzött tesztkövetelmények száma.

Célfüggvény: a célfüggvény méri az egyes egyedek minőségét, ezt próbálja minimalizálni az algoritmus. A kívánt tesztkészletek eléréséhez a célfüggvénynek a következőket kell figyelembe vennie:

– A tesztkészlet futtatási költségét minimalizálni szeretnénk, a lefedett tesztkövetelmények redundanciájának minimalizálásával.
– A tesztkészlet fedje le az összes követelményt.

Célfüggvényünk az összes teszteset végrehajtási költségeinek összege, valamint egy büntető érték minden egyes lefedetlen tesztkövetelményért:

$$O = c3 * C + c4 * M \qquad (3)$$

ahol *C* az egyed költsége, *M* a lefedetlen követelmények száma, *c3* és *c4* pedig súlyozó tényezők, amelyeket úgy kell megválasztani, hogy ne legyen gazdaságos elhagyni a teszteseteket lefedetlen követelmények árán.

3.2. Genetikus algoritmus

A genetikus algoritmus egy olyan optimalizációs eljárás, amely a természetben lejátszódó szelekciós folyamatokat modellezi [7]. A kanonikus GA, amelyet itt alkalmaztunk, az alábbiak szerint működik:

Inicializálás
Kezdeti populáció létrehozása
Kezdeti populáció kiértékelése
generáció := 0

Generációs hurok
{
Fitness értékek számítása
Szelekció
Rekombináció
Mutáció
Új egyedek kiértékelése
Új egyedek visszahelyettesítése

generáció := generáció + 1
} amíg generáció < max. generáció

Az egyedek bitsorozatok, mivel a keresztezés alkalmazása sokkal intuitívabb volt így. Tekintsük át egyenként az algoritmus lépéseit:

Fitness: Az egyedek fitness-értékét a lineáris rangsor-alapú módszer alapján végeztük, ahol az *i.* egyed *Fi* fitness-értékét az alábbi képlet adja:

$$F_i = 2 - sp + 2 * (sp - 1) * \frac{pos(fi) - 1}{N_{ind} - 1} \qquad (4)$$

Ahol *sp* a szelekciós nyomás (a mi esetünkben *sp=2*), *pos(fi)* az *i.* egyed pozíciója a célfüggvény alapján, és *Nind* a populáció mérete.

Szelekció: Az egyedeket az utódok létrehozására a Sztochasztikus Univerzális Mintavételezési módszerrel választjuk ki: leképezzük az egyedeket egy számtengelyre, ahol minden egyednek a fitness-értékének megfelelő hossz jut. Ezek után generálunk egy véletlen számot az [1..szülők_száma] intervallumban, ahol *szülők_száma* az utódok létrehozására kiválasztandó egyedek száma. Ezek után ezt az értéket eltoljuk az i*(fitness-ek összege)/(szülők_száma) értékkel, ahol i ∈ [0 .. szülők_száma – 1], és minden egyes alkalommal kiválasztjuk azt az egyedet, amelyre ez az érték mutat a számtengelyen.

Rekombináció: Itt az egyenletes keresztezési módszert alkalmazzuk: generálunk egy véletlenszerű bitmintát. Ezek után úgy állítjuk elő az utódokat, hogy a szülők bitjeit felcseréljük azokban a pozíciókban, ahol ennek a maszknak az értéke 1.

Mutáció: Minden egyedet kis valószínűséggel mutálunk, hogy egy nagymértékű változásokat is lehetővé tegyünk. Egy véletlenszerű pozíciótól egy előre meghatározott hosszúságú szegmensen minden bitet *Pm* valószínűséggel mutálunk.

3.3. Pszeudo-bakteriális genetikus algoritmus

A 90-es évek második felében kifejlesztett bakteriális algoritmusok a baktériumok evolúciós folyamatait modellezik. A legegyszerűbb bakteriális algoritmus a pszeudo-bakteriális genetikus algoritmus [8].

Az algoritmus elején létrehozunk egy véletlenszerű egyedet, amelyre alkalmazzuk a bakteriális mutációt. Az eredeti egyedről *n – 1* másolatot (klónt) hozunk létre. Ezek után véletlenszerűen kiválasztjuk a kromoszóma egy részét, amelyet minden klónnál mutálunk, de változatlanul hagyjuk az eredeti egyednél. A mutáció után kiértékeljük az összes egyedet, és a legjobb egyed mutált részét átmásoljuk a többi klónba.

Ezt a mutáció-kiértékelés-szelekció-visszahelyettesítés ciklust addig ismételjük, amíg a kromoszóma összes részét nem mutáltuk. Ezek után kiválasztjuk a legjobb egyedet, a többit pedig megszüntetjük. A ciklust addig ismételjük, amíg kielégítő eredményt kapunk, vagy elérünk egy előre meghatározott generációszámot.

Ezt az algoritmust mindkét típusú egyeddel létrehoztuk. A bitsorozat típusú egyedeknél a mutáció megegyezik a GA esetén alkalmazottal. A mutató-halmaz egyedek esetében a mutációnak lehetővé kell tennie, hogy az egyed hossza megváltozzon, mivel nincsen a priori információnk az optimális egyedhosszúságról. Így a mutáció három típusú változást idézhet elő:

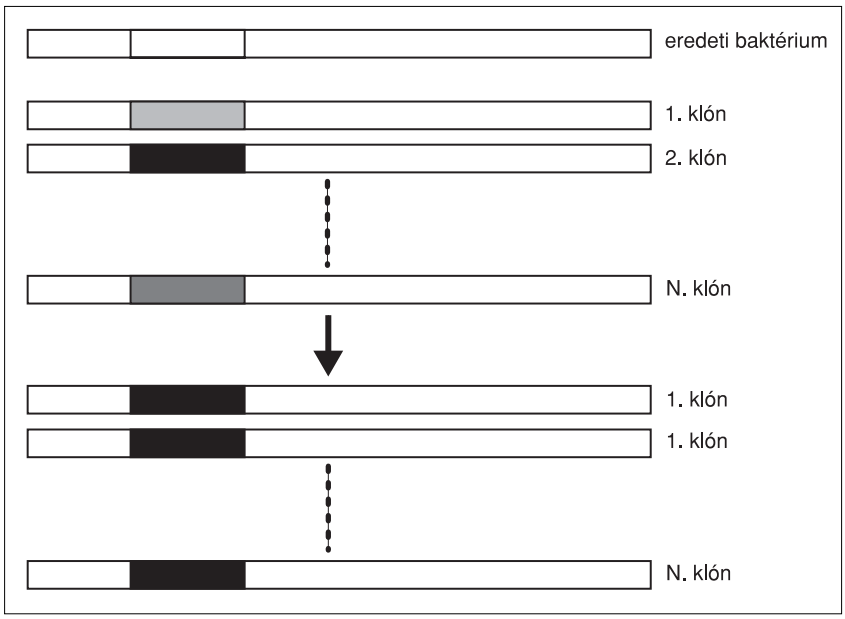
– teszteset helyettesítését egy másik tesztesettel;
– egy teszteset törlését, vagy
– egy teszteset hozzáadását.

3.4. Bakteriális evolúciós algoritmus

A bakteriális evolúciós algoritmus a PBGA egy továbbfejlesztett változata, ahol a keresést egyszerre több egyeden végezzük párhuzamosan. Ezt az algoritmust a baktérium-populációk géntranszfer képessége ihlette [9].

Az algoritmus az alábbiak szerint működik:

1) Létrehozunk egy *n* egyedből álló véletlenszerű populációt
2) Minden egyedre alkalmazzuk a bakteriális mutációt (a 3.3.-ban leírtak szerint)
3) *Ninf*-szer alkalmazzuk a géntranszfer műveletet, ahol *Ninf* az infekciók száma. Ennél a lépésnél egy alsó (rosszabb egyedek) és egy felső (jobb egyedek) félre osztjuk a populációt, és a felső félből az alsó félbe géneket helyezünk át.
4) A 2-4. lépéseket addig ismételjük, amíg kielégítő eredményt nem kaptunk, vagy elértünk egy előre definiált generációszámot.



3. ábra
A pszeudo-bakteriális genetikus algoritmus

Ennél az algoritmusnál módosítanunk kellett az egyedek felépítésén, hogy jól elhatárolt géneket tartalmazzanak, mivel a géntranszfer-műveletnél szükség van egy mérőszámra, ami azt mutatja meg, mennyire „jó” egy gén. A mutató-halmaz egyedeket egy előre meghatározott számú génre osztottuk, amelyek változó számú tesztesetet tartalmazó csoportok. A gén jóságának két különböző verzióját használtuk:

Első változat

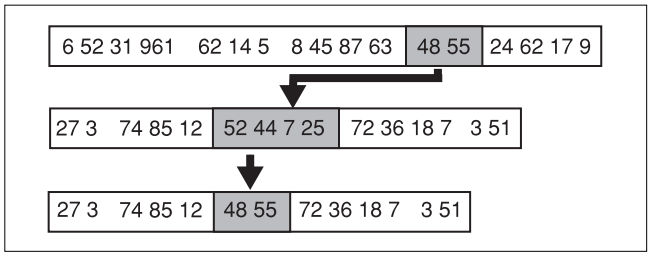
Ennél az implementációnál egy gén jóságát az határozza meg, hogy átlagosan milyen költséggel fed le egy tesztkövetelményt.

Így ezt a következőképpen számítjuk:

$$F = \frac{\sum_{i \in I} C_i}{R} \quad (5)$$

ahol *F* a gén jósága, *Ci* a tesztesetek költsége, *I* a gén teszteseteinek halmaza, és *R* a gén által lefedett tesztkövetelmények száma.

A géntranszfer-művelet során a felső fél egy baktériumából a legjobb génnel helyettesítjük az alsó fél egy baktériumának legrosszabb génjét *(4. ábra)*.



4. ábra Géntranszfer 1

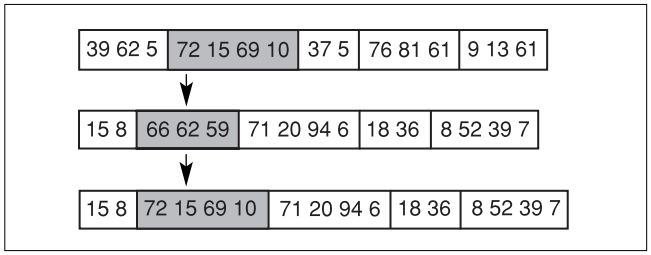
Második változat

Ennél a megközelítésnél annyi részre osztjuk a tesztkövetelményeket, ahány gént tartalmaz a baktérium. A célunk az, hogy minden gén a tesztkövetelmények egy meghatározott részét fedje le. Egy gén jóságát ugyanúgy határozzuk meg, mint a célfüggvényt az előző esetekben, de a lefedetlen tesztkövetelményeket csak a gén által lefedett intervallumon vesszük figyelembe. A gén jóságát az alábbi képlet adja *(5. ábra)*.

$$F = c1 * C + c2 * M_i \quad (5)$$

ahol *F* a gén jósága, *C* a gén költsége, *Mi* a gén által lefedett halmazon kihagyott tesztkövetelmények száma, *c1* és *c2* pedig súlyozó tényezők.

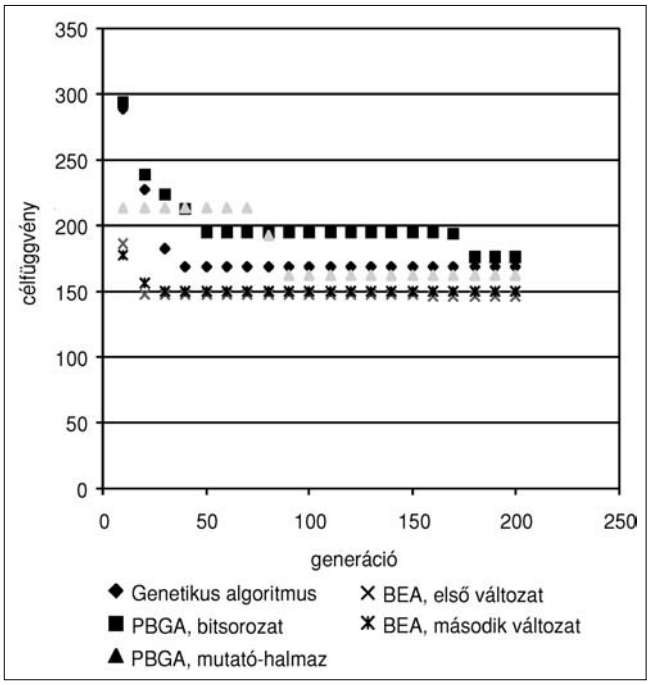
A géntranszfer során egy a felső félből vett forrásbaktériumból kiválasztunk egy véletlenszerű gént, és ha jobb az alsó félből vett célbaktérium ugyanazon pozíciójú génjénél, akkor helyettesítjük vele:



5. ábra Géntranszfer 2

3.5. Algoritmusok összehasonlítása

Hogy összehasonlíthassuk ezen algoritmusok hatékonyságát a teszteset-szelekció során, egy fiktív, 100 tesztesetet tartalmazó halmazon futtattuk őket (amint azt később látni fogjuk, az INRES protokoll kezdeti tesztkészlete csak 41 tesztesetet tartalmaz, ami túl kevés, hogy különbségek mutatkozzanak ezen algoritmusok konvergenciájában).



6. ábra Algoritmusok konvergenciája

A különböző algoritmusok konvergenciája a *6. ábrán* látható.

4. Automatikus tesztkészlet-generálás

Bemutatjuk a teljes tesztkészlet-generálási eljárást. Ezt a folyamatot a jól ismert INRES mintaprotokoll példájával illusztráljuk:

0. Létrehozunk egy formális SDL protokollspecifikációt. Erre a célra kiforrott eszközök állnak rendelkezésre [3]. A *7. ábra* mutatja az INRES protokoll SDL specifikációjának rendszer-áttekintő részét.
1. Az SDL specifikáción lefuttatunk egy állapottér-bejáró algoritmust, amely egy nagymértékben redundáns, strukturálatlan tesztkészletet eredményez.
2. A mutáció-analízis segítségével meghatározzuk a tesztkövetelmények mátrixát erre a teszteset-halmazra. Az *1. táblázat* az INRES rendszer SDL specifikációjának állapottér-bejárásából eredő 41 tesztesetből álló teljes tesztkészletét mutatja, az egyes tesztesetek költségével, valamint a teljes tesztkészlet költségével, ahol a tesztesetek költségét (2) szerint számítottuk, c1=20 és c2=5 értékekkel.
3. Kiválasztjuk a tesztesetek optimális részhalmazát a halmazból a fent bemutatott evolúciós algoritmusok egyikével. Ez egy olyan tesztkészletet eredményez, amely minimális redundanciával és végrehajtási költséggel lefedi az összes tesztkritériumot. A *2. táblázat* az INRES protokoll optimalizált tesztkészletét mutatja.

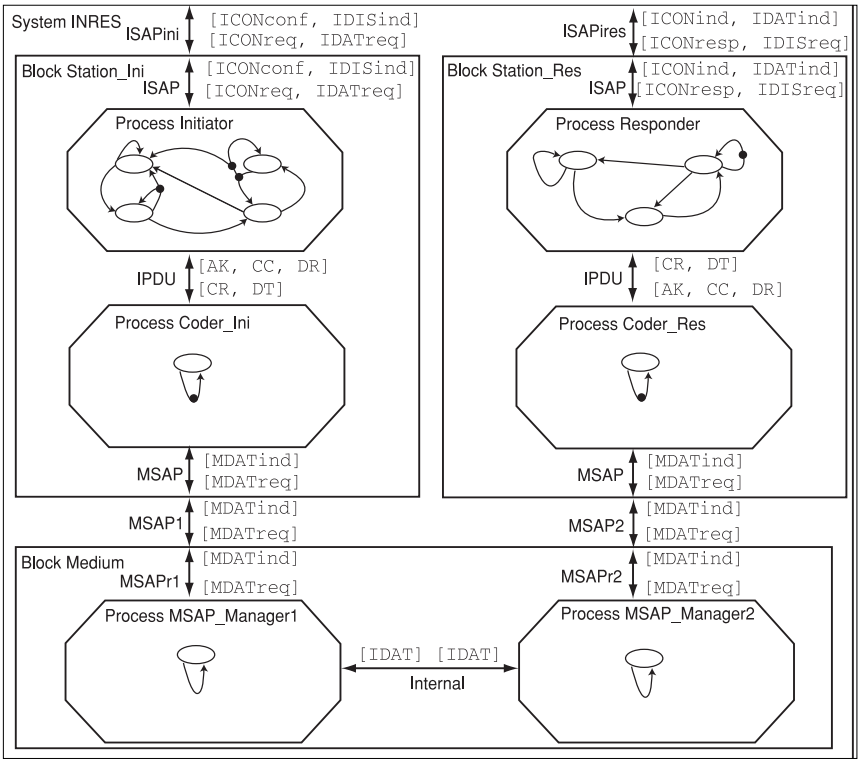
(Megjegyzés: Ebben az esetben a tesztesetek kiválasztása elég egyszerű, és bár nem feltétlenül van így nagyon nagy tesztkészletek esetében, minden evolúciós algoritmus ugyanazt a megoldást találta meg néhány generáció alatt.)

Teszteset	Lefedett tesztkritériumok	Teszteset költsége
inres01	48	260
inres02	19	115
inres03	36	200
inres04	21	125
inres05	44	240
inres06	34	190
inres07	46	250
inres08	21	125
inres09	27	155
inres10	60	320
inres11	11	75
inres12	46	250
inres13	89	465
inres14	59	315
inres15	58	310
inres16	49	265
inres17	17	105
inres18	46	250
inres19	47	255
inres20	66	350
inres21	21	125
inres22	65	345
inres23	24	140
inres24	82	430
inres25	25	145
inres26	26	150
inres27	78	410
inres28	29	165
inres29	71	375
inres30	30	170
inres31	36	200
inres32	34	190
inres33	66	350
inres34	62	330
inres35	35	195
inres36	88	460
inres37	37	205
inres38	39	215
inres39	84	440
inres40	41	225
inres41	48	260
Teljes tesztkészlet költsége:		**10145**

1. táblázat Kezdeti teszteset-halmaz

inres10	60	320
inres13	89	465
inres14	59	315
inres23	24	140
inres27	78	410
inres28	29	165
Teljes tesztkészlet költsége:		**1815**

2. táblázat Optimalizált teszteset-halmaz



7. ábra Az INRES SDL specifikáció

5. Konklúzió

Itt egy teljes automatikus tesztgenerálási módszert mutattunk be, amely a rendszer SDL specifikációjából állít elő egy tesztkészletet. Csupán egy egyszerű példával illusztráltuk az eljárást, de a mutáció-analízis bizonyítottan jól alkalmazható valós problémákra [4], és az evolúciós algoritmusok kifejlesztése mögötti motiváló erő kifejezetten a rendkívül komplex problémák kezelése volt.

A konformancia-vizsgálat a távközlési protokollok fejlesztési folyamatának kulcsfontosságú része. Mivel a tesztkészletek előállítása időigényes folyamat, az automatikus tesztgenerálás egyre fontosabb szerepet játszik a fejlesztési folyamatban. Ez a tesztkészlet-generációs eljárás könnyen implementálható, és működőképes megoldást kínál a való életbeli távközlési protokollok automatikus tesztgenerálására, nagymértékben lerövidítve ezzel a fejlesztési folyamatot.

További kutatások tárgyát képezheti, hogy milyen állapottér-bejáró algoritmusokat érdemes alkalmazni a legkedvezőbb kezdeti tesztkészlet kialakításához. A mutáció-analízis szintén egy gyorsan fejlődő terület, itt is lehetőség nyílhat a tesztesetek eddiginél gyorsabb és hatékonyabb azonosítására. A szelekciós folyamatban alkalmazott algoritmusok körét érdemes lehet tovább bővíteni, az újabb algoritmusok hatékonyságát megvizsgálni.

Irodalom

[1] ITU-T. Z.100 ajánlás (1992):
Specification and Description Language (SDL)
[2] CCITT. X.292 ajánlás (1992):
The Tree and Tabular Combined Notation (TTCN)
[3] Telelogic Tau, http://www.telelocig.com
[4] Black P. E., Okun V., Yesha Y. (2000):
Mutation Operators for Specifications.
In The Fifteenth IEEE International Conference on Automated Software Engineering,
Proceedings ASE 2000, pp.81–88.
[5] Gábor Kovács, Zoltán Pap, Gyula Csopaki (2002):
Automatic Test Selection based on CEFSM,
Acta Cybernetica 15, pp.583–599.
[6] B. Kotnyek, T. Csöndes:
Heuristic methods for conformance test selection.
[7] J. H. Holland (1992):
Adaptation in Nature and Artificial Systems:
An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence, MIT Press, Cambridge
[8] M. Salmeri, M. Re, E. Petrongari, G. C. Cardarilli (1999):
A Novel Bacterial Algorithm to Extract the Rule Base from a Training Set,
Dept. of Electronic Engineering, University of Rome
[9] N. E. Nawa, T. Furuhashi (1999):
Fuzzy System Parameters Discovery by Bacterial Evolutionary Algorithm,
IEEE Tr. Fuzzy Systems 7, pp.608–616.

A GPRS adatátviteli technológia és a GTP protokoll bemutatása

PAPP ANDRÁS, POÓS KRISZTIÁN

Veszprémi Egyetem, Műszaki Informatikai Kar, Információs Rendszerek Tanszék
papp.andras@irt.vein.hu, poos.krisztian@irt.vein.hu

Kulcsszavak: GTP, 2,5G és 3G hálózatok, SGSN, GGSN, PLMN, adatátviteli technológiák

A mobil távközlés gyors terjedése miatt megnőtt az igény az Interneten fellelhető szolgáltatások mozgás közbeni elérésére is. Az emberek egyre nagyobb része szeretné útközben is elérni információs és szórakoztató oldalait, elolvasni leveleit, esetleg kapcsolódni más adathálózatokhoz. A közelmúltig minderre csak a kevésbé hatékony vonalkapcsolt adatátvitel állt rendelkezésre, azonban a színes, egyre több információt hordozó WAP- és weboldalak eléréséhez ez már nem bizonyult elégségesnek. Ezért a 2,5G, majd 3G hálózatok kapcsán érdemes megismerkednünk a GPRS-hez és Internethez egyaránt kötődő GTP protokollal is.

Bevezetés

A korábbi GSM rendszerek adatátviteli sebessége jelentős mértékben korlátozott volt, ezért a GSM rendszert továbbfejlesztették. Ennek eredményei: a HSCSD (High Speed Circuit Switched Data – nagy sebességű vonalkapcsolt adatátvitel), GPRS (General Packet Radio Service – általános csomagkapcsolt rádiószolgáltatás), valamint a napjainkban fokozatosan terjedő EDGE (Enhanced Data rates for GSM/Global Evolution – fejlett adattovábbítás a GSM/globális fejlődésért).

Röviden tekintsük át ezek jellemzőit:

• A HSCSD [6] segítségével a GSM 14,4 kbit/s-os átviteli sebessége 28,8 vagy akár 57,6 kbit/s-osra növelhető, oly módon, hogy a felhasználóhoz egynél több időrést rendel (abban az esetben, ha a szolgáltató, a telefonkészülék, valamint a rendelkezésre álló üres időrések ezt lehetővé teszik). Hátránya a vonalkapcsolás tulajdonságából fakad: az adatkapcsolat megléte alatt sem hívást fogadni, sem kezdeményezni nem tudunk.

• A GPRS – az Internet esetében is alkalmazott technológiához hasonlóan – csomagkapcsoláson alapul. Mivel a felhasználó csak a ténylegesen forgalmazott adatmennyiség után fizet (ellentétben a fentebb említett vonalkapcsolás idő alapú számlázásával), állandóan kapcsolatban maradhat a hálózattal, így jóval gyorsabban férhet hozzá a kívánt adatokhoz, miközben – a technológiának köszönhetően – folyamatosan elérhető is marad.

• Az EDGE olyan, a GSM rendszerben alkalmazott adatátviteli eljárás és technológia, mely a hagyományos GSM szerkezet (frekvenciacsatorna, valamint az azon belüli időrésosztás) használatán alapul, de a korábbinál (GMSK – Gaussian Minimum Shift Keying) nagyobb sebességű, modulációs 8-PSK-t (8 Phase Shift Keying – 8 fázisú jelkódolás) alkalmaz. Az EDGE révén harmadik generációs technológiákra jellemző sebességű (384+ kbit/s) és minőségű adatátvitel érhető el.

Összefoglalásul álljon itt egy táblázat, melynek segítségével áttekinthetőbbé válnak az alkalmazható modulációs technikák és kapcsolási módok:

1. táblázat A ma használatos adatátviteli módok

	modulációs technika	
kapcsolási mód	GMSK (GSM)	8-PSK (EDGE)
vonalkapcsolt	CSD over GSM	CSD over EDGE
csomagkapcsolt	GPRS over GSM	GPRS over EDGE

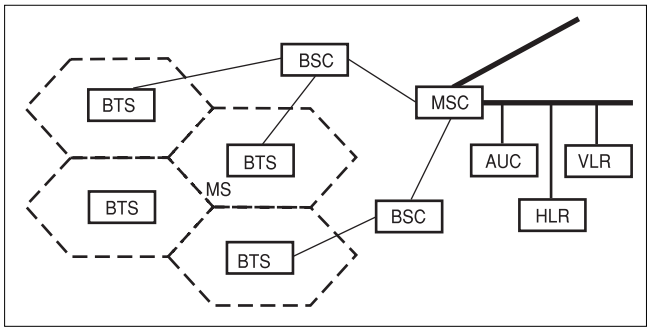
Mivel a GPRS nemcsak a ma még jóval elterjedtebb GSM, hanem a közeljövőben egyre inkább teret hódító EDGE hálózatokon is használható, érdemes erre fokozott figyelmet szentelnünk neki.

Napjainkban a legelterjedtebb alkalmazások kapcsán is elérhető ez a szolgáltatás, hiszen GPRS-t használhatunk WAP és Internet oldalak böngészésekor, vagy MMS üzenetek küldése során. Éppen ezért a GPRS mérföldkőnek számít a GSM hálózatok fejlődésében, útban a 3G hálózatok felé, a ma rendelkezésre álló hálózati infrastruktúrán.

Figyelembe véve a Magyarországon fellelhető GPRS képességű mobiltelefonok számát, kijelenthetjük, hogy maga a technológia már elterjedt. Egyes felmérések alapján a GPRS képességű készülékkel rendelkezők igen nagy hányada (több, mint 183 ezer felhasználó; 2003/III. negyedév [3]) használja is a csomagkapcsolt átvitelt WAP-ozás vagy MMS-ezés közben. Mindezek ellenére az emberek igen kis hányada van tisztában magával a szolgáltatás technikai hátterével.

A GPRS rendszer felépítése

A GPRS szolgáltatást nyújtó rendszerek alapjául a már létező GSM rendszerek [1] szolgáltak, ez utóbbiak felépítése az *1. ábrán* látható.



1. ábra A GSM rendszer vázlatos felépítése

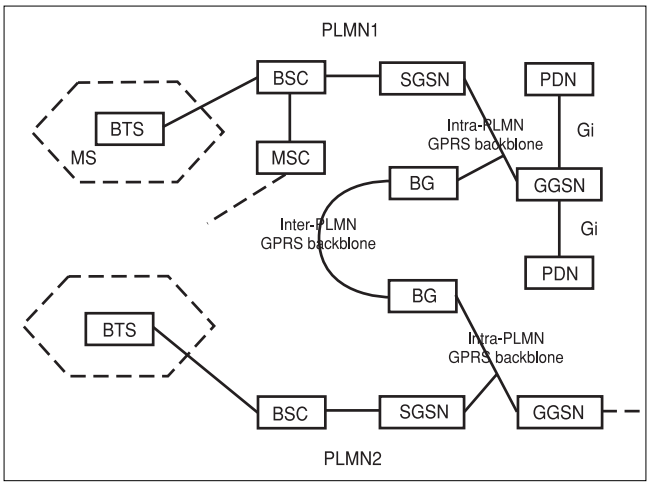
Az MS (Mobile Station) a GSM telefont jelöli. A telefon a BTS-ekhez (Base Transceiver Station – bázisállomás) csatlakozik. Egy adott BTS által lefedett területet cellának nevezünk, ezt az *1. ábrán* a BTS körüli hatszög jelzi. A BSC (Base Station Controller – bázisállomás-vezérlő) feladata a hatáskörébe tartozó BTS-ek vezérlése, erőforrás menedzselése.

Az MSC (Mobile Switching Centre – mobil kapcsoló-központ) felelős azért, hogy az MS által forgalmazott adatokat az egyik cellából a másikba juttassa. Hogy mindez megvalósulhasson, az MSC-nek a következő adatbázisokra van szüksége: HLR (Home Location Register), VLR (Visitor Location Register) és AUC (AUthentication Center).

A fenti elemekből felépített rendszer azonban nem alkalmas adatcsomagok közvetlen továbbítására. Hogy a már meglévő rendszerek csomagkapcsolt átvitel megvalósítására is képesek legyenek, új elemekkel kellett kiegészíteni őket.

Az egyik ilyen csoport a GSN-eké (GPRS Support Node). Ezek felelősek az adatcsomagok célba juttatásáért (szállítás, forgalomirányítás) az MS és a külső csomagkapcsolt hálózat között. Az SGSN (Serving GPRS Support Node) feladata a forgalomirányítás, az adatcsomagok továbbítása a hatáskörébe tartozó MS-ektől, illetve MS-ekhez, az MM (Mobility Management [1,7]), a hitelesítés, a számlázás stb. A GGSN (Gateway GPRS Support Node) a GPRS gerinchálózatát és a külső csomagkapcsolt hálózatot köti össze. A GGSN végzi a különböző hálózatok közti adatcsomagok, továbbá a PDP és GSM címek konvertálását. Mindezek mellett a GGSN-t is ellátták hitelesítési és számlázási képességekkel is. Az azonos PLMN-hez (Public Land Mobile Netwok) tartozó GSN-eket egy IP-alapú GPRS gerinchálózat köti össze (a 2. ábrán ezt az Intra-PLMN GPRS backbone jelöli).

A hálózaton közlekedő PDN (Public Data Network) csomagok alagút technikával közlekednek a két végpont között, itt alkalmazzák a cikk tárgyát is képező GTP-t (azaz GPRS Tunneling Protocolt). Mind a VPN (Virtual Private Network), mind a mobil IP kapcsán alkalmazott alagút-technikáktól eltérően itt nem biztonsági (láthatósági problémák – visibility problems) szempontok, vagy a saját IP cím megtartása, esetleg az átirányítás megvalósíthatósága vezetett a technika alkalmazásához [8].



2. ábra A GPRS rendszer vázlatos felépítése

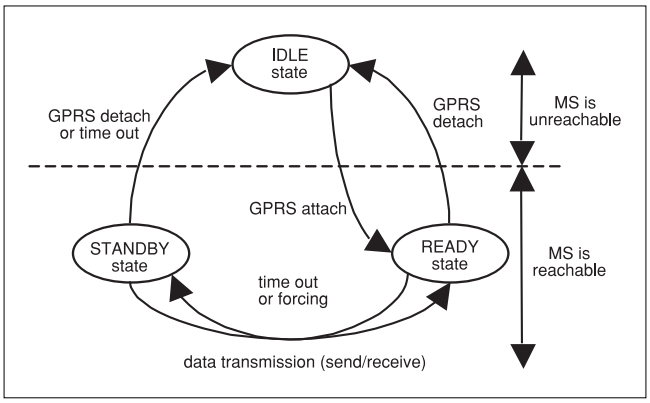
A különböző PLMN-hez tartozó GSN-eket az Inter-PLMN GPRS backbone, végül pedig a PLMN-t és a külső PDN-eket (ilyen az Internet is) a Gi interfész kapcsolja össze. A fentieknek megfelelően összeállított GPRS-képes rendszer [1,4,6] a *2. ábrán* látható.

3. A GPRS hálózat használata

A GPRS hálózat használata előtt a mobilkészüléknek be kell jelentkeznie (azaz regisztrálnia kell magát) egy SGSN-nél. A folyamat során – melyet GPRS bejelentkezésnek (GPRS attach) hívnak – a hálózat azonosítja a felhasználót (HLR-ben tárolt adatai alapján, melyeket át is másol a szóban forgó SGSN-be), majd hozzárendel a felhasználóhoz egy P-TMSI-t (Packet-Temporary Mobile Subscriber Identity). A fentiekből következik, hogy létezik GPRS kijelentkezés (GPRS detach) is, mely lehet explicit vagy implicit. Ennek során a felhasználó leválik a GPRS hálózatról.

A külső PDN hálózattal folytatott adatforgalmazáshoz elengedhetetlen egy sikeres GPRS bejelentkezés, mely után az MS feladata egy, az adott külső hálózatban is használatos cím megszerzése (például IP címé, ha a szóban forgó hálózat IP hálózat). Ezt a címet hívjuk PDP (Packet Data Protocol) címnek, melyet már fen-

3. ábra Az MS állapotai, és azok kapcsolata



tebb is említettünk. Minden kapcsolat idejére kialakul tehát egy, az adott kapcsolatra jellemző, annak karakterisztikáját leíró környezet, melyet PDP környezetnek nevezünk [1].

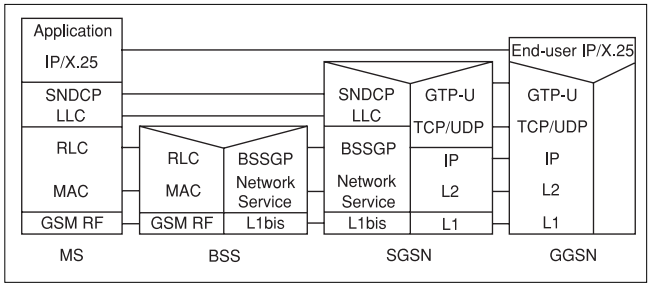
Az előzőekhez kapcsolódóan kitérnénk az MS és az SGSN különböző állapotaira is (MM), melyekből összesen hármat-hármat különbözetünk meg, ezek pedig:

- *IDLE* állapot: az MS nem kapcsolódik a GPRS hálózathoz, ilyenkor sem adatforgalmazás, sem a felhasználó kiértesítése (paging) nem lehetséges, a felhasználó elérhetetlen; a PDP környezet létrehozásához az MS-nek el kell végeznie a GPRS bejelentkezés folyamatát.
- *STANDBY* állapot: az MS már kapcsolódott a GPRS hálózathoz, így kiértesítése lehetséges, adatforgalmazás azonban ilyenkor sem.
- *READY* állapot: az MS képes PDP PDU-k (Packet Data Unit) küldésére és fogadására; az SGSN folyamatosan frissíti az útvonal- és cellaválasztási információkat.

Az állapotok közti kapcsolatot [6] a *3. ábra* teszi szemléletesebbé.

A GPRS specifikáció három különböző osztályba sorolja a GPRS szolgáltatásra képes mobilkészülékeket:

- *Class A:* az MS egyszerre kapcsolódik GPRS és GSM szolgáltatásokhoz, és egyszerre (párhuzamosan) használja őket [az ilyen típusú készülékek még nem készültek el].
- *Class B:* az MS egyszerre kapcsolódik GPRS és GSM szolgáltatásokhoz, de felváltva veszi igénybe őket (azaz egyszerre csak az egyiket), az egyes módok közti átkapcsolás automatikus (a fent ismertetett STANDBY/IDLE módok felhasználásával).
- *Class C:* az MS kizárólag CS vagy GPRS szolgáltatásokat használ, a megfelelő mód kiválasztása kézzel történik.



4. ábra A GPRS protokollkészlet felépítése

A GPRS protokollkészlet felépítése [6] a *4. ábrán* látható. A rétegekből álló felépítésnek köszönhetően könnyebbé válik a hibadetektálás, illetve -javítás.

4. A GTP szerepe, feladata és működési szakaszai

A GTP (GPRS Tunnelling Protocol) használatával lehetőség nyílik a GPRS gerinchálózat két GSN-je között multiprotokoll csomagok szállítására.

A GTP működése két fő szakaszra, a jelzési síkra (signalling plane), illetve az azt követő átviteli síkra (transmission plane) bontható.

A működés első szakaszában a GTP feladata az adatátvitelt lehetővé tevő csatorna vezérlését és menedzselését végző protokoll meghatározása, mivel a későbbiekben ennek a protokollnak a segítségével nyílik lehetősége az MS-nek a GPRS hálózat elérésére. Szintén ebben a szakaszban történik az említett csatornák létrehozása, de ekkor lehetséges módosításuk vagy törlésük is.

A 'transmission plane' (átviteli sík) alkalmazásakor a GTP alagút technika felhasználásával juttatja a felhasználó adatait célba. Az, hogy ez éppen melyik alagúton, vagy mely csatornán történik, csak és kizárólag attól függ, hogy szükség van-e megbízható kapcsolatra vagy sem stb.

Rövidítések

3G	3rd Generation
8-PSK	8 Phase Shift Keying
AUC	AUthentication Centre
BG	Border Gateway
BSC	Base Station Controller
BSSGP	Base Station System GPRS Protocol
BTS	Base Transceiver Station
CS(D)	Circuit Switched (Data)
EDGE	Enhanced Data rates for GSM/Global Evolution
GGSN	Gateway GPRS Support Node
GMSK	Gaussian Minimum Shift Keying
GSN	GPRS Support Node
GTP	GPRS Tunnelling Protocol
GTP-U	GPRS Tunneling Protocol, User plane messages
HLR	Home Location Register
HSCSD	High Speed Circuit Switched Data
L1/L2	Layer 1/Layer 2
LLC	Logical Link Control
MAC	Medium Access Control
MM	Mobility Management
MS	Mobile Station
MSC	Mobile Switching Centre
PDN	Public Data Network
PDP	Packet Data Protocol
PDU	Protocol Data Unit
PLMN	Public Land Mobile Network
P-TMSI	Packet-Temporary Mobile Subscriber Identity
RF	Radio Frequency
RLC	Radio Link Control
SDL	Specification and Description Language
SGSN	Serving GPRS Support Node
SN	Sequence Number
SNDCP	Sub Network Dependent Convergence Protocol
TCP	Transmission Control Protocol
TID	Tunnel Identifier
UMTS	Universal Mobile Telecommunications System
VLR	Visitor Location Register
VPN	Virtual Private Network

A GTP protokoll a GPRS szolgáltatást nyújtani képes hálózat felépítésének ismertetésekor már említett SGSN-ek és GGSN-ek közt használatos, a hálózat többi része nem is tud annak jelenlétéről. A GSN-ek közt húzódó GTP alagút emiatt több MS multiplexelt adatát is szállíthatja egyidejűleg anélkül, hogy azok bármit is tudnának egymásról.

5. A GTP fejléc

A GTP csomagok mindegyike egy fix (20 oktet) hosszúságú fejléccel [4] kezdődik. A fejlécben található mezők hosszai és jelentései a következők:

– Version (3): ha a PT mező értéke ’1’, a verziószámot jelöli. A jelenlegi verzió a ’0’.
– PT (1): az alkalmazott protokoll típusa, GTP esetén ’1’, a ’0’-s típus a GTP’ számára fenntartott érték (ennek használata esetén azonban az egyes mezők jelentése módosulhat).
– SNN (1): ha a csomag SNDCP N-PDU számot tartalmaz, értéke ’1’.
– Message Type (8): az üzenettípust határozza meg, a ’255’-ös jelöli az adatcsomagokat, az ettől eltérők pedig a különböző jelzéseket.
– Length (16): az üzenet (G-PDU) hossza a fejléc nélkül
– Sequence Number (16): sorszám; a jelzésre szolgáló csomagok esetén az utasítást/parancsot, míg az adatátvitel során a forgalomban lévő csomagot (T-PDU) azonosítja.
– Flow Label (16): az adatfolyam egyértelmű azonosítására szolgál a GTP csatornán belül.
– SNDCP N-PDULLC Number (8): SGSN-ek közti ’routing area’ frissítése során használatos (ez koordinálja az adatátvitelt az MS és az SGSN között); a SubNetwork Dependant Convergence Protocol kezeli az MS működési környezetéhez tartozó ’routing area’-kat.
– TID (64): a GTP csatorna azonosítója, mely egyértelmű meghatározója a GTP kapcsolatnak.

A fejléc minden mezőjének kitöltése kötelező, de tartalmuk az üzenet típusa szerint (azaz attól függően, hogy jelzésről, vagy adatok átviteléről van-e szó) változhat.

5. ábra A GTP fejléc szerkezete

8 7 6	5	4 3 2	1	
Version (=0)	PT (1=GTP)	Spare '1 1 1'	SNN	1
Massege Type				2
Length				3-4
Sequence Number				5-6
Flow Label				7-8
SNDCP N-PDULLC Number				9
Spare '1 1 1 1 1 1 1 1'				10
Spare '1 1 1 1 1 1 1 1'				11
Spare '1 1 1 1 1 1 1 1'				12
Tunnel ID (TID)				13-20 (8)

6. A jelzési sík (Signalling Plane)

Bár a GTP jelzések [4] szorosan kapcsolódnak az adatátvitelhez (hiszen az ahhoz szükséges csatornák létrehozásáért, módosításáért és lebontásáért ők felelnek), bizonyos szempontból teljesen függetlenek is tőlük, hiszen a jelzéseknek nem kell ugyanazon a csatornán közlekedniük, mint az adatoknak.

Egy jelzést tartalmazó üzenet küldésekor – az időzítő elindítása mellett – az üzenet bekerül a küldő GSN kimeneti sorába is, a megfelelő sorszámmal (SN) ellátva, és mindaddig ott marad, míg az üzenet sikeres kézbesítésének visszaigazolása meg nem érkezik (természetesen azonos sorszámmal (SN) ellátva). Ha a megadott időn belül nem jön válasz, megtörténik az üzenet újraküldése. Kettőzött üzenetek esetén a másodikként érkezett üzenetet figyelmen kívül kell hagyni. A GTP ezzel a módszerrel próbálja meg elérni a jelzést szolgáló csomagok biztonságos célba juttatását, hiszen mint ismeretes, a GTP az IP-re épül, az pedig nem minden esetben nyújt hibamentes átviteli szolgáltatást.

A GPRS kapcsolat (mindkét oldali) felépítésének, illetve lebontásának vázlatos SDL leírása (processz szinten) a függelékben található. Az ábrázolás nem teljes, hiszen az általunk vizsgált részek specifikációi nem tértek ki például az előforduló hibák kezelésének módjára, csak annak észlelésére.

A jelzési síkhoz tartozó SDL leírások a cikk végén található Függelékben találhatóak meg *(I-IV. ábrák)*. Ezek a kapcsolat felépítést és lebontást mutatják, azon belül pedig az egyes jelek irányát az SGSN, GGSN szempontjából.

7. Adatátviteli sík (Transmission Plane)

Ha egy MS (egy SGSN által) GPRS adatkapcsolatot (PDP connection) szeretne létesíteni, GTP jelzőüzenetek [4] segítségével egy csatornát hoz létre (PDP Context Activation), melyet a már ismertetett egyedi azonosítóval (Flow Label) lát el.

Ezt a csatornát használja a GTP adott GSN párok közti adattovábbításra, mégpedig oly módon, hogy a beérkező T-PDU-hoz a fentebb vázoltak alapján egy GTP fejlécet illeszt (G-PDU), majd az egy irányba közlekedő, és azonos GSN-hez tartozó adatokat multiplexeli. A vevő oldalon ennek természetesen a fordítottja történik, azaz az adatfolyam demultiplexelése után leválasztásra kerül a GTP fejléc, így visszakapjuk a tényleges adatunkat, a T-PDU-t.

Az adatátvitel során alkalmazott útprotokoll lehet UDP/IP (ha az MS által forgalmazott adatok csomagkapcsoltak) vagy TCP/IP (ha pedig kapcsolatorientáltak, például X25 hálózatban).

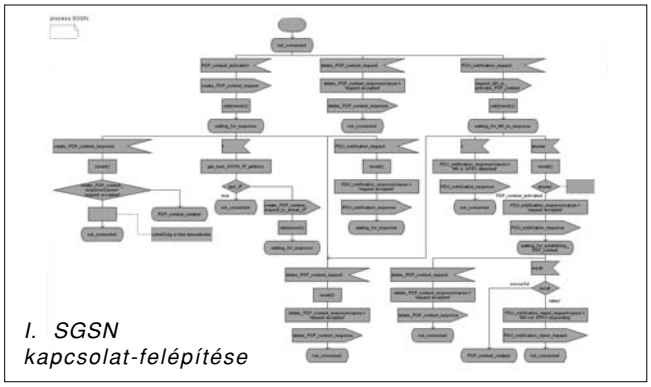
A Függelékben természetesen megtalálható az adatátviteli szakasz (specifikáció [5] alapján) általunk elkészített SDL leírása is *(V. ábra)*.

8. Összefoglalás

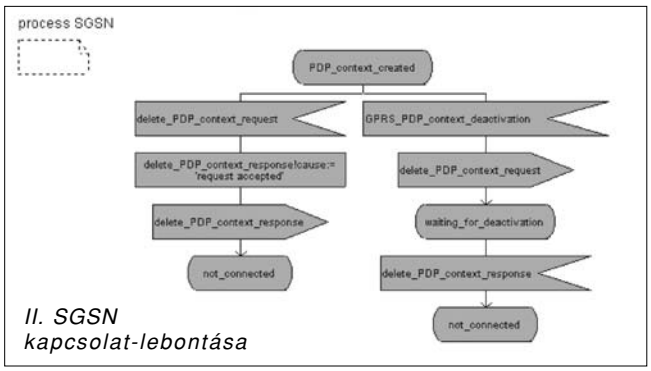
A fentiekből látható tehát, hogy a GPRS használatával jóval szélesebb körű információszerzési lehetőségek nyílnak meg előttünk. Ezzel mobil eszközeinkről is elérhetjük a különböző IP alapú hálózatokat, s tehetjük mindezt a hagyományos GSM rendszer esetén elérhető sebesség többszörösével. S végül, de nem utolsó sorban, a GPRS az első (de nem utolsó) lépés a harmadik generációs mobil hálózatok (pl. UMTS) felé.

Függelék

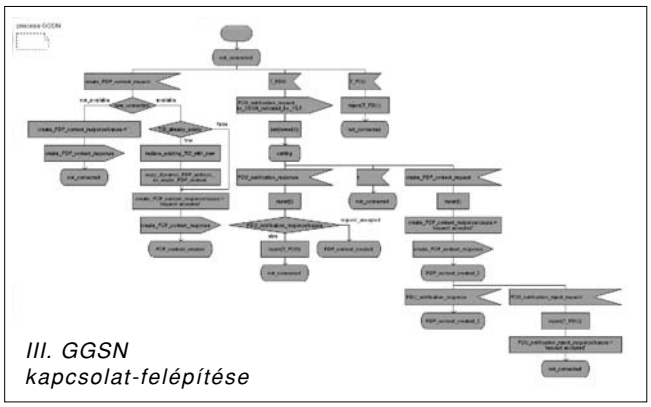
A jelzési sík



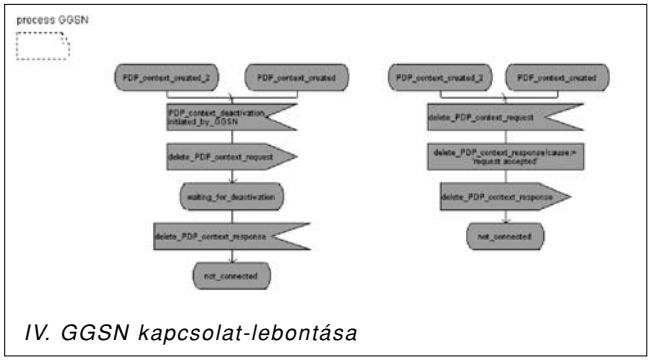
I. SGSN kapcsolat-felépítése



II. SGSN kapcsolat-lebontása

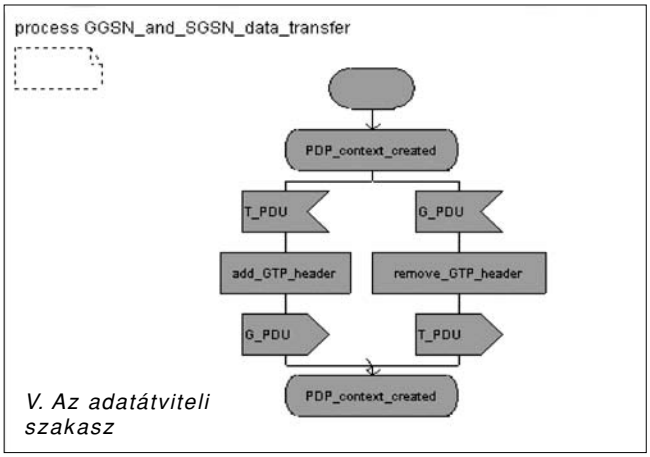


III. GGSN kapcsolat-felépítése



IV. GGSN kapcsolat-lebontása

Adatátviteli sík



V. Az adatátviteli szakasz

Irodalom

[1] Maryland Center for Telecommunications Research
Shantanu Prasade, Anjali Parekh, Viral Shah: GPRS
http://apollo.cs.umbc.edu/~classes/cmsc681/fall2002/
Network Architectures and Protocols, Projects

[2] mpirical limited
http://www.mpirical.com/, Companion

[3] Nemzeti Hírközlési Hatóság, www.nhh.hu;
Piaci információk; Tanulmányok, elemzések

[4] Jyke Jokinen:
GPRS & UMTS Protocols: GTP details
Tampere University of Technology, Dep. of IT,
Advanced Topics in Telecommunications
http://www.cs.tut.fi/kurssit/8309700/

[5] TS 101 347 Version 7.10.0 (2002-12)
Digital cellular telecommunications system (Phase 2+);
General Packet Radio Service (GPRS);
GPRS Tunnelling Protocol (GTP) across
the Gn and Gp Interface
(3GPP TS 09.60 version 7.10.0 Release 1998)

[6] T. Araour, Y. Rabbani, O. Ahmed (2003):
Le Lancement du GPRS
Univ. de Versailles St Quentin en Yvelines,
http://dessr2m.adm-eu.uvsq.fr/,
Journées Portes Ouvertes Annuelles du DESS R2M

[7] Yannick Marcq:
Glossary of GPRS abbreviations,
My GPRS Questions & Answers, GPRS Q&A Book
http://users.evtek.fi/~k0300183/

[8] S. Giacometti, R. Mameli:
Tunneling Effectiveness in the Access Environment
http://www.coritel.it, Publications,
Papers published on Scientific/Technical Journals
Fitce Conference, August 1999, Utrecht

Általános célú biztonságos anonimitási architektúra

TÓTH GERGELY, HORNÁK ZOLTÁN

BME, Méréstechnika és Információs rendszerek Tanszék
tgm@mit.bme.hu, hornak@mit.bme.hu

Reviewed

Kulcsszavak: anonimitás, hálózati architektúra, biztonságos kommunikáció

A távközlésben legújabb követelményként napjainkban egyre inkább megjelenik az anonimitás (tipikusan elektronikus szavazás, közvélemény-kutatás vagy fizetés során). A jelenlegi hálózati réteghierarchia azonban önmagában nem tartalmazza ezt a funkciót. Ezen probléma megoldására tesz javaslatot a cikk egy általános célú biztonságos anonimitási architektúrával, amely a jelenlegiek mellett új, kifejezetten anonimitási funkciókat teljesítő rétegeket vezet be és meghatározza azok helyét a jelenlegi modellben.

Az elmúlt évtizedek során a számítástechnika, a hardver, a szoftver, valamint a távközlés terén tapasztalható rohamos fejlődés lehetővé tette a rendszerek egyre nagyobb fokú integrálását. Ez a tendencia az Internet térhódításával az informatika elé újabb és újabb kihívásokat állít. Az elsőként fellépő távközlési problémákra – a szükséges sávszélesség és megbízhatóság biztosítására – már léteznek átfogó architekturális megoldások.

Az előző évtizedben újabb igények merültek fel: a meglévő adottságok mellett már bizalmas információcserére is szükség volt. A titkosítás, integritás-védelem, hitelesítés stb. megoldására már szintén léteznek bevált megoldások [1].

Újabban a személyi és személyes adatok védelme került előtérbe. Ahogy egyre több adatbázist kapcsolnak össze és tesznek – részben nyilvánosan – kereshetővé, úgy lehet az egyes emberekről egyre több információt összegyűjteni. Egyfajta ellenintézkedésként ezért van szükség az adatvédelemre, a személyi és személyes adatok illetéktelen hozzáférés elleni védelmére.

Az anonimitást ezen belül tekinthetjük egyfajta extrém adatvédelmi módszernek, ahol az alany személyazonosságát rejtjük el, ezáltal szüntetjük meg (vagy csökkentjük elfogadható mérték alá) annak esélyét, hogy egy támadó az esetleg megtudható személyes adatokat hozzárendelje egy személyhez és így egy nem megengedett on-line profilt állítson össze [2].

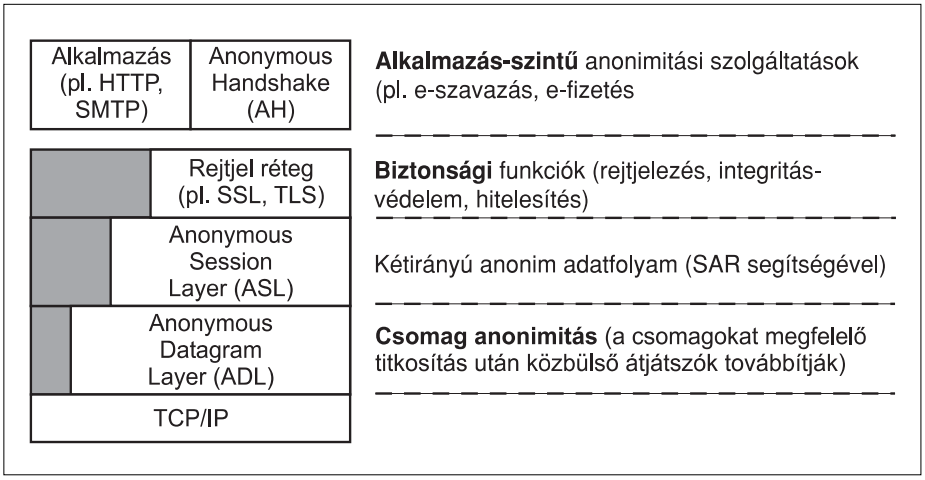
Anonimitás elérésére léteznek már különböző technikák, azonban hiányzik egy olyan egységes keretrendszer, ahol a biztonsági (rejtjelezési) módszerek mellett tetszőleges anonimitási szolgáltatás is megvalósítható. Az általános célú biztonságos anonimitási architektúra *(general-purpose secure anonymity architecture,* GPSAA) célja pont ennek az űrnek a betöltése, azaz a biztonsági funkciók ötvözése az anonimitási megoldások két nagy csoportjával:

• Anonim üzenetküldési technikák:

Két fél közötti kommunikáció során biztosítják, hogy még a hálózati forgalom megfigyelése és módosítása esetén sem deríthető ki adott küszöbértéknél nagyobb valószínűséggel, hogy ki kinek küld adatot [3]. Tipikus alkalmazás az anonim levelezés vagy anonim böngészés.

• Anonim engedélyezési sémák:

Lehetővé teszik, hogy egy szolgáltató az anonimitási hatóság segítségével megbizonyosodjon, hogy egy számára anonim alany jogosult-e egy szolgáltatás igénybevételére. Tipikus alkalmazási területek: e-fizetés (az anonimitási hatóság a bank, az engedélyezés pedig az elektronikus pénz beszerzése és átadása a szolgáltatónak), e-szavazás.



1. ábra
Az általános célú biztonságos anonimitási architektúra rétegei

Architektúrális felépítés

A bevezetőben leírtak alapján egy olyan általános keretrendszerre van szükség, mely lehetővé teszi a fenti csoportokba sorolható tetszőleges anonimitási módszer megvalósítását és ezzel együtt biztonsági funkciók alkalmazását.

A távközlés során fellépő anonimitási problémák alapvetően az IP protokollcsalád tulajdonságából adódnak (egy tetszőleges lehallgatott IP csomag tartalmazza mind a küldőjét, mind a fogadóját). Azonban az Internet elterjedtsége miatt ezt megváltoztatni nem lehet, felsőbb rétegekben kell az anonimitást garantálni. Ezen megkötés mellett dolgozták ki a GPSAA rétegszerkezete *(1. ábra)*.

A TCP/IP feletti első réteg az ADL *(Anonymous Datagram Layer)*, melynek feladata fix méretű csomagok egyirányú anonim átvitele. Erre építve a következő réteg, az ASL *(Anonymous Session Layer)*, már képes kétirányú anonim adatfolyam kezelésére SAR *(Segmentation And Reassembly)* segítségével. Az anonim adatfolyamot felhasználva már alkalmazhatóak a bevált rejtjel rétegek. Végül legfelül helyezkedik el az AH *(Anonymous Handshake)*, mely az anonim engedélyezés alkalmazás-szintű feladatait látja el.

ADL – Csomagok anonimitása

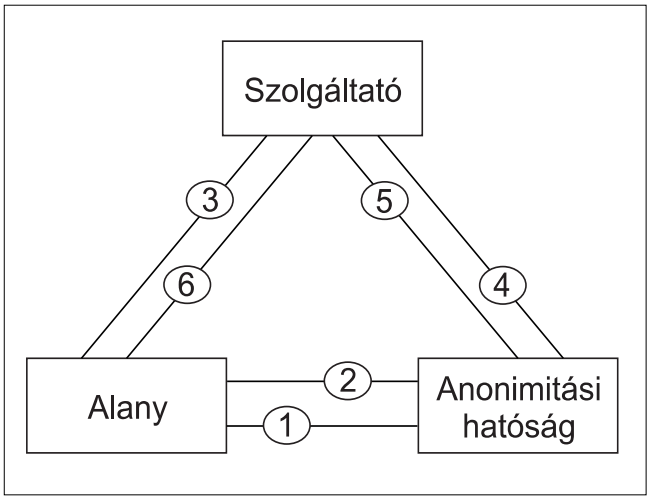
Az ADL réteg feladata két kommunikáló fél között egységes méretű csomagok anonim továbbítása. Ennek során *(2. ábra)* a feladónál a csomagot rejtjelezik, majd az ADL csatornán keresztül jut a fogadóhoz. Fontos megemlíteni, hogy egyrészt az ADL csatorna nem feltétlenül egy fizikai egység, lehet több átjátszó elosztott hálózata, másrészt minden egyes átjátszó átkódolja és összekeveri a csomagokat, annak érdekében, hogy ne lehessen a hálózat lehallgatásával azok útját követni.

Az, hogy az átkódolások milyen algoritmust követnek, hány átjátszó építi fel a hálózatot, nem része az ADL specifikációnak, a réteg meghatározásánál csak az interfészt rögzítik, mely a szolgáltatással kapcsolatos követelményeket tartalmazza.

ASL – Kétirányú anonim adatfolyam

Az ADL rétegre építve következő lépésként lehetővé kell tenni a kétirányú anonim adatfolyam kiépítését. Ezt a célt szolgálja az ASL réteg. Különösebb anonimitási funkciója nincs, egyedüli feladata, hogy a felsőbb rétegektől kapott adatfolyamot a küldő oldalon feldarabolja az ADL réteg által megkövetelt méretű fix csomagokra, majd a fogadó oldalon ezeket a csomagokat helyes sorrendben adatfolyammá állítsa össze, (hiszen az ADL a lehallgatók megtévesztése érdekében a csomagok kézbesítési sorrendjét tipikusan össze is keveri).

Az ASL réteg felett helyezkedik el a rejtjel réteg, mely az adatfolyamokon végzi a különböző biztonsági feladatokat. Ugyan már az ADL rétegben is van rejtjelezés, azonban ott csak az anonimitás kompromittálása ellen (mely során bizonyos átjátszók láthatják a kódolatlan üzenetet), itt pedig már az átjátszókban sem bízva a lehallgatás ellen kell végpont-végpont titkosítást végezni, ahol biztosítható, hogy csak a fogadó fél tudja dekódolni az üzenetet.



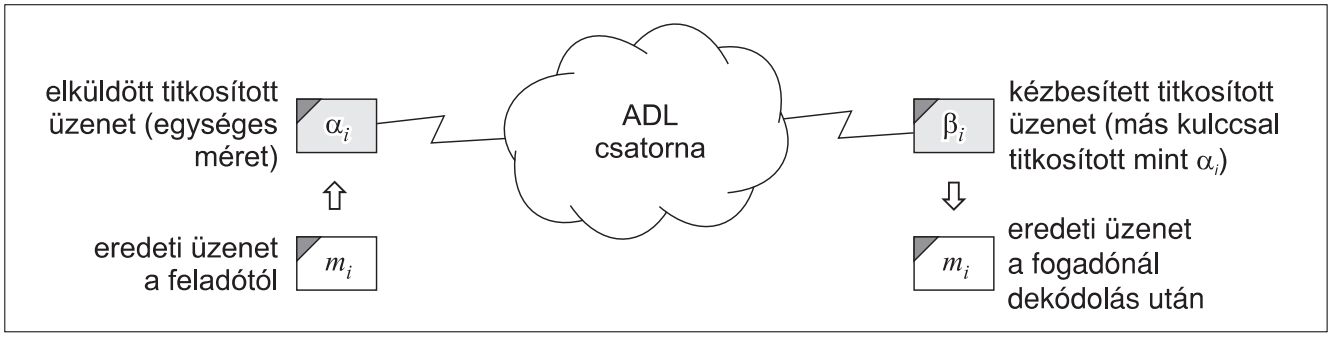
3. ábra Az anonim engedélyezés általános lefolyása az AH keretében

AH – Alkalmazás-szintű anonimitási szolgáltatások

A most már biztonságos, kétirányú anonim adatfolyam felett történhet meg ezek után az anonim engedélyezés az AH keretében *(3. ábra)*.

Az anonim engedélyezés folyamata két fázisból áll. Az első fázisban az alany az anonimitási hatóságtól beszerzi az anonimitási okmányokat (az ATM-es pénzfelvét analógiájára) (1) (2), melynek során nem anonim, sőt személyazonosságát kifejezetten igazolja.

2. ábra Csomagok küldése általános esetben az ADL rétegen keresztül



A második fázisban történik meg a szolgáltatás tényleges igénybevétele, itt már az alany anonim. Először átadja az okmányokat és kéri a szolgáltatást (3). Ezután a szolgáltató ellenőrzi az okmányokat (4), majd az anonimitási hatóság válaszától (5) függően teljesíti a kérést (6). GPSAA az AH keretében is csak követelményeket és egy interfészt fogalmaz meg, melyben ezek után különböző algoritmusok is megvalósíthatók.

Gyakorlati példaként említhetnénk a Chaum-féle vak aláírás módszerét [4], melyet elektronikus anonim fizetés lebonyolítására dolgoztak ki.

Alkalmazás

Amellett, hogy a GPSAA interfészeket és követelményeket definiál, folyamatban van egy referencia-megvalósítás elkészítése is, mely a *4. ábrán* ismertetett sémát követi.

A kezdeti tesztekhez ADL szinten a PROB-csatorna [5], míg AH keretében a Chaum-féle vak aláírás módszer [4] került felhasználásra.

Összefoglalás

A GPSAA egy olyan általános keretrendszer, mely lehetővé teszi különböző anonimitási módszerek és biztonsági szolgáltatások együttes alkalmazását. A keretrendszer megvalósítása után következő lépésként a rendszer által nyújtott anonimitás mérése és a rendszer finomhangolása következik.

Irodalom

[1] Dierks, T., Allen, C.:
RFC 2246 – The TLS Protocol Version 1.0.
Certicom, 1999

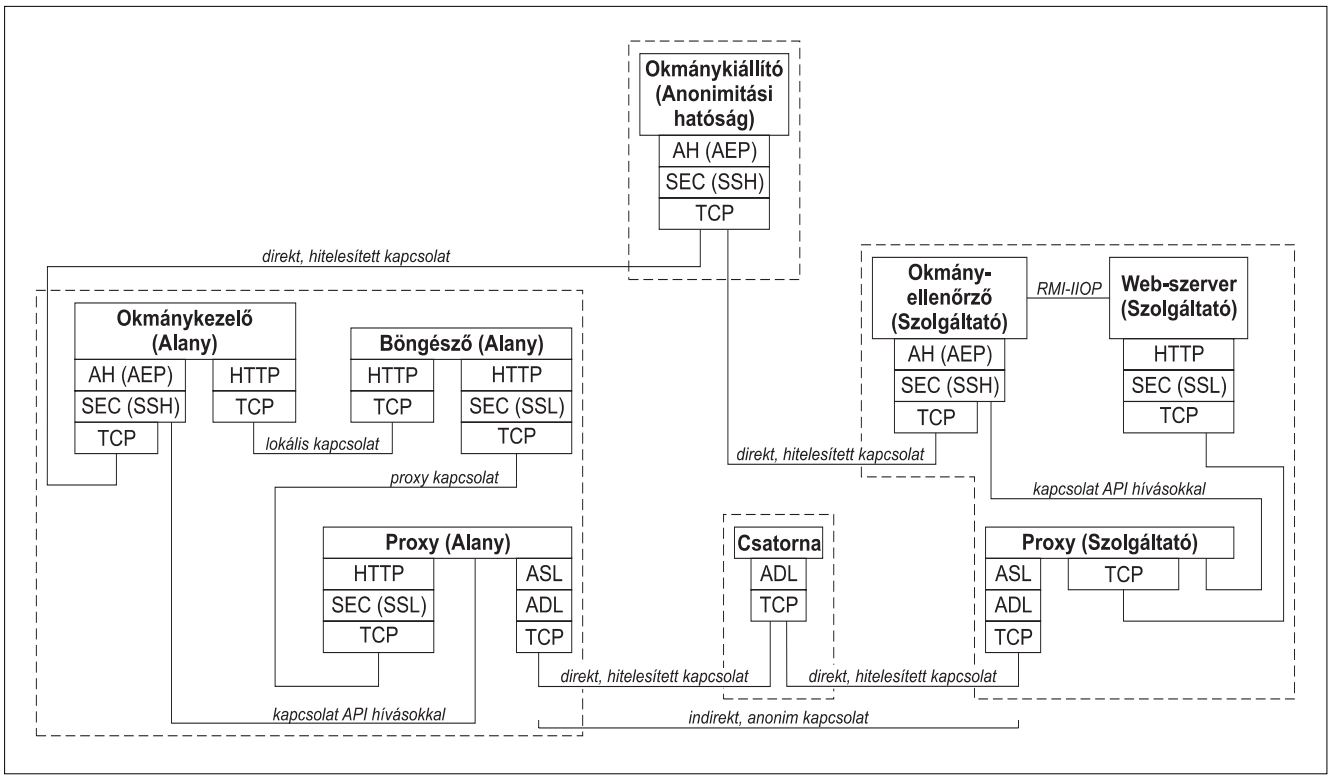
[1] Froomkin, A. M.:
Flood Control on the Information Ocean:
Living with Anonymity,
Digital Cash and Distributed Databases, 1996.,
http://www.law.tm/

[1] Reed M., Syverson, P., Goldschlag, D.:
Anonymous Connections and Onion Routing.
IEEE Journal on Selected Areas in
Communication Special Issue on Copyright and
Privacy Protection, 1998., pp.482–494.

[1] Chaum, D.:
Blind Unanticipated Signature Systems.
USA szabadalom: 4 759 064, 1998.

[1] Tóth, G., Hornák, Z.:
Megfigyelhető black-box csatorna
forrásrejtő tulajdonsága.
Híradástechnika, 2003/05, pp.41–44.

4. ábra Az általános célú biztonságos anonimitási architektúra implementációja



Processz algebrai eszközök a szenzorhálózatok biztonsági vizsgálatában

GÉMESI ROLAND*, IVÁDY BALÁZS**, ZÖMBIK LÁSZLÓ***

**BME-TMIT, gemesi@alpha.tmit.bme.hu*
***BME-TMIT, ivady@alpha.tmit.bme.hu*
****Ericsson Magyarország, BME-TMIT, laszlo.zombik@ericsson.com*

Reviewed

Kulcsszavak: biztonsági protokoll vizsgálat, szenzorhálózat kódolása, CSP, kulcs-csere

A kommunikációs és hálózati technológiák nagymértékű fejlődése, valamint a mind nagyobb fokú miniatűrizáció lehetővé tette napjainkra vezeték nélküli érzékelőrendszerek megvalósítását. A szenzor-számítógépek önszerveződő ad hoc hálózatot alakítanak ki, melyeknek biztonsága a hagyományos rendszerekhez képest nehezebben garantálható. Cikkünkben bemutatjuk, miként alkalmazhatóak a hagyományos távközlőhálózatokban bevált processz algebrai eszközök ilyen rendszerek biztonsági tulajdonságainak ellenőrzésére.

1. Szenzorhálózatok

A beágyazott és a távközlési technológiák az utóbbi években rohamos fejlődésen mentek keresztül. Mára az egyre kisebb és takarékosabb eszközökben található mikroszámítógépek viszonylag egyszerűen és olcsón kiegészíthetőek a vezeték nélküli kapcsolat képességével. Ez lehetővé teszi, hogy különálló egységeink együttműködésével összetettebb funkciók valósulhassanak meg.

A *szenzorhálózatok* számos vezeték nélküli érzékelő egységet egy közös rendszerbe kapcsolnak. A kommunikációban résztvevő egységek az érzékelőelemen kívül magukba foglalnak egy komplett mobil mikroszámítógépet, azaz energiaforrást, processzort, memóriát, valamint a vezeték nélküli kapcsolat létesítésének képességét is. A további integrációval egyetlen chipbe zsugorított, majd a méretek további csökkentésével akár porszemnyi méretű szenzorok is olcsón elérhetővé válnak. A nagyszámú és területileg elszórt egységek összekapcsolására használt vezeték nélküli technológia jelentős előnyöket kínál és új lehetőségeket nyit [1].

A parányi szenzor-számítógépek erősen korlátozott erőforrásokkal rendelkeznek, melyek csak korlátozott számítási kapacitást tesznek lehetővé. Ennek ellenére a sok egység együttműködésével az érzékelt bemeneteken elosztott jelfeldolgozási megoldásokkal akár komplex mintafelismerési feladatok is megvalósíthatóak.

A kommunikáció felépítésének automatikusan és önszerveződő módon kell végbemennie. *Mobil ad hoc hálózatoknak* nevezzük az olyan vezeték nélküli hálózatokat, melyek nem igényelnek előzőleg kiépített infrastruktúrát, vagyis előfeltételezések nélkül is képesek működni. Ilyen esetekben a központi funkciók ellátását elosztott módon kell végezni.

Biztonsági oldalról közelítve megállapíthatjuk, hogy a vezeték nélküli szenzorhálózatok számos fenyegetésnek vannak kitéve [3]. A kommunikáció nyilvános médiumon keresztül történik, melyhez rosszindulatú felek is hozzáférhetnek. A korlátozott erőforrások miatt a rendszer biztonságának védelméhez nem használhatóak a túlságosan nagy számítási igényű kriptográfiai algoritmusok. Az egyes kicsiny szenzorok kompromittálódása is jelentős fenyegetést jelent, hiszen a bennük található információk fizikai védelme nehezen oldható meg. További probléma, hogy a klasszikus távközlési rendszerekben elterjedt hitelesítési mechanizmusok gyakran igényelnek megbízható harmadik felet, melyek ad hoc környezetben nem állnak rendelkezésre.

E problémák ellenére szeretnénk, hogy rendszerünk biztonságosan és megbízhatóan működjön. Létezik néhány olyan biztonsági mechanizmus, amely a teljes önszerveződésre támaszkodik [2], ám ezek számos kérdést hagynak maguk mögött. Rendszerünket biztonságosnak tekinthetjük, ha bizonyosságot szereztünk arról, hogy az alkalmazott mechanizmusok tetszőleges támadói viselkedés esetén is megfelelő biztonságot nyújtanak.

Az elmúlt évizedekben a klasszikus kommunikációs rendszereken processz algebrai eszközökkel végzett biztonsági analízisek jelentős eredményeket hoztak. Mind támadások prezentálására, mind a megfelelő biztonság bizonyítására alkalmasnak bizonyultak.

Cikkünk célja, hogy bemutassa, miként alkalmazhatóak e módszerek szenzorhálózatok biztonságának analízisére. Bemutatjuk a *CSP (Communicating Sequential Processsses)* processz algebra alapelveit és fő szintaktikai elemeit. Áttekintjük a biztonsági vizsgálatra használt modellt, majd szenzorhálózatok biztonsági protokolljain végzett analíziseinket mutatjuk be. A cikk végén a modell további kiterjeszthetőségének kapcsán biztonságos útvonalválasztás vizsgálatának lehetőségeit vizsgáljuk meg.

2. Biztonsági protokollok és ellenőrző módszereik

A biztonságos kommunikációs rendszerekkel szemben támasztott követelmények már kiforrottnak tekinthetőek. Bár az önszerveződő hálózatok működése gyöke-

resen más szemléletet rejt, a biztonsági igények hasonlóak maradtak. Mobil ad hoc hálózatokban felmerülő biztonsági követelmények [2] közül munkánk során a titkosság, integritás, hitelesség és frissesség tulajdonságokra térünk ki.

A biztonsági protokollok olyan üzenetváltási szabályok, amelyek végrehajtásuk befejeztével valamilyen biztonsági tulajdonságot alakítanak ki. Egy támadó úgy igyekszik beavatkozni az üzenetváltásokba, hogy meghiúsítsa e kialakuló tulajdonságot. A biztonsági protokollok tervezése nagy körültekintést igényel, jóságuk bizonyítása nehéz feladat.

Hogy a kérdéshez egzaktul közelítsünk, rögzítenünk kell a támadó feltételezett képességeit. A legszélesebb körben használt támadó modell a *Dolev-Yao* támadó, amely az ellenséges médiumot reprezentálja. Hozzáfér az összes távközlési csatornához, így üzeneteket távolíthat el, lehallgathatja az üzenetváltásokat és újakat is beszúrhat. A megszerzett információk szétbontásával és a komponensek újraösszeállításával, valamint következtetésekkel új elemekhez is juthat. Egyetlen korlátja a *tökéletes-kriptográfia (perfect cryptography)* feltevés, amely szerint a kriptográfiai építőelemek tökéletes működésűek, azok próbálgatáson alapuló kompromittálódása nem lehetséges.

Nem létezik olyan általános algoritmus, melyet tetszőleges biztonsági protokollon lefuttatva, bizonyíthatná annak megfelelőségét [4]. Garantált biztonsághoz a protokollok formális analízisével juthatunk.

Az utóbbi időkben biztonsági protokollok analízisében jelentős sikereket hozott a CSP processz algebra: sokáig biztonságosnak hitt protokollokat kompromittáló új támadási viselkedéseket mutatott [5]. A módszer egy eloszott rendszer állapotgépként való modellezésére és ellenőrzésére alkalmas. Mivel e processz algebrához gépi modell ellenőrző eszköz is létezik, az analízis folyamata teljesen automatikusan, emberi beavatkozás nélkül történhet meg. A módszer megadja a támadói viselkedésének részleteit, illetve véges résztvevőből és protokollfutamokból álló rendszeren a megfelelőség bizonyítására is alkalmas.

A protokoll futamainak korlátozása jelentős gyengítésnek tűnik, azonban Lowe [5]-ban bizonyította a protokollok egy igen széles osztályára, hogy hiba esetén a támadás már néhány, meglehetősen kevés, lépés esetén is jelentkezik. A protokollok jelentős része ide tartozik, így általában már viszonylag kis állapottéren végzett ellenőrzéssel is általános állításra juthatunk.

A következőkben ezért rátérünk a CSP azon elemeinek bemutatására, melyeket a későbbi modellezésben és az analízisekben felhasználunk.

3. A CSP alapelemei

A CSP egy párhuzamos rendszerek leírására hivatott nyelv. A párhuzamos rendszerekben egyidejűleg több független folyamat (processz) is fut, melyek egymással kölcsönhatásba kerülhetnek, azaz kommunikációt folytathatnak. A CSP leírónyelv kezdetben algebrai rendszerként létezett mely lehetővé tette a különböző jelenségek formális leírását és vizsgálatát (például deadlock, livelock, nemdeterminizmus) [6].

Párhuzamos rendszerek CSP modellezése során az egyes processzek viselkedését lehet kezelni. Egy processz különféle állapotokban lehet, melyek között diszkrét események (event) hatására átmenet lehetséges. Egy processz számos eseményt ismerhet, melyek teljes halmazát a processz abc-jének (Σ) nevezzük. Egy processz egyes állapotában bizonyos eseményeket elfogadhat, vagy visszautasíthat. Egy processz *Trace*-énekk nevezzük az általa végrehajtható összes eseménysorozatok halmazát.

A $P \mathrel{\hat{=}} e \rightarrow Q$ egy olyan P processzt ír le, amely az e eseményben való részvétel után Q processzként viselkedik. A *Stop* névre hallgató processz semmilyen eseményben nem vesz már részt, míg a *Skip* állapot jelzi egy processz sikeres befejeztét.

Általános koncepció, hogy az események csatornákon jelentkeznek, ilymódon a $c.e$ esemény a c csatornán bekövetkező e eseményt jelöli. Ilymódon összetettebb konstrukciók is használhatóak, mint például *c.e.f.g.*

Az olyan szituációkat, amikor a processz többféle eseményt is elfogadhat, *választásnak* nevezzük. Egyik fajtája a külső választás *(external choice)*: a $P \Box Q$ processz a külvilágtól függően P-ként vagy Q-ként viselkedik. Ez gyakran kiegészül a *guarded alternative* operátorral, amely az egyes választásokat explicit feltételhez köti.

Az eddigi jelölések segítségével független processzek írhatóak le. Kommunikáció, vagyis közös esemény leírására hivatott a párhuzamosság *(parallel)* operátor. Ennek hatására bizonyos processzek csak egyszerre (párhuzamosan) vehetnek részt egy eseményben. Például a $P \mid\mid_e Q$ jelöléssel megadott processz a P és Q processzek olyan egyesítése, amelyben az e esemény csakis közösen hajtható végre. A teljesen független működésű processzek leírója az összefésülés *(interleave)* operátor. A $P \mid\mid\mid Q$ processz a P és Q olyan kombinációja, melyek minden eseményben csakis külön-külön vehetnek részt.

Egy operátor sokszoros alkalmazását segítik a replikált oprátorok. Ennek jelölési szisztémája a külső választás operátorra a következő: $\Box\, a : S \bullet a \rightarrow P$. Ennek jelentése, hogy bármely $a \in S$ esemény bekövetkezhet, mellyel a P processzhez jutunk.

Összetettebb rendszerek modellezéséhez szükség lehet moduláris zárt komponensek létrehozására. Lehetséges ezért az elrejtő *(hiding)* operátorral bizonyos eseményeket elrejteni a külvilág elől. A $P \setminus E$ processz P-ként viselkedik, ám az E eseményhalmaz a P-n kívülről rejtett. Megvalósítható események átnevezése is, a $P[[e \leftarrow f]]$ processz a P-vel megegyező működésű, ám annak eredeti e eseménye most kívülről f-ként látható.

Ezen operátorok segítségével lehetséges a különálló processzek leírása, valamint azok egymással való összekapcsolása. Eljuthatunk egy olyan processzhez, amely egy teljes összekapcsolt kommunkációs rend-

szert reprezentál, hordozza annak minden lehetséges eseménysorozatát.

Az általunk ellenőrzésre használt reláció a *finomítás*. Egy *Q* processz *traces finomítottja* egy *P* processznek, ha a *Q* által elvégezhető összes eseménysorozat elvégezhető *P* által is. Ezt a $P \sqsubseteq_T Q$ fejezi ki, vagyis ez esetben $Traces(Q) \subseteq Traces(P)$.

Ha *S* egy specifikációul szolgáló processz és *I* egy implementáció, a megvalósítás konformanciája ellenőrizhető a $S \sqsubseteq_T I$ finomítás ellenőrzésével. Véges állapotterű processzek közötti finomítás ellenőrzést az *FDR2* modell ellenőrző segítségével automatikusan elvégezhetjük.

4. Kommunikációs rendszerek modellezése CSP-ben

A kommunikációs protokollok modellezésére kiválóan alkalmas a CSP keretrendszer. A kommunikáció résztvevői adott szabályoknak eleget téve, a protokoll szabályai szerint működnek.

A résztvevők tehát processzek, melyek a protokoll egyes lépései során a küldés *(send)*, illetve a fogadás *(recv)* csatornákon eseményekben vesznek részt, majd ezzel új állapotba kerülnek. Egy-egy ilyen esemény reprezentálja a forrás és célentitásokat, valamint az aktuális üzenetet is. Így például az *A*-tól a *B*-be irányuló üzenetváltás során a *send.A.B.üzenet*, illetve a *recv. A.B.üzenet* események jelennek meg.

A modell az üzenetek konstruálását oszthatatlan adattagok egy véges halmazából kiindulva, a különféle kriptográfiai műveleteket megtestesítő konstrukciós operátorokkal képezi. Ilyen konstrukciós operátor például a titkosítást leíró *Encr.(Adat, Kulcs)*.

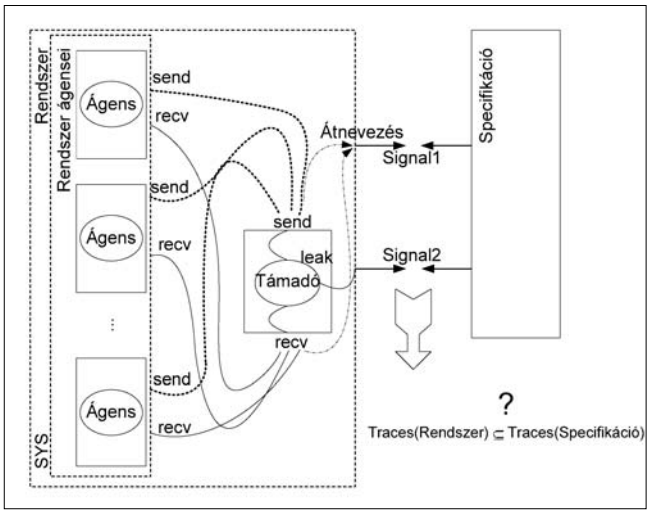
A résztvevők tehát ilyen eseményekkel érintkeznek a külvilággal. A küldés rendszerint egyértelműen adott, ám üzenet érkezése esetén gyakran előzőleg ismeretlen elemek is megjelennek a protokollfutamban. Az ilyen új adattagok típusuknak megfelelően bármilyen új értéket felvehetnek, így itt választásról van szó. Ilyen, halmaz elemei közötti választást a replikált külső választás operátor határozza meg.

A résztvevők egymástól független működésűek: a *SYS*_0 processz ezért az ágens-halmaz *(Agent)* elemeinek összefésüléssel vett egyesítéséből adódik. Ebben a küldés és fogadás események még összehangolatlanul, tetszőlegesen történhetnek meg. A kommunikáció rendjét a médium állítja elő. A *Dolev-Yao* támadó, vagyis az ellenséges médium egyben maga az összekapcsoló processz *(INTRUDER)*.

$$SYS\_0 \mathrel{\hat{=}} |||\ A : Agent \bullet A$$
$$SYS \mathrel{\hat{=}} SYS\_0\ ||_{\{send,recv\}}\ INTRUDER$$

A támadó processz meglehetősen bonyolult felépítésű. Kezdeti tudáshalmazzal rendelkezik, melyet új információkkal bővít. A lehallgatott üzenetek szétbontása után különböző dedukciós szabályokkal következtetéseket végezhet és új elemeket állíthat elő. Belső bonyolultsága ellenére kívülről is látható eseményei egyszerűen olyanok, melyekkel a többi ágens eseményeivel érintkezhet. A belső működés pusztán az analízis sebességének optimalizálása szempontjából fontos.

1. ábra Rendszerünk ellenőrzésének modellje



E Dolev-Yao médium tehát az ágensek tetszőleges üzenetküldés *(send)* eseményeivel bővíti tudáshalmazát és felajánlja az összes olyan fogadás *(recv)* eseményt, amely ezek alapján megvalósulhat. Amikor egy ágens üzenetet küld, a legegyszerűbb eset, hogy az módosítatlanul a valós címzettjéhez jut el. De a támadó felajánlja a csomag fogadását az összes többi ágensnek is. Ráadásul ha a támadó képes más ismereteiből szabályaival ugyanolyan üzenetstruktúrát képezni, a módosított csomagok fogadása is bekövetkezhet.

Az előállt rendszer tehát tartalmazza a támadóval megzavart kommunikációs folyamat összes lehetséges eseménysorozatát. A feladat ennek alapján leellenőrizni, hogy a protokoll összes kimenetele megfelelő biztonsági tulajdonságokat hordoz-e. Tehát olyan specifikációul szolgáló processzt kell készíteni, amely csak biztonságos állapotokat tartalmaz.

E cikkben csak a titkosság és autentikáció ellenőrzésére hagyatkozunk, részletesebben a [7]-ben találhatunk erről információkat. Az alkalmazott modellben a feltételezett titok kiszivárgását a támadó egy speciális eseménnyel jelezheti, ilymódon az ellenőrzés adott esemény egzisztenciájának kérdése. Autentikáció ellenőrzése során arról kell megbizonyosodni, hogy egy elem forrás általi kibocsátása valóban megelőzte-e annak vételét. Emódon itt események egymásutániságának ellenőrzéséről kell megbizonyosodnunk.

A protokoll biztonsági ellenőrzése finomítás-ellenőrzésből áll, vagyis a *specifikáció* $\sqsubseteq_T$ *implementáció* ellenőrzéséből. Ezen ellenőrzés a rendelkezésre álló FDR2 modell-ellenőrzővel automatikusan elvégezhető.

A biztonsági analíziséhez szükséges modell elkészítése hosszadalamas folyamat, mely számos hibázási lehetőséggel jár. De mivel a modellezési alapelvek különböző protokollok esetén is hasonlóak maradnak,

a folyamat tovább automatizálható. A CASPER (Compiler for the Analysis of Security Protocols) fordító képes egy biztonsági protokollból és specifikációinak viszonylag egyszerű szabványos leírásából [8] automatikusan elkészíteni az ellenőrzés alapjául szolgáló CSP leírást.

A bemenetül szolgáló fájl első része a protokoll változóit, az üzenetek leírását és a specifikációkat tartalmazza. A második rész a vizsgálandó véges rendszert, valamint a támadó kezdeti tudását definiálja. A CASPER fordító a modell ellenőrzés eredményeképp kapott eseménysorozatok értelmezésében is segítséget nyújt.

A CASPER fordító klasszikus rendszerek kulcs csere protokolljainak egyszerű és gyors analízisére született és az ismertetett rendszermodellt használja. A következőkben szenzorhálózatokhoz javasolt protokollok biztonságának vizsgálatára fogjuk e keretrendszert alkalmazni, majd ezután megvizsgáljuk a kiterjeszthetőség további lehetőségeit.

5. A SNEP protokoll analízise

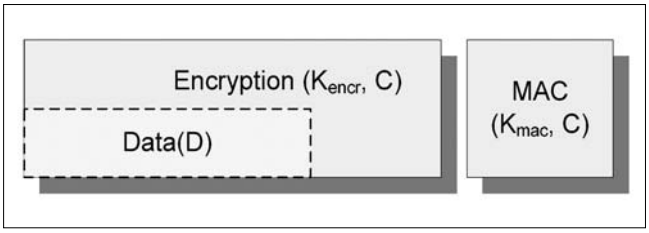
A szenzorhálózatokban, az érzékeny információk védelmét a szűkös erőforrások figyelembevételével kell biztosítani. A *SNEP (Sensor Network Encryption Protocol)* hivatott garantálni [9], hogy a kommunikáló felek között átküldött információ titkos, hiteles, és friss legyen. Ezzel akadályozza meg, hogy rosszindulatú felek hozzájuthassanak vagy módosításokat végezhessenek az átküldött adatokon. Működése feltételezi egy bázisállomás meglétét, amely minden egyes szenzorral osztott kulccsal rendelkezik. Ilyen bázisállomás gyakran része a mai szenzorhálózat-implementációknak.

A mechanizmus osztott kulcsokon alapszik. A protokoll először a közös kulcsból származtatott K_{enc} kulccsal szimmetrikus kulcsú titkosítást végez, majd egy másik származtatott (K_{mac}) kulcsot üzenet hitelesítő *(Message Authentication Code – $MAC_{K_{mac}}$)* kód képzésére használ *(2. ábra)*. Az üzenetek frissességét egy, mindkét fél által karbantartott számláló, illetve erősebb kritérium esetén egyszer használatos *nonce* elem valósít meg.

A protokoll, mely vizsgálatunk alapjául szolgált a következő:

1. $B \rightarrow A : N_b, R_b$
2. $A \rightarrow B : \{D\}\{K_{enc}, C\}, MAC_{K_{mac}, C}(N_b, \{D\}\{K_{enc}\})$

2. ábra A SNEP mechanizmus



E leírásban *B* a bázismállomás, *A* a szenzor. Első lépésben a bázis a szenzornak elküldi a frissesség alapjául szolgáló N_b nonce tagot, valamint a lekérdezést inicializáló R_b azonosítót. A kérésnek megfelelően a válaszul szolgáló információt (*D*) a szenzor az ismertetett védelemmel küldi vissza, mely során a kulcsokon kívül az aktuális *C* számlálót alkalmazza. A $\{D\}\{K_{enc},C\}$ tag a titkosított adat, míg az üzenet második tagja a generált lenyomat (MAC).

A CASPER fordító által generált modell elegendő az analízis elvégzéséhez. Az üzenetváltások az imént megadotthoz hasonló formátumban kerülhetnek megadásra. Az analízis során nem szerepeltettük az említett számlálót, hiszen a nonce által nyújtott frissesség jóval erősebb tulajdonság. A protokoll specifikációját az információ titkossága és hitelessége képezi.

Nézzük most meg a protokoll ágenseinek CSP leírását, hogy megvizsgájunk egy valódi biztonsági protokollt felépítő processzeket:

$$
\begin{aligned}
&Base(B, N_b, R_b, K_{enc}, K_{mac}, A) = \\
&\quad send.B.A.(Msg_1, < N_b, R_b >) \rightarrow \\
&\quad \Box\ D : Message \bullet \\
&\quad recv.A.B.(Msg_2, < \{D\}\{K_{enc}, C\}, \\
&\qquad\qquad MAC_{\{K_{mac}, C\}}(N_b, \{D\}\{K_{enc}\}\ ) >) \rightarrow \\
&\quad Skip
\end{aligned}
$$

$$
\begin{aligned}
&Sensor(A, D, K_{enc}, K_{mac}, B) = \\
&\quad \Box\ N_b : Nonce \bullet \Box\ R_b : Message \bullet \\
&\quad recv.B.A.(Msg_1, < N_b, R_b >) \rightarrow \\
&\quad send.A.B.(Msg_2, < \{D\}\{K_{enc}, C\}, \\
&\qquad\qquad MAC_{\{K_{mac}, C\}}(N_b, \{D\}\{K_{enc}\}\ ) >) \rightarrow \\
&\quad Skip
\end{aligned}
$$

Mindkét ágens esetében az első paraméter a saját azonosító, majd a továbbiak az egyéb szükséges információk. A leírásban a protokoll korábbi leírásának megfelelő változónevek szerepelnek. A processzek a következő módon viselkednek.

A bázis ágens első eseménye egy küldés *(send)*, mellyel az ágens egy elágazáshoz érkezik. A következő fogadás *(recv)* esemény *D* adateleme, vagyis a mérés eredménye, tetszőleges lehet. Ezt modellezi a $\Box\ D : Message \bullet$ külső választás, mely szerint az eseményben szereplő *D* adatelem a *Message* halmaz bármely eleme lehet. Ezzel a protokoll sikeresen terminálódik, vagyis a *Skip* állapotba kerül. A szenzor ágens hasonló elvek alapján működik, az előbbiek alapján az végigkövethető.

Az analízis során támadási lehetőséget nem találtunk, több résztvevőre és protokollfutamok különféle kombinációira is elvégezve az ellenőrzést állíthatjuk hogy a protokoll nem megtéveszthető. Kisebb módosításokkal különleges, nem üzemszerű esetekre is elvégeztük az analízist. Ilyen speciális eset volt a nem megfelelő nonce tag használata. A *nonce* specifikációjának megfelelően egyszer használatos, véletlen és előre nem jósolható véletlen számra van szükségünk. A szenzorokba implementált véletlenszám-generátor gyengesé-

ge fenyegetést jelent. Vizsgálatunk ugyanis kimutatta, hogy rossz nonce esetén üzenetvisszajátszáson alapuló támadás lehetséges.

Megvizsgáltuk továbbá a kulcsok kompromittálódásának hatásait is. A szenzorok fizikai védelmének hiányában fizikai hozzáférés esetén a kulcsok kinyerhetőek az egységekből. A modellben a támadó kiindulási tudáshalmazához hozzáadtuk a K_{mac}, illetve K_{enc} kulcsokat. A K_{mac} kulcs egyedüli megszerzésével a támadó nem jut új képességhez, a K_{enc} kulccsal azonban felfedheti a titkos adattagokat. A hitelesség csak mindkét kulcs birtoklása esetén sérül, ilyenkor a *D* adattag megszerezhető és megváltoztatható. A K_{mac} kulcs azonban csak tökéletes kriptográfia esetén felesleges, hiszen egy valóságos titkosítás nem szükségszerűen biztosítja az integritásvédelmet is.

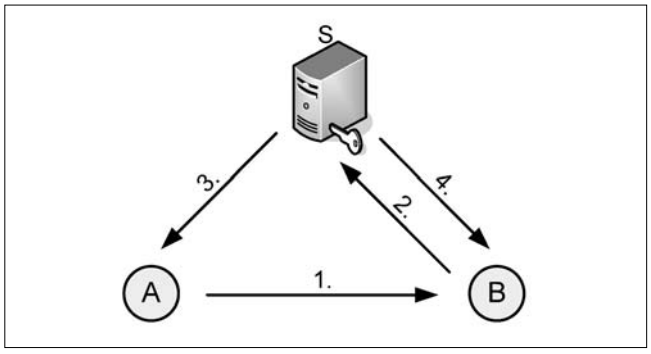
6. Szenzor kulcs-csere protokoll

Mivel a szenzorok biztonságos kommunikációja gyakran osztott kulcsok kialakítását igényli, ennek megvalósítására [9] nézzük a következő kulcs-csere protokollt. A két szenzor (*A* és *B*) egy szerver segítségével hozza létre (például a létező bázisállomás) az osztott titkot, mellyel külön-külön kulccsal (K_{AS} és K_{BS}) rendelkeznek. A protokoll lépései a következőek *(3. ábra)*:

1. $A \rightarrow B : N_A, A$
2. $B \rightarrow S : N_A, N_B, A, B, MAC_{K_{BS}}(N_A, N_B, A, B)$
3. $S \rightarrow A : \{SK_{AB}\}_{K_{AS}}, MAC_{K_{AS}}(N_A, B, \{SK_{AB}\}K_{K_{AS}})$
4. $S \rightarrow B : \{SK_{AB}\}_{K_{BS}}, MAC_{K_{BS}}(N_B, A, \{SK_{AB}\}_{K_{BS}})$

Az első lépésében az *A* szenzor kezdeményezi a protokollt, melyben egy N_A nonce tagot használ. A megszólított *B* ezután a szerverhez fordul, ahol N_B egy friss nonce és a K_{BS} a hitelesítő tag (MAC) kulcsa. A protokoll utolsó két lépése osztja ki az új kulcsot (SK_{AB}) a részvevőknek.

3. ábra Szenzor kulcs-csere



A protokoll specifikációja szerint e lépések sikeres befejezéssel a két szenzor hiteles és titokban tartott SK_{AB} kulcsokkal rendelkeznek.

Az analízis támadási lehetőséget mutatott, mely olyan esetben áll fenn, amikor *A* és *B* szenzor is kezdeményezheti a protokollt a másikkal egyidejűleg. Ez valódi fenyegetést jelent, hiszen a valóságban az *A* és *B* szenzorok egyenrangúak, nincsenek rögzített kezdeményezők.

$\alpha 1.\ I(B) \rightarrow A : N_M, B$
$\alpha 2.\ A \rightarrow I(S) : N_M, N_A, B, A, MAC_{K_{AS}}(N_M, N_A, B, A)$
$\beta 1.\ I(A) \rightarrow B : N_A, A$
$\beta 2.\ B \rightarrow S : N_A, N_B, A, B, MAC_{K_{BS}}(N_A, N_B, A, B)$
$\beta 3.\ S \rightarrow I(A) : \{SK_{AB}\}_{K_{AS}}, MAC_{K_{AS}}(N_A, B, \{SK_{AB}\}_{K_{AS}})$
$\alpha 4.\ I(S) \rightarrow A : \{SK_{AB}\}_{K_{AS}}, MAC_{K_{AS}}(N_A, B, \{SK_{AB}\}_{K_{AS}})$

A támadás leírásában két protokollfutam (α és β) összefésülését láthatjuk. A leírásban $I(A)$, illetve $I(S)$ jelenti az *A* és *S* résztvevőket megszemélyesítő támadót. A támadó a N_M nonce taggal kezdeményezi a protokollt, majd úgy juttatja el az fi futamot a sikeres befejezéshez, hogy közben a másik résztvevő valójában nem is vett részt abban a futamban. A 3. és 4. protokoll lépés lenyomatát kiegészítve a másik résztvevő nonce-ával is, a támadási lehetőség megszűnik, így a protokoll javítható.

Az előző két protokoll jó példaként szolgált biztonsági analízisünk alapjainak demonstrálására. A fenti támadási lehetőség felderítése lehetőséget adhat a hiba javítására. A SNEP protkoll esetében láthattuk, hogy a modell elég rugalmas ahhoz, hogy speciális eshetőségek (például kulcs kompromittálódás) hatásait is figyelembe vegyük.

A következőkben vizsgáljuk meg a modell nyújtotta további lehetőségeket, különös tekintettel a szenzor és ad hoc hálózatok dinamikus jellemzőire.

7. További modellezési lehetőségek

A CASPER fordító és az ismertetett modell kulcs-csere protokollok analízisére született. Ezekben, mint láttuk, minden egyes ágens meghatározott szereppel rendelkezik a protokollban. E szerep határozza meg az ágens viselkedését, az általa fogadható üzenetek és reakcióinak determinálásával.

A szenzor és ad hoc hálózatok résztvevői azonban rendszerint egyenrangú felekként viselkednek. A résztvevők egyformák, aktuális szerepük a környezettől (például szomszédok) függően alakul ki. A CSP keretrendszer lehetőséget ad ilyen dinamikus rendszerek modellezésére is.

A résztvevő ágensek azonos ismeretekkel rendelkeznek, azonban többféle üzenet vételére is képesek, ez határozza meg további működését. Mivel a vett üzenetek eseményekként jelennek meg, így események egy véges halmaza ($event_1,...,event_n$) közötti választásról van szó, melyek az ágenst szerepének megfelelő állapotba ($role_1,...,role_n$) juttatja. Ez a külső választás operátorral írható le, egy ilyen *Node* leírása a következő formájú:

$Node(A,...) =$
$(event_1 \rightarrow role_1) \Box (event_2 \rightarrow role_2) \Box ... \Box (event_n \rightarrow role_n)$

Így modellezhetővé és analizálhatóvá válnak például a biztonságos útvonalválasztó mechanizmusok. Az analízisben végrehajtásában problémát okoz a sok választás, így szerteágazó trace, okozta hatalmas állapottér. E állapottérrobbanás megelőzésére különféle optimalizálási technikák alkalmazhatóak, melyek a jövőben lerövidíthetik az analízist.

8. Összegzés

E cikkben megismerkedhettünk a szenzor és ad hoc hálózatok tulajdonságaival. Láthattuk a jellemzőikből és önszerveződő természetükből eredő biztonsági fenyegetéseiket. A CSP alapjaiba nyert betekintő után láthattuk, hogy miként alkalmazható az biztonsági protokollok analízisére. Láthattuk a módszer alkalmazását két, szenzorhálózatokhoz javasolt biztonsági protokollon és rámutattunk a módszer további lehetőségeire is.

Irodalom

[1] Jason Hill, Robert Szewczyk, Alec Woo, Seth Hollar, David E. Culler, Kristofer S. J. Pister (2000): System Architecture Directions for Networked Sensors

[2] Gémesi Roland, Ivády Balázs, Zömbik László: Mobil ad hoc hálózatok biztonsága. Híradástechnika, 2002/12.

[3] Buttyán Levente, Holczer Tamás, Schafier Péter: Kooperációra ösztönző mechanizmusok többugrásos vezeték nélküli hálózatokban. Híradástechnika, 2004/3.

[4] Hubert Comon, Vitaly Shmatikov: Is it possible to decide whether a cryptographic protocol is secure or not? (2001)

[5] Peter Ryan, Steve Schneider, Michael Goldsmith, Gavin Lowe, Bill Roscoe: Modelling and analysis of security protocols (2001)

[6] Steve Sneider: Concurrent and Real-time Systems. The CSP approach (2000)

[7] Peter Ryan, Steve Schneider, Michael Goldsmith, Gavin Lowe, Bill Roscoe: Modelling and Analysis of Security Protocols (2000)

[8] Gavin Lowe, Philippa Broadfoot, Mei Lin Hui: A Compiler for the Analysis of Security Protocols (2001)

[9] Adrian Perrig, Robert Szewczyk, Victor Wen, David Culler, J. D. Tygar: SPINS: Security Protocols for Sensor Networks (2001)

Hírek

A Sun Microsystems megalakította Globális Kormányzati Irodáját (Global Government Office), melynek feladata, hogy speciális megoldásokat dolgozzon ki a világ kormányzati szerveinek IT problémáira. A kormányzati szektornak szánt biztonságos megoldások központjában a Trusted Solaris operációs rendszer áll, amely az amerikai kormányzaton belül már gyakorlatilag platformszabványnak számít. Ez beépített hálózati biztonságot kínál: a személyes adatok védelmét, jobb elszámoltathatóságot, valamint a biztonsági rendszer sérülésének alacsonyabb kockázatát.

A Sun annak érdekében, hogy a kevésbé fejlett országok kormányzatait is kiszolgálhassa, speciális, az állampolgárok számát alapul vevő árképzési modellt jelentett be a Java Enterprise Systemre vonatkozóan. A rendszer ára polgáronként évi 0,33 dollártól 1,95 dollárig terjed, pontos értékét pedig két tényező határozza meg: az ország fejlettségi szintje az Egyesült Nemzetek meghatározása szerint, valamint az adott kormányzati egység alá eső polgárok száma.

A Sun Microsystems a Project Looking Glasst és Java 3D technológiáit a nyílt forráskódú közösség rendelkezésére bocsájtja, ezzel is megerősítve elkötelezettségét az élenjáró asztali technológiák mellett. Emellett két további, a Java fejlesztői közösségével együttműködve készülő nyílt forráskódú rendszert is bejelentett: a JDesktop Network Components és JDesktop Integration Components rendszereket. A JDNC leegyszerűsíti a gazdag funkcionalitású, hálózatos asztali alkalmazások fejlesztését, a JDIC pedig integrálja a többplatformos Java alapú alkalmazásokat a natív asztali környezettel.

A Projekt Looking Glass háromdimenziós asztali környezet fejlesztésére irányuló projekt, amelynek innovatív felülete intuitív 3D-környezetet kínál, ahol az ablakok átlátszóak, forgathatóak, lapozhatóak és tetszés szerint méretezhetők. A Project Looking Glass fejlesztői kiadásában az alábbi funkciók értetők el:

– 3D ablakkezelő platform, segítséget nyújt a fejlesztőknek a dokumentumok megtervezésében, a kezdeti specifikációkban és a prototípusok megvalósításában,
– Natív alkalmazásintegrációs modulok – Modul X11 alkalmazások a 3D környezetben futtatásához,
– Minta 3D ablakkezelő – tesztelési és demonstrációs célokra.

Információátvitel nagy sebességű közegek között

DR. CSERNOCH JÁNOS

Budapesti Műszaki Főiskola, Kandó Kálmán Villamosipari Kar, Híradástechnikai Intézet
csernoch.janos@kvk.bmf.hu

1. Síkhullámok visszaverődése mozgó közeg határfelületén

Tételezzük fel, hogy a síkhullám a *K* koordinátarendszer X tengelye mentén pozitív X irányban terjed. A hullámforrás a *K* koordinátarendszerben valahol a negatív végtelenben van, tehát a síkhullám hullámfrontja az X tengelyre merőleges. Tudjuk, hogy a fény terjedési sebessége vákuumban minden koordinátarendszerben azonos.

A *K'* koordinátarendszerről feltételezzük azt, hogy annak O' origója *K* koordinátarendszer X tengelye mentén annak pozitív irányában *v* egyenes vonalú egyenletes sebességgel mozog oly módon, hogy a megfelelő koordinátatengelyek egymással párhuzamosak:

$$X \,||\, X'$$
$$Y \,||\, Y'$$
$$Z \,||\, Z'$$

A *K'* koordinátarendszer x'=0 (tehát Y'Z') síkjáról tételezzük fel, hogy visszaverő felületként szerepel oly módon, hogy a pozitív X' tengely által meghatározott térrészt (az Y'Z' síktól jobbra) $\varepsilon_r>1$ relatív dielektromos, állandójú veszteségmentes anyag tölti ki, továbbá azt, hogy a szóban forgó síktól balra eső rész vákuum.

A *K'* koordinátarendszerbeli *M'* megfigyelő teljesen szabályszerű visszaverődést, illetve visszaverődési tényezőt tapasztal, melynek értéke

$$p'=\frac{\sqrt{\frac{\mu}{\varepsilon}}-\sqrt{\frac{\mu_0}{\varepsilon_0}}}{\sqrt{\frac{\mu}{\varepsilon}}+\sqrt{\frac{\mu_0}{\varepsilon_0}}}=\frac{\frac{1}{\sqrt{\varepsilon_r}}-1}{\frac{1}{\sqrt{\varepsilon_r}}+1}=-\frac{\sqrt{\varepsilon_r}-1}{\sqrt{\varepsilon_r}+1}\prec 0$$

Ez a nyugalmi reflexiós tényező.
Itt a dielektrikum hullámimpedanciája

$$Z'_H=\frac{E'_Y}{H'_Z}=\sqrt{\frac{\mu}{\varepsilon}}$$

és a vákuum hullámellenállása

$$Z_0=\sqrt{\frac{\mu_0}{\varepsilon_0}}=120\pi ohm$$

(Ellenkező esetben az *M'* megfigyelő valahogyan megtudná, hogy a *K'* koordinátarendszer mozog! Tudjuk azt, hogy a jel sűrűbb közeg határáról 180°-os fázisban verődik vissza. – *1. ábra*)

A kérdés itt az, hogy milyen reflexiós tényezőt tapasztal a *K* koordinátarendszerben levő *M* megfigyelő? A dielektrikum határfelületén tapasztalt hullámimpedancia számítási erdményét anyagjellemzőkkel átírva a következőket kapjuk:

$$Z_H=\sqrt{\frac{\mu}{\varepsilon}}\left(\frac{1+\frac{\beta}{\sqrt{\varepsilon_r\mu_r}}}{1+\beta\sqrt{\varepsilon_r\mu_r}}\right), \qquad (1)$$

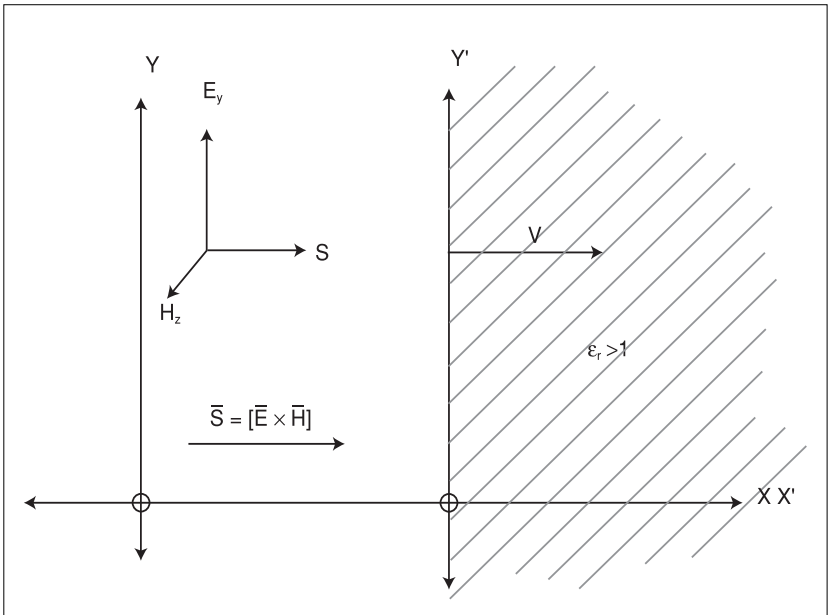
ahol $\beta = v/c$
$c = 2{,}9987.10^8$ m/sec = 3.10^8 m/sec

Itt a következő esetek lehetségesek:

a.) Abban az esetben, ha $v > 0$, illetve $\beta > 0$, tehát a közeg a hullámforrástól távolodik, akkor a *K* koordinátarendszerben mért sugárzási ellenállás

$$Z_H=\sqrt{\frac{\mu}{\varepsilon}}\left(\frac{1+\frac{1}{\sqrt{\varepsilon_r\mu_r}}\beta}{1+\sqrt{\varepsilon_r\mu_r}\beta}\right)\leq Z'_H=\sqrt{\frac{\mu}{\varepsilon}}$$

mert $\varepsilon_r\mu_r > 1$.



1. ábra Visszaverődés mozgó dielektrikumon

b.) Nyugalmi állapotban $v = 0$, illetve $\beta = 0$, mint ahogy ezt vártuk.

$$Z_H = \sqrt{\frac{\mu}{\varepsilon}} = Z'_H$$

c.) Abban az esetben, ha $v < 0$, illetve $\beta < 0$, tehát a kérdéses közeg a hullámforráshoz közeledik, akkor a *K* koordinátarendszerben mért hullámellenállás nagyobb, mint a *K'* rendszerben mért.

$$Z_H = \sqrt{\frac{\mu}{\varepsilon}}\left(\frac{1+\frac{1}{\sqrt{\varepsilon_r\mu_r}}\beta}{1+\sqrt{\varepsilon_r\mu_r}\beta}\right) \geq Z'_H = \sqrt{\frac{\mu}{\varepsilon}} \qquad (2/a)$$

Érdemes itt elvben megjegyezni azt, hogy

$$\sqrt{\mu_r\varepsilon_r}\beta = -1$$

esetén $Z_H \to \infty$ (Totális reflexió a *K* rendszerben!)

Feltételezve azt, hogy

$\varepsilon_r = 100$ és $\mu_r = 1$

$\beta = -10^{-1}$

Ennek a technika mai állása szerint gyakorlati jelentősége nincsen, mert ilyen nagy sebesség esetén az átvitt információnak igen nagy torzulásával kellene számolni. ($\varepsilon_r > 100$ igen ritkán fordul elő!)

Nem ferromágneses anyag esetén jó közelítéssel érvényes az, hogy $\mu_r \approx 1$. Ennek figyelembevételével a közegnek a *K* koordinátarendszerben mért hullámellenállása $\beta \leq 10^{-2}$ esetén, jó közelítéssel

$$Z_H = \sqrt{\frac{\mu_0}{\varepsilon}}\left[1+\left(\frac{1}{\sqrt{\varepsilon_r}}-\sqrt{\varepsilon_r}\right)\beta\right] \qquad (2/b)$$

A közelítés hibája $|\beta| \leq 10^{-2}$ sebességi hányadot feltételezve kisebb, mint 1,5%.

A *K* koordinátarendszerbeli *M* megfigyelő által észlelt reflexiós tényező

$$p = \frac{Z_H - Z_0}{Z_H + Z_0} = -\frac{\left(1-\frac{1}{\sqrt{\varepsilon_r}}\right)+\left(1-\frac{1}{\sqrt{\varepsilon_r}}\right)\beta}{\left(1+\frac{1}{\sqrt{\varepsilon_r}}\right)-\left(1-\frac{1}{\sqrt{\varepsilon_r}}\right)\beta} \qquad (3)$$

A reflexiós tényező $|\beta| \leq 10^{-1}$ esetén

$$p \approx p'[1+2\beta] \qquad (4)$$

Ez a formula kényelmesen használható és gyors közelítő becslésre kitűnően alkalmas. A közelítés hibája kisebb, mint 2%.

Tanulságok:

1.) $|\beta| \leq 10^{-2}$ esetén a reflexiós tényező értéke kézben tartható határok között változik (és a hiba kisebb, mint 2%):

– $\beta > 0$, tehát távolodás esetén a reflexiós tényező értéke a nyugalmi állapothoz képest növekszik,

– $\beta < 0$, tehát közeledés esetén a reflexiós tényező értéke a nyugalmi állapothoz képest csökken.

2.) $10^{-2} \leq |\beta| \leq 10^{-1}$ esetén a reflexiós tényező értéke szélesebb határok között változhat. (Erre a hibaszámítás során megállapított feltételek alapján lehet egyértelműen következtetni.)

3.) A reflexiós tényező jó közelítéssel (4. egyenlet) felírt képlete növekményében (2β) nem tartalmazza a dielektromos állandót. Ez azért fontos, mert a közeg relatív dielektromos állandója változik a *v* sebességével. Ez a változás azonban a β szorzótényezővel együtt csak a hibaszámításban, de elhanyagolható mértékben jelentkezik.

Így a reflexiós viszonyokat a 4. képlet 2% pontossággal, a relatív dielektromos állandó említett változásától függetlenül, a gyakorlati követelményeknek megfelelően jól írja le.

2. Síkhullámok mozgó dielektrikumban

Törésmutató megváltozása

Tágabb értelemben akkor beszélhetünk síkhullámokról, ha a

D	és	D^X
=		=
D^M	és	D^{MX}
=		=

mátrix minden eleme egyenesen arányos a

$$\Phi = 2\pi f\left(t - \frac{x\cos\alpha_x + y\cos\alpha_y + z\cos\alpha_z}{c_k}\right) \qquad (5)$$

fázisszög koszinuszával.

Itt f = frekvencia

$c_k = \frac{1}{\sqrt{\varepsilon\mu}}$

c_K = a fény terjedési sebessége a közegben, melynek anyagállandói ε és μ

$\cos\alpha_X$,
$\cos\alpha_Y$,
$\cos\alpha_Z$ = a síkhullám frontja normálisának az iránykoszinuszai

Mindezen adatok a *K* „nyugvó” koordinátarendszerben érvényesek. Mozogjon a *K'* koordinátarendszer O' origója a *K* koordinátarendszer X tengelye mentén pozitív irányban. (Egyesvonalú egyenletes mozgás!)

A fázis a koordinátarendszerektől független invariáns skalár mennyiség (6):

$$\Phi = 2\pi f\left(\frac{jct}{jc} - \frac{x\cos\alpha_x + y\cos\alpha_y + z\cos\alpha z}{c_k}\right)$$

$$\Phi = \Phi' = 2\pi f'\left(\frac{jct'}{jc} - \frac{x'\cos\alpha_x' + y'\cos\alpha_y' + z'\cos\alpha_z'}{c_k}\right)$$

$$\Phi = \Phi' = -2\pi f' \left(\frac{j}{c} jct' + \frac{x'\cos\alpha_x' + y'\cos\alpha_y' + z'\cos\alpha_z'}{c_k} \right)$$

A megfelelő koordináták a négy dimenzióban:

$$x = x\ (x_1 = x, y_1 = y, z_1 = z)$$

A transzformáció mátrixa

$$\alpha = \begin{vmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} & \alpha_{13} \\ \alpha_{21} & \alpha_{22} & \alpha_{23} & \alpha_{24} \\ \alpha_{31} & \alpha_{32} & \alpha_{33} & \alpha_{34} \\ \alpha_{41} & \alpha_{42} & \alpha_{43} & \alpha_{44} \end{vmatrix} = \begin{vmatrix} \kappa & 0 & 0 & j\kappa v/c \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ j\kappa v/c & 0 & 0 & \kappa \end{vmatrix}$$

ahol a

$$\kappa = \frac{1}{\sqrt{1-\beta^2}}$$

A fázisszög skalár invariáns és ennél fogva felírhatjuk az 1_i vektort. Ügyelni kell azonban arra, hogy a fázisszögben a c_k nem emelhető ki. Ez a számítást a vákuum esetéhez viszonyítva kissé bonyolítja, de mint látni fogjuk a problémát kellő biztonsággal lehet kezelni és a megfelelő következtetéseket, tanulságokat levonhatjuk.

Tehát a hullám-vektor komponensei a *K* koordinátarendszerben

$$l_1 = \frac{f\cos\alpha_x}{c_k}$$

$$l_2 = \frac{f\cos\alpha_y}{c_k} \qquad (7)$$

$$l_3 = \frac{f\cos_z}{c_k}$$

$$l_4 = \frac{jf}{c}$$

A hullám-vektor komponenseire a *K'* koordinátarendszerben, melyet most „nyugvó” koordinátarendszernek fogunk nevezni, ugyanezen összefüggések vesszős alakja érvényes.

A c_K a hullám fázissebessége a *K'* koordinátarendszerben és ennél fogva a közeg abszolút törésmutatója, melyet a *K'* koordinátarendszer O' origójában levő *M'* megfigyelő mér.

$$n' = n_0 = \frac{c}{c_k'}$$

Ezt nyugalmi törésmutatónak nevezzük.

A közeg abszolút törésmutatója, melyet a *K* koordinátarendszerben levő *M* megfigyelő mér:

$$n = \frac{c}{c_k}$$

Mivel

$$-\frac{\Phi}{2\pi} = (\bar{l}\bar{x}) = l_i x_i = l_1 x_1 + l_2 x_2 + l_3 x_3 + l_4 x_4$$

invariáns skalár mennyiség, ezért az 1_i és az $1_{i'}$ hullám-vektor a Lorentz-transzformáció szerint transzformálódik:

$$\bar{l}' = \underline{\underline{\alpha}}\,\bar{l}$$

A *K* koordinátarendszerben észlelt frekvencia a transzformáció elvégzése után:

$$f = \kappa f' \left[1 + \frac{v}{c_k'} \cos\alpha_x' \right] \qquad (8)$$

A *K'* rendszerben az *M'* megfigyelő által észlelt frekvencia inverz transzformációval

$$f' = \kappa f \left[1 - \frac{v}{c_k} \cos\alpha_x \right] \qquad (9)$$

Az

$$n' = n_0 = \frac{c}{c_k}$$

„nyugalmi” törésmutató és az

$$n = \frac{c}{c_k}$$

mozgási törésmutató közötti összefüggést most már felírhatjuk.

A *K* koordinátarendszerben mért „mozgási” törésmutató négyzete a számítás elvégzése után:

$$n^2 = \frac{(\beta + n_0\cos\alpha_x')^2 + n_0^2(1-\beta^2)\sin^2\alpha_x'}{\left[1 + n_0\beta\cos\alpha_x'\right]^2} \qquad (10)$$

Ha $\alpha_x = 0$ $\alpha_{x'} = 0$

$$n_0^2 = \left(\frac{n-\beta}{1-n\beta} \right)^2 \qquad n^2 = \left(\frac{n_0+\beta}{1+n_0\beta} \right)^2$$

Ha $\alpha_x = \pi$ $\alpha_{x'} = \pi$

$$n_0^2 = \left(\frac{n+\beta}{1+n\beta} \right)^2 \qquad n^2 = \left(\frac{n_0-\beta}{1-n_0\beta} \right)^2$$

Speciális esetek

Két speciális esetet vizsgálunk meg.

1.) $\alpha_x = 0$ esetén, amikor a hullámfront normálisának az iránya egybeesik a *K* közeg mozgásának az irányával. A *K* rendszerben mért törésmutató

$$n(\beta) = \frac{\beta + n_0}{1 + \beta n_0}$$

(*K'* rendszerben mért törésmutató n_0).

A négyzetgyökvonás után, csak a pozitív előjelet vesszük figyelembe, mert a negatív előjelnek nincsen fizikai értelme.

Következtetések

a.) A *K* koordinátarendszerbeli *M* megfigyelő kisebb törésmutatót mér, mint a *K'* koordinátarendszerbeli *M'* megfigyelő (n_0) és $n_0^2 > 1$

$$n = \frac{n_0 + \beta}{1 + \beta n_0} = n_0 \left(\frac{1 + \frac{\beta}{n_0}}{1 + \beta n_0} \right) \prec n_0$$

A sebesség tartománya:

$$0 \le \beta \le 1$$

Ha a $v = 0$, illetve $\beta = 0$, akkor $n = n_0$,
mint ahogy azt vártuk.

Ha a $v = c$, illetve $\beta = 1$, akkor $n = 1$.

Tehát a fény terjedési sebességének a közelében a *K* koordinátarendszerben levő *M* megfigyelő szemszögéből nézve, minden anyag egyre inkább „elveszíti" a törésmutatóját. Ennek következtében a lencsék fókusztávolsága megváltozik, (melynek világosan láthatóan semmi köze nincsen a hosszúság-kontrakcióhoz.)

A megváltozott törésmutató:

$$n = n_0 - \beta (n_0^2 - 1) = n_0 + \Delta n$$

A törésmutató megváltozása
(elhanyagolható, max. 0,05%-os hibával számítva)

$$\Delta n = - \beta (n_0^2 - 1)$$

A lencse fókuszképlete

$$\frac{1}{f_F} = (n-1)\left(\frac{1}{R_1} + \frac{1}{R_2} \right)$$

ahol f_F a lencse fókusztávolsága,
R_1, R_2 a lencsék felületeinek görbületi sugarai.

A fókusztávolság relatív megváltozása a törésmutató megváltozása következtében

$$\frac{\Delta f_F}{f_F} = -\frac{\Delta n}{(n_0 - 1)} = \beta (_0 - 1) \qquad (12)$$

A viszonyokat $\beta_2 = 10^{-2}$ és $\beta_3 = 10^{-3}$ esetén táblázatban tüntettük fel.

A táblázatban $D = \frac{100}{f_F\,(cm)}$ dioptriát jelent, mely a cm-ben vett fókusztávolság reciprok értékének a százszorosa.

A törésmutató értéke a *K'* rendszerben legyen

$$n_0 = 1{,}5.$$

D	f_F (cm)	$\beta_2 = 10^{-2}$ $\Delta f_{F2} = \beta_2(1+n_0)f_F$ (mm)	$\beta_3 = 10^{-3}$ $\Delta f_{F3} = \beta_3(1+n_0)f_F$ (mm)
20	5	1.25	0.125
10	10	2.5	0.25
5	20	5	0.5
2.5	40	10	1.0

A *K* koordinátarendszerben levő *M* megfigyelő $\beta = 10^{-3}$ és optikai adó esetén az adóteljesítményszint jelentős süllyedését észlelheti. Az R_1 és az R_2 változását nem számoltuk.

b.) Vákuum esetén $n_0 = 1$

$$n = \frac{1 + \beta}{1 + \beta} = 1$$

Mindkét koordinátarendszerben ugyanaz az $n = 1$ törésmutató mérhető.

c.) A törésmutatónak a frekvenciával való megváltozása, azaz a diszperzió a két koordinátarendszerben hasonló és csak előjelet vált.

Normális diszperzió esetén:

K' koordinátarendszerben $\frac{dn_0}{df'} \succ 0$

K koordinátarendszerben $\frac{dn}{df} = \frac{dn}{df'}\frac{df'}{df} = \frac{dn}{df'} c_{v0} \succ 0$

Anomális diszperzió esetén:

K' koordinátarendszerben $\frac{dn_0}{df'} \prec 0$

K koordinátarendszerben $\frac{dn}{df} = \frac{dn}{df'}\frac{df'}{df} = \frac{dn}{df'} c_{v0} \prec 0$

Tehát a *K'* rendszerben a maximumnak a *K* rendszerben is maximum felel meg, a *K'* rendszerbeli minimumnak pedig a *K* rendszerben szintén minimum felel meg.

d.) A röntgensugarak tartományában a *K'* koordinátarendszerben levő *M'* megfigyelő $n_0 = 1$ törésmutatót lát. Ugyanez a *K* koordinátarendszerben levő *M* megfigyelő szemszögéből nézve

$$n = \frac{1 + \beta}{1 + \beta} = 1$$

e.) A mikrohullámok tartományában a *K'* koordinátarendszerben *M'* megfigyelő nem ferromágneses anyag esetén ($\mu_r \approx 1$ UHF, SHF, EHF tartomány)

$$n_0 = \sqrt{\varepsilon_{rN}'}$$

törésmutatót lát.

Ugyanez a koordinátarendszerben levő *M* megfigyelő szemszögéből nézve

$$n = \sqrt{\varepsilon_{rN}} = \frac{\sqrt{\varepsilon_{rN}}' + \beta}{1 + \beta\sqrt{\varepsilon_{rN}}'} = \sqrt{\varepsilon_{rN}}' \frac{1 + \frac{\beta}{\sqrt{\varepsilon_{rN}}'}}{1 + \beta\sqrt{\varepsilon_{rN}}'} \prec \sqrt{\varepsilon_{rN}}'$$

Tehát a *K* koordinátarendszerbeli *M* megfigyelő kisebb dielektromos állandót észlel. Itt ε_{rN} „alacsony” frekvencián mért dielektromos állandó.

2.) $\alpha_x' = \pi$ esetén, amikor a hullámfront normálisa a *K'* közeg mozgási irányával ellentétes. A következtetések értelemszerűen megfelelnek az 1.) pontban lefektetett következtetéseknek.

Tanulságok:

1.) Amikor a hullámfront normálisának az iránya megegyezik a *K* rendszer mozgásának az irányával, ($\alpha_X = 0$) akkor a *K* rendszerben kisebb törésmutatót mérnek, mint a *K'* rendszerben.

$$n < n_0$$

2.) Amikor a hullámfront normálisának iránya ellenkezik a *K* rendszer mozgásának az irányával ($\alpha_X' = \pi$), akkor a *K* rendszerben nagyobb törésmutatót mérnek, mint a *K'* rendszerben.

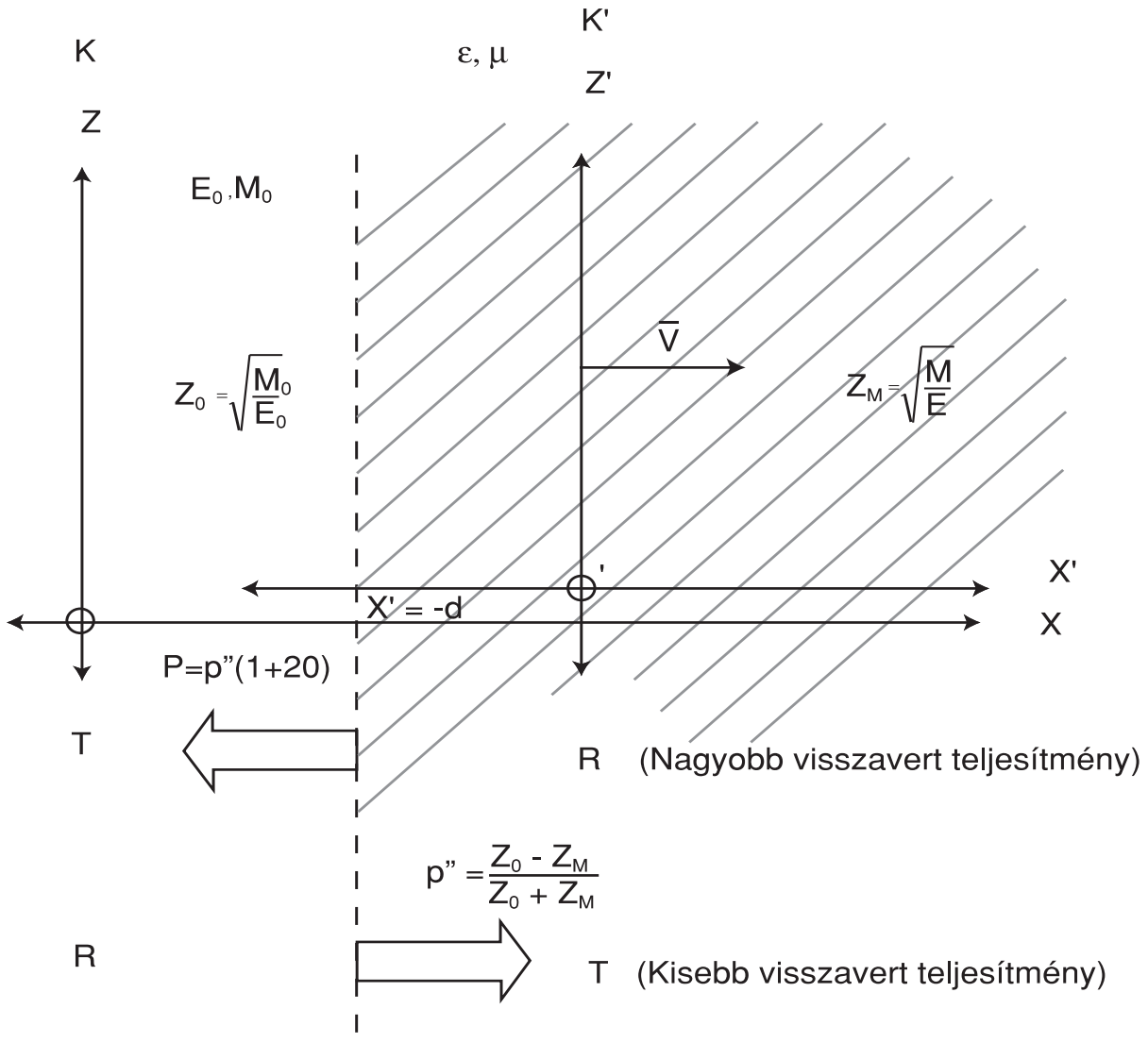
$$n > n_0$$

3.) Mindkét esetben a *K'* rendszerben $\beta \geq 10^{-2}$ mellett a *K'* rendszerben elhelyezett optikai adó defókuszálása lehetséges, aminek a *K* rendszerben szintcsökkenés az eredménye.

4.) Ugyanakkor a két koordinátarendszerben mért törésmutatók diszperziós karakterisztikái hasonlóak.

5.) $\alpha_X' = 0$ és $\alpha_X' = \pi$ esetén a törésmutatók megfelelnek az Einstein-szabálynak. Másirányú hullámfront esetén, (miután a c_k sebességű mozgást elektromágneses hullám végzi) némi eltérés mutatkozik, melynek az említett megállapításokra lényeges befolyása nincsen. Ezzel a kérdéssel itt nem foglalkozunk.

2. ábra Inhomogén tér



3. Reciprocitás kérdése

A két antenna közötti inhomogén

Változtassuk meg a helyzetet a 2. fejezethez viszonyítva úgy, hogy a *K'* koordinátarendszerben az x' = -d síktól jobbra egy ε ,μ és δ_F anyagállandókkal rendelkező, egyébként önmagában homogén közeg foglal helyet és a nevezett síktól balra vákuumot tételezünk fel. (σ_F = a fajlagos vezetőképesség. Ezt az elrendezést a szemléltető példa kedvéért gondoltuk el, és d > 0.)

A többi adat azonos a 2. fejezet adataival.

Abban az esetben, ha a *K'* koordinátarendszer nem végezne mozgást, akkor a *T* adót a *K* koordinátarendszer O origójába helyezve az említett közeg határ

$$p' = \frac{Z_H - Z_0}{Z_H + Z_0}$$

feszültség-reflexiós tényezőt mérnénk, ahol

$$Z_0 = \sqrt{\frac{\mu_0}{\varepsilon_0}} = 120\pi ohm$$

a vákuum sugárzási ellenállása és

$$Z_H = \sqrt{\frac{\mu}{\varepsilon}}$$

a közeg sugárzási ellenállása.

Ugyanakkor a *T* adót a *K'* koordinátarendszer O' origójába helyezve a *K'* koordinátarendszerben

$$p'' = \frac{Z_0 - Z_H}{Z_0 + Z_H}$$

feszültség-reflexiós tényezőt mérnénk.

A két reflexiós tényező abszolút értéke azonos, tehát a teljesítmény-reflexiós tényező mindkét esetben ugyanaz. Így nyugalmi állapotban a reciprocitási tételt érvényesnek kell tekinteni.

Ha most a *K'* koordinátarendszer, mint ahogy az előző fejezetben feltételeztük, távolodik a *K* koordinátarendszer O origójától egyenes vonalú egyenletes sebességgel, akkor a 2. fejezetben leírt feltételek mellett a reciprocitási tétel nem érvényes.

A megfelelő koordinátatengelyek párhuzamosak egymással:

X || X'
Y || Y'
Z || Z'

Ugyanis, ha a *T* adót a *K* koordinátarendszer O origójába helyezzük, akkor az ebben a koordinátarendszerben levő *M* megfigyelő $|\beta| < 10^{-2}$ esetén jó közelítéssel

$$p \approx p' (1 + 2\beta)$$

feszültség-reflexiós tényezőt mér.

Ugyanakkor, ha a *T* adót a *K'* koordinátarendszer O' origójába helyezzük, akkor az itt levő *M'* megfigyelő változatlanul, a hozzá viszonyítva nyugalmi

$$p'' = \frac{Z_0 - Z_H}{Z_0 + Z_H}$$

feszültség-reflexiós tényezőt méri.

Mivel feltételezésünk szerint a *K'* koordinátarendszer a *K* koordinátarendszer O origójától távolodik, ezért β > 0 és

$$|p| > |p'| = |p''| \qquad (13)$$

Ennek az a következménye, hogy az első esetben az O origóból (adó) a *K'* koordinátarendszer O' origójába (vevő) kevesebb teljesítmény jut, mint fordított felállásban.

Tanulság:

Ha az adó és vevő közötti tér inhomogén és a vevő a közeggel együtt egyenes vonalú egyenletes sebességgel mozog, ez esetben a reciprocitási tétel nem érvényes.

Irodalom

[1] Novobácky Károly:
Relativitás elmélet. Egyetemi tankönyv,
Egyetemi Nyomda, Budapest

[2] Albert Einstein:
Über die spezielle und allgemeine Relativitaetstheorie.
Druck und Verlag von
Vieweg und Dohn Braunschweig 1921.

[3] Albert Einstein:
Les fondements de la théorie de la relativité générale.
Librerie scientifique Hermann.

[4] Novobácky Károly:
Elektrodinamika. Egyetemi tankönyv,
Tankönyvkiadó, Budapest 1950.

[5] Simonyi Károly:
Theoretische Elektrotechnik.
Deutscher Verlag der Wissentschaften, Berlin 1979.

[6] Csepeli Miklós, Dr.Selmeczi Kálmán,
Tóthné Szemes Marianne:
Műszaki fizika I. (Főiskolai jegyzet)

[7] Richard P. Feinmann, Robert B.Leighton,
Matthew Sands:
Mai fizika. Massachusetts, USA

Tájékoztatás a Híradástechnika szerzőinek

A Híradástechnika szerkesztőbizottsága szeretné, ha egyre több szerzője lenne különböző területekről, így tovább bővülne az újságban megjelenő témák köre, és változatosabbá válna az eltérő szemléletű szerzők gondolatvilágától. Leendő szerzőink számára a cikkírással kapcsolatban szeretnénk néhány tájékoztató gondolatot közölni:

• **Témák:** Az újságban elsősorban a híradástechnika szakmai újdonságait szeretnénk közzétenni. Eszerint a távközlés, a műsorszórás, továbbá a teleszolgálatok minden területe és a velük kapcsolatos témák érdekesek. Tehát egyaránt szerepelnek az újságban a távközlőhálózatok, berendezések, ezen belül jelzésrendszerek, átviteli módok, az ehhez szükséges új alkatrészek, kapcsolástechnikai megoldások, méretezési módszerek és telepítési kérdések. A mobil rendszerek és a rádiózás kapcsán a hullámterjedés, az elméleti villamosságtani problémák is érdeklődésre tarthatnak számot. Ezen túlmenően a híradástechnikával kapcsolatos gazdasági megfontolások, számítási módszerek is helyet kapnak, de szeretnénk a távközlés-politika újdonságairól is tájékoztatást adni, valamint az ezzel kapcsolatos szociológiai és oktatási problémák is szerepelnek a profilban.

• **Terjedelem:** A szakmai cikkek az újságban általában 3-6 oldal terjedelemben jelennek meg. Ennél rövidebbek inkább csak a hírek vagy beszámolók lehetnek. 6 oldalnál hosszabban pedig csak olyan alapvető újdonságok írhatók le, ahol a megértéshez az elméleti alapok és a gyakorlati megvalósítás egyaránt szükséges. Ez azt jelenti, hogy ábrák nélkül 12-20 ezer karakter lehet egy cikk szövege. Nyomtatott oldalanként kb. 1-3 ábra elhelyezése teszi az olvasó számára áttekinthetővé, vonzóvá az ismertetést.

• **Forma:** Sem betűtípus, sem rajzkivitel nem köti a szerzőket. A szövegeket *word formátumban* kérjük elkészíteni. Az újság egységessége kedvéért ugyanis az elektronikusan érkező szövegeket a nyomdának az újságban használt betűtípusú változatban küldjük tovább. Az ábrák megrajzolásánál is egyetlen kötöttség, hogy az újság *fekete-fehér kivitelben* jelenik meg, tehát a színes ábrák is fekete-szürke-fehér képként láthatók az újságban. Ennek megfelelően kérjük a szerzőket, hogy lényeges dolgokra ne hivatkozzanak úgy, hogy a piros vonal, vagy a kék alapterületű rész, ehelyett szaggatott, pontozott, vastag és vékony vonalak legyenek megkülönböztethetők, a területnél sraffozással lehet különbséget tenni.

• **Lektorálás:** A cikkek különböző minősítési folyamatoknál értékes pontokat jelenthetnek. Növeli a cikk értékét, ha azt lektorálják. A szerző kérésére bármikor lektoráltathatjuk a cikket, ez esetben a cím alatt *Reviewed* felirat utal arra, hogy nemcsak a szerkesztőség, hanem más is ellenőrizte a munkát, ami további pontokat jelenthet. Minden fél évben az első 5 számból kiválogatjuk azokat a cikkeket, melyek külföldi, nem magyar anyanyelvű olvasóink számára is érdekesek lehetnek. Ezeket angolra fordítva a 6. és 12. számban jelentetjük meg. Ez idegen nyelvű publikációnak számít.

• **Hivatkozások:** A cikk végén kérjük a kapcsolatos, vagy előzményként felhasznált cikkeket megadni. A hivatkozásokat számozzuk, a szám után következik a szerző, majd a cikk vagy a könyv címe, a megjelenés helye és időpontja. A szöveg közben szögletes zárójelben helyezzük el a hivatkozásoknál megadott sorszámot.

• **Megjelenés:** Az újság minden hónap 22. és 28. között jelenik meg. A pontos időpont függ az ünnepektől és a hétvégék helyzetétől. Mindig az előző hónap utolsó napjáig beérkezett cikkeket vesszük számításba. Tematikus megfontolásokból előfordulhat, hogy későbbi számban előnyösebbnek látszik a témakör tárgyalása. Általában a beküldést követő negyedévben helyet kap a munka az újságban. Késes esetén az átnézés vagy lektorálás után a beküldéstől számított két héten belül a szerző visszaigazolást kaphat a cikk elfogadásáról.

• **Szerzői adatok:** Annak érdekében, hogy az olvasók problémáikkal, véleményükkel közvetlenül kapcsolatba léphessenek a szerzőkkel, a cikk előtt lévő szürke részben, a cím alatt, szerepel a szerzők neve, munkahelyük és e-mail címük. Célszerű tehát, hogy ha a cikket úgy küldik be, hogy rajta van a név, a beosztás (egyetemi tanár, doktorandusz, osztályvezető stb.), a munkahely (olyan részletességgel, hogy a munkahely telefonszámáról már tudják kapcsolni a szerzőt) és az e-mail cím. Ez utóbbi a leglényegesebb az esetleges kérdések tisztázásához.

• **A beküldés módja:**

A cikkek eljuttathatók a főszerkesztőhöz:
Zombory László (BME, laszlo.zombory@mht.mbe.hu),
vagy a szerkesztőbizottság elnökéhez,
Lajtha György (lajtha.gyorgy@ln.matav.hu),
vagy a HTE titkárságának (hte@mtesz.hu).

A cikkeket elektronikus formában kérjük, tehát e-mailen, vagy lemezen.

Reméljük, hogy ezen ismeretek segítik kollégáinkat, hogy gondolataikat, új eredményeiket, műszaki megoldásaikat, számítási módszereiket közkinccsé tegyék. Várjuk tehát a cikkeket oktatási intézményekből, fejlesztőhelyekről, gyártóktól, üzemeltetőktől, tanulóktól, szakértőktől, oktatóktól és mindenkitől, akinek mondanivalója van a közösség számára.

A Szerkesztőbizottság

Az első hazai magfizikai gyorsítóberendezés újrafelállítása

KOSTKA PÁL

Részecske és Magfizikai Kutatóintézet
kostka@rmki.kfki.hu

Az Eötvös Lóránd Tudományegyetem lágymányosi épülete északi tömbjének földszintjén ismét felállították azt a Van de Graaff generátort, amely több mint 50 évvel ezelőtt az első magyarországi magfizikai gyorsítóberendezés volt.

Eredetileg ezt a magfizikai gyorsítóberendezést a soproni egyetem egyik laboratóriumában Simonyi Károly professzor és munkatársai építették. Ez azért jelentős magyar tudománytörténeti emlék, mert segítségével 1951. december 22-én Magyarországon először végeztek mesterségesen gyorsított elemi részecskékkel atommagátalakítást.

A generátorral előállított 440 keV-os protonokkal lítium fémet bombáztak, amikor is a Li (p,γ) Be magreakció révén berillium magok keletkeztek. A berillium magok erős γ sugárzás fellépése után a részecskékké bomlanak. Mivel ez a magreakció rezonanciaszerűen, csak meghatározott energiájú protonokkal való bombázáskor jön létre, a gyorsítófeszültség lassú növelésekor 440 kV-nál hirtelen jelenik meg a γ sugárzás. Így a γ sugárzás fellépte könnyen detektálható.

A fenti eseményt azért tekinthetjük jelentős hazai tudománytörténeti emléknek, mert a mesterségesen gyorsított részecskékkel történő kutatások magyarországi kezdőpontját jelentette. A berendezés sikeres üzembehelyezésével Simonyi Károly szempontjából lezárult egy több, mint tízéves periódus. A fiatal tudós már az 1940-es évek elején is foglalkozott gyorsítóberendezés létrehozásával a Bay Zoltán vezette Műegyetemi Atomfizikai Tanszéken. Ennek üzembehelyezését akkor a II. világháború eseményei meghiúsították. A háború alatt fejlesztett gyorsító eléggé előrehaladott állapotban volt, de sajnos a készüléknek csak egyes darabjai élték túl a háború viszontagságait. Mint a Soproni Egyetem tanára Simonyi professzor töretlen lendülettel és ugyanazon elméleti alapokon folytatta a gyorsító fejlesztését. 1951 végén elkészült a berendezés, mellyel ugyanazt a célt akarta elérni, mint korábban, mesterségesen gyorsított részecskékkel akart magfizikai kísérleteket végezni.

A berendezés az 1949-51 években épült. A Műegyetem soproni Bánya-, Kohó- és Erdőmérnöki Kara lehetőségeihez képest támogatta az Elektrotechnikai Tanszéken folyó munkát és annak vezetőjét, Simonyi professzort. A készülék eredetileg a most felállított műemlék készüléknél körülbelül 30%-kal alacsonyabb volt. A soproni laboratóriumok méretei ugyanis nem engedték meg, hogy az elvileg elképzelt magasságban realizálják a gyorsítót. Körülbelül 700 kV feszültséget tudott maximum szolgáltatni. Természetesen nemcsak magát a feszültségforrást kellett létrehozni, hanem ionforrást a maga tápegységeivel, gyorsítócsővet és vákuumrendszert is, valamint a gyorsító körül szükséges segédberendezéseket.

Az ionforrás – a szokásostól eltérően – földpotenciálon volt elhelyezve, így közvetlenül, manuálisan volt kezelhető és ellenőrizhető. Ebből következik, hogy a target volt nagyfeszültségen. Így a jelenség észlelőjének kellett a többszáz kV-on levő nagyfeszültségű elektród belsejében elhelyezkedni és figyelni a sugárzás detektort. A nevezetes esemény alkalmával Simonyi professzor vállalta azt a rendkívül kockázatos feladatot, hogy a 400-500 kV-on levő elektród belsejében dolgozzon és figyelje az eseményeket. Az eredményről és a gyorsítóépítésben szerzett tapasztalatokról az 1952. évi, II. Fizikus Vándorgyűlésen számoltak be.

1952-ben Simonyi Károlyt kinevezték a KFKI Atomfizikai Osztályának vezetőjévé. A további munka és a jobb lehetőségek kihasználása érdekében a készüléket Sopronból Csillebércre telepítették. Mivel a KFKI-ban iongyorsítónak egy másik készülék épült, Simonyi ezt a készüléket elektrongyorsítónak szánta, maximális feszültségét pedig 1000 kV-ban határozta meg. A készülék megkapta jelenlegi magasságát, a nagyfeszültségű elektródba elektronforrás és annak tápegységei kerültek, kiépült azok energiaellátása, mechanikus távvezérlése, földpotenciálon pedig egy eltéritő mágnes hozta vizszintesbe az elektronnyalábot. A KFKI csillebérci laboratóriumában megfelelő hely állt rendelkezésre a kiterjedtebb targetszerelvények és kísérleti eszközök elhelyezésére. A mágnes a részecskeenergia meghatározására is szolgált.

A készülék 1962. végéig volt üzemben, segítségével számos kísérlet folyt. A teljesség igénye nélkül megemlítünk néhány jelentősebb mérést: nagyenergiájú röntgensugárzás abszorpciójának vizsgálata, elektronok kisszögű szórásának mérése fóliákon és gáztargetek esetében. A berendezéssel „alkalmazott” kutatásokat is végeztek. Ezek közül érdemes megemlíteni az

olajok tulajdonságainak sugárzás okozta változását, vagy a biológiai kísérleteket, melyek közül különösen érdekes volt az élelmiszereken élősködő bizonyos rovarokat elpusztítani képes sugárdózis meghatározása.

A gyorsító 10 éves működése során számos fiatal fizikusnak volt lehetősége kísérlete elvégzésére. Felnőtt egy olyan generáció, amely később a korszerűbb eszközöket már szakszerűbben tudta munkájában alkalmazni. Ismereteik és új eredményeik a nemzetközi kapcsolatok kiépülését is segítette. Számos fizikus tudott később rövidebb-hosszabb időt a leghíresebb külföldi kutatóintézetekben eltölteni, és tehetett szert nemzetközi elismertségre. A mesternek azonban politikai okok miatt idő előtt meg kellett szakítania kapcsolatát az Intézettel.

1962-ben, a KFKI új gyorsítóépületének elkészültekor, a készüléket le kellett bontani, így az előbb a KFKI, később a Budapesti Történeti Múzeum, majd végül az Országos Műszaki Múzeum raktárába került. Ezalatt egy alkalommal, 1973-ban a „Pest, Buda, Óbuda egyesítésének 100-ik évfordulójára” rendezett kiállításon, más technikai emlékekkel együtt megtekinthették az érdeklődök. A készülék sorsa továbbra is a raktározás maradt, míg végül az ELTE Természettudományi Kara és azon belül különösen Kiss Ádám professzor és az Atomfizikai Tanszék vette oltalmába a gyorsítót.

Amikor ezt az Egyetem elhatározta, az volt a célja, hogy a TTK új lágymányosi épületében egyszer, mint a magyar tudománytörténet jelentős emlékét, felállítsák. Mielőtt ez bekövetkezett, a berendezés mégegyszer a nyilvánosság elé került. Mint jelentős tudománytörténeti emléket 2001-ben kiállították az „Álmok álmodói – Világraszóló magyarok” kiállításon. A nagy érdeklődés indokolta, hogy mielőtt a közönség megtekinti alaposan restaurálják a berendezést. Ekkor a körülbelül 40 éves raktározás eléggé jelentős nyomait ki lehetett javítani. Ugyanakkor az esztétikai benyomást is élvezetesebbé kívánták tenni. Ezt a munkát a KFKI Részecske és Magfizikai Kutató Intézet Magfizikai Főosztály munkatársai végezték.

Végül 2004-ben a készüléket az ELTE TTK épületében, az elegendően magas és tetszetős galériában állították fel. Ez látható az első képen, mely felvétel Kajcsos Zsolt művészi munkája. Ez a gyorsítónak méltó helye, hiszen abban az épületben van, ahol azokat az ismereteket oktatják, és azokat a tudományágakat művelik, melyek érdekében Simonyi professzor úr többek közt ezt a készüléket is létrehozta. Így ez a készülék ezen a helyen szimbolikusan mintegy összekapcsolja Simonyi munkásságát azzal a tudományos tevékenységgel, amelyet ebben az épületben most is művelnek.

A műemlék felavatásán az Egyetem vezetőin kívül jelen voltak azok a kutatók, mérnökök, szakemberek, akik a berendezés üzembehelyezésekor és az első kísérletek elvégzése során együtt dolgoztak. Jó volt találkozni a régi kollégákkal és visszaemlékezni a 40 év távlatából rendkívül szépnek tűnő időszakra.

Hogy látja egy szociológus?
Beszélgetés Pintér Róberttel, a BME-ITTK kutató-tanárával

NAGY BEATRIX HAVASKA
nbh@mailbox.hu

1. A történelemben volt már rabszolga-, feudális-, ipari-, fogyasztói-, stb. társadalom és nem olvastunk sehol arról, hogy e a társadalmak kialakulásakor a tudósok igyekeztek volna ezeknek a lényegét, működését meghatározni. Az igényektől és az emberi szokásoktól függően ezek többé-kevésbé működtek, majd valamilyen műszaki, vagy társadalmi változás hatására átmentek a következő társadalmi formába. Miért van szükség arra, hogy az információs társadalom kérdésével több egyetem, kutatóintézet és társadalmi szervezet foglalkozzon, mi a végcélja ezeknek a kutatásoknak?

Ha röviden akarnék válaszolni, akkor azt mondanám, hogy foglalkoztak ezzel a kérdéssel, vagy másképpen mondva, csak ezzel foglalkoztak a tudósok a saját korszakukban (például a rabszolgatartó társadalmak idején élt több nagy mester, teszem azt Arisztotelész). Igenis foglalkoztak azzal, hogy a társadalmaknak hogyan kellene működniük.

De egy nagy ugrással, most beszéljünk inkább arról a korszakról, ami tulajdonképpen elődje az információs társadalomnak. Az ipari korszakhoz képest itt már nincs akkora változás, mint az előtte levő feudális vagy az azt megelőző korszakokhoz képest. Az ipari fejlődés korszakában alakult ki az ideáktól, eszméktől független tudomány. Ennek hívei foglalkoztak azzal, hogy ennek a korszaknak hogyan kellene alakulnia, kiknek kellene irányítania a folyamatokat, mik lennének a főbb csomópontok. Saint-Simon-nak vagy Comte-nak a nevét említeném meg, akik nagy hatással voltak a kor társadalmára.

És végül, most is foglalkoznak az átalakulás természetével a tudósok. A wilsoni tengelyek mentén csoportosítanám ezeket a szereplőket. Négy szereplőnek a kapcsolatrendszerén, igyekezetén múlik ugyanis, hogy merre halad az a világ, ahol élünk, hogyan fejlesztjük az információs társadalmat.

Az első az állam vagy kormányzat, a politikusok; a második a gazdasági szereplők; a harmadik a civil szféra, civil társadalom (de a médiát is ide szoktam sorolni); a negyedik pedig a kutatás, az akadémiai szektor. A négy pontban emblematikus figurákat lehet találni, akik néha egy kicsit elvakultan, ügybuzgó módon, de viszik előre ezt az átalakulást. A kérdésekkel tehát a tudósok, a politikusok, a civil társadalom harcosai, különböző mozgalmak, és a gazdasági élet szereplői foglalkoznak.

Kicsit satnyább vagy szürkébb lenne a világ, ha a tudósok nem adnának ehhez a folyamathoz, az értelmezésekhez muníciót. Társadalmi diskurzusokat határoznak meg, jó fogalmakat, adatokat adnak és azokat értelmezik. Mindezek nélkül nincs fejlődés.

Nagy segítség, hogy a tudomány irányvonalakat ad, értelmezi a folyamatokat, trendeket mutat meg, majd ezeket összerakja. Eredményeiket használhatják a politikusok véleményalkotásra, és a különböző beavatkozások következményeit is megjósolhatják más országok tapasztalatai alapján .

A kutatásnak alapvetően két célja lehet: az egyik a világot megérteni, a másik megváltoztatni.

Marxnál ez a kettő úgy kapcsolódott össze, hogy a világot nem megérteni kell, hanem megváltoztatni. Ez alapvetően nem egy szenvtelen vagy értékmentes tudományfelfogás, aminek csak eszköze a megértés, és az a célja, hogy megváltoztassuk a világot. Ez egy forradalmi hevület, ami szerint a változások szükségesek és meghatározhatók. És – folytatva a gondolatmenetet – nyilván a tudósok tudják, mi a helyzet, igyekezzenek ezen jobbítani ők maguk. Ezek szerint például a tudósoknak kellene a parlamentben ülni, és döntéseket hozni, nem pedig a politikusoknak.

Ezzel szemben egy másik álláspont az mondja, hogy nem megváltoztatni kell a tudósoknak a világot, hanem megérteni. Vagyis eszközt adni azoknak, akik felelősséggel bírnak, és merik vállalni számolva a veszéllyel, hogy rosszabb is lehet. Ezek a politikusok és a gazdasági élet szereplői. Ha én, a tudós megértem a világot és azt leírom, akkor ennek a segítségével a társadalom egy teljesebb képet kaphat. De persze a tanácsokat nem feltétlenül kell megfogadni.

2. Az emberek nagy többsége a társadalmat meghatározó folyamatokat csak lassan, fokozatosan fogadja be. Van egy időszak, amikor még tiltakoznak a gépek, vagy a mobilitás elfogadása ellen. Azután lassan természetükké válik a számítógép, az internet és az informatika használata, ennek különböző megoldásait a fiatalok már elfogadták, ebbe nőttek bele. Szükségesnek tartja-e, hogy a 60 éven felüliek meggyőzése érdekében a társadalom profétái sokat dolgozzanak?

Ha cinikus akarnék lenni, akkor röviden azt válaszolnám, hogy nem. Nemrégiben volt nálunk egy előadás, ahol felvetődött, hogy tulajdonképpen ez a változás

nem a 60 fölötti korosztálynak szól. Tehát nem az ő lelküket kell megváltanunk, nem ők a fő célcsoport. Ugyanakkor személyes meggyőződésem, hogy a társadalmi programokon túl létezik egy „kispályás világmegváltás" is, vagyis mindenkinek a saját felelőssége, hogy a környezetében lévő emberekkel mit kezd. Nem, mint tudós, politikus stb., hanem mint családtag. Ez egy fordított szocializációs kérdés, ahol a gyermek felelőssége a szülő felkarolása, tanítása – ezért vettem a szüleimnek számítógépet.

De még komolyabbra fordítva a szót, azt mondanám, hogy ennek a korosztálynak is kell egy bizonyos sajátos célrendszeren belül esélyegyenlőséget biztosítani, például olyan szolgáltatásokkal, amely ezt a korcsoportot érinti. Tehát ezzel a társadalmi csoporttal is kell foglalkozni, de látni való, hogy nem ők jelentik a sodrás fő irányát.

Ennek okát a technológiák szétterjedésének sajátos folyamatában kell keresnünk. A rogersi innovációterjedési modell öt csoportra bontja a társadalmat: az innovátorok, a korai adaptálók a korai követők, a kései többség és a lemaradók-ellenzők. Jellemzően – bár az elmélet ezt így nem mondja ki – de az információtechnológia kapcsán Magyarországon jól látni, hogy elsősorban nem az idősek közül kerülnek ki azok, akik a korai csoportban vannak. Az idősek a harmadik vagy sokkal inkább a negyedik csoportból kerülnek ki, vagy egyszerűen ellenzik a technikai újdonságokat. Ehhez persze az is hozzátartozik, hogy milyenek ezek az eszközök.

Megnézném például azt a mai fiatalt, aki használja ezeket, hogy néhány évtized múlva ugyanezeket az eszközöket hogyan tudná üzemeltetni amikor már a nyomógombot sem tudja az ujjával eltalálni, vagy nem látja a mobiltelefon kis képernyőjét. A mai világ nem túlzottan segítőkész, hogy ezeket az új infokommunikációs eszközöket egy ilyen idősebb célközönségnek hoznák létre. Ami egyébként nem csak itt jellemző, hanem általános trend, például, ha a reklámokat nézzük, azok is a 18-50 éves fogyasztóképes korosztálynak próbálnak mindent eladni.

Tehát végeredményben mindenki megküzdhet a saját lelkiismeretével, hogy az idősebb korosztállyal foglalkozik-e, mert a politikai programok és a világ általában nem őket állítja a középpontba.

3. Ha tökéletes lenne az információs társadalom szociális, filozófiai megalapozása, akkor milyen különbség jelentkezne ahhoz képest, ha ezzel a kérdéssel senki nem foglalkozna?

Ez egy nagyon jó, de egyúttal egy erősen teoretikus kérdés. Három állapot képzelhető el ezek szerint:

– Senki nem foglalkozik az információs társadalom megalapozásával és a dolog mégis működik – ez az állapot szerintem csak elméletileg létezhet, nincs rá igazi példa.

– Ennek teljes ellentéte a tökételes megalapozás, amikor a releváns kérdésekkel és a válaszokkal mindenki tisztában is van. Szerintem ez az állapot is csak elméletileg létezhet.

– Végül kimondatlanul benne van a kérdésben az is, hogy bár a világon (Magyarországon) az információs társadalom problémáival foglalkoznak, de azért korántsem tökéletes ez a megalapozás.

Ezt a három állapotot úgy képzelem el, mint egy szakaszt, aminek egyik vége, ahol még senki sem foglalkozik a kérdéssel, másik vége, ahol mindenki ismeri a megfelelő kérdésekre a választ, és a két szélső állapot között számtalan közbeeső pont létezik, amikre egyszerre jellemző, hogy ismerik is, meg nem is a kérdéseket és a válaszokat.

Azt gondolom, hogyha tökéletes lenne az információs társadalom elméleti és társadalmi megalapozása, akkor egy picit tudatosabb, jobb lenne az a fajta fejlesztői munka, amit a fentebb említett négyes szereplő rendszerben (állam, gazdaság, civil társadalom és akadémia) folytatnak.

Persze, ha senki sem foglalkozna ennek a megalapozásával, akkor is úgymond „csinálnánk". Akkor is lenne mögötte valami feltételezés, hogy jó-e ez az egész vagy nem. Lennének körülötte elgondolások, még ha tudományosan nem is alapoznák azt meg. Az is megalapozás, hogy ha valaki azon gondolkodik, hogy bevezessem-e a piacra ezt vagy azt a terméket vagy sem. Véleményem azonban az, hogy jobb lehetne az információs társadalom fejlesztésének az egész kontextusa, ha az elméleti és szociális megalapozása is jobb lenne.

4. A bevezető kérdések után térjünk át a szakmai tevékenységre. Hallottam arról, hogy az ITTK Klub csütörtök esti összejövetelein mindig kiváló előadók vannak és élénk vita alakul ki. Hogyan választja ki a témákat és hogyan éri el, hogy a résztvevők mindig kiemelkedő aktivitással kapcsolódjanak be a beszélgetésbe?

Eddig 32 Szakmai Klub volt az ITTK történetében – bár mindegyiken ott voltam, mint házigazda és csak elfogult véleményt tudok mondani –, még így sem mondanám, hogy minden előadás kiváló vagy izgalmas lett volna. Anélkül, hogy bárkit megbántanék, volt olyan beszélgetés, ami nem sikerült igazán érdekesre, de ez természetes.

Maga a rendezvénysorozat eléggé sokszínű volt a négy év alatt. Előadónk volt rögvest kinevezése után az informatikai miniszter, az adatvédelmi biztos, beszélhettünk az ország egyetlen informatikai önkormányzati tanácsnokával, de foglalkoztunk klasszikus tudományos kérdésekkel is, tartottunk pódiumbeszélgetést, és még kihelyezett konferencia tagozati ülést is. Ez a sokszínűség azonban végeredményben esetleges volt, és nem szándékoltan sikerült ilyen kacskaringós utat bejárni.

Mindig próbálunk olyan témát hozni, ami egyrészt közérdekű, másrészt, amihez van előadó is, olyan témát, amit nagyon fontos megvitatni. Ilyen volt például a

legutóbbi alkalom is Sükösd Miklóssal, amely a média és az ökológiai válság kapcsolatával foglalkozott. Ez úgy gondolom társadalmi szempontból nagyon fontos diskurzus.

Éppen a napokban jutottunk kollégáimmal arra a meggyőződésre, hogy – mivel minden vitán előjönnek ugyanazok az érvkészletek és mítoszok – szeptembertől egy új sorozatot indítanánk, amelynek ezek lennének a visszatérő általános elemei, ugyanakkor ezzel párhuzamosan érvényesülne az eddigi kiválasztási politika is, annak érdekében, hogy az aktuális kérdések kerüljenek napirendre.

5. Az elmúlt időszak információs társadalmának fejlődéséről sok statisztikát készítettek és ebből következtetéseket vontak le. Általában az a tapasztalat, hogy következtetéseikhez képest a változások kedvezőbbek, gyorsabbak, vagy tán inkább nehezebben tudják az emberek elfogadni a felkínált lehetőségeket?

A rendszerváltás után rendeztek egy konferenciát a szociológusok, aminek az volt a témája, hogy: „miért nem láttuk, hogy jön”. Itt a beszélgetés témája az volt, hogy miként lehet az, hogy ha mi jó társadalomtudósok vagyunk, itt éltünk ebben a társadalomban, és mégsem gondoltuk a 80-as évek végén, hogy rendszerváltás lesz, és ekkora átalakulás zajlik le ilyen rövid idő alatt. Ennek a kudarcnak a kapcsán felmerül az általános kérdés, mi a feltétele az események „bejóslásának”, úgymond a „jövőbe látásnak”. Erről a hallgatóimmal is szoktam beszélgetni, akkor általában az a meggyőződés alakul ki bennünk, hogy azokat az eseményeket lehet jól jósolni, melyeknek van valami lendülete, dinamikája, és többé-kevésbé lineárisak, nincs bennük jelentős törés.

Ilyennek tűnik egyelőre például az internet fejlődése Magyarországon: szép lassan, egyenletesen növekszik. Könnyű ezt az adatsort meghosszabbítom, és azt feltételezem, 2005-ben 30% lesz a penetráció. De ez a lusta, egyenletes fejlődés sem mindig bizonyul valósnak. Még nehezebb a helyzet, ha „törés” van a fejlődési vonalban, mert akkor felborul a „papírforma”. Erre jó példát mutatnak a „dot.com” krízis előtt készült statisztikák, ha összevetjük a válság után készítettekkel.

A krízis 2000 márciusban volt, amikor félig-meddig összeomlott ez a terület és hihetetlen tempóban vonták ki a tőkét a szektorból. Összehasonlítva az adatokat, ugyanazok az emberek, ugyanazokon a konferenciákon egyik évben még azt mondták, hogy határ a csillagos ég, a következő évben pedig azt, hogy egy jelentős törésnek, egy érthető visszarendeződésnek vagyunk a tanúi. Teljesen megváltozott tehát a diskurzus.

Tulajdonképpen az a baja a témával foglalkozó legtöbb tudománynak, hogy csak a jóslásokra szorítkozhat, amikor a jövőről tesz kijelentéseket. Tehát a lineáris folyamatokat jól látja, de a töréseket nem. És ezek a törések időről-időre megjelennek, létre hozve rendszer-állapot változást és átstrukturálódást.

A 2000-ben napvilágot látott Technológiai Előretekintési Programban (TEP) van egy hivatkozás egy nyolcvanas évek végi, kilencvenes évek eleji becslésre, ami arra vonatkozott, hogy előreláthatólag mennyi mobiltelefon használó lesz majd Magyarországon 2000-ben. Optimista forgatókönyv azzal számolt, hogy a lakosság 2,5%-a fog mobiltelefont használni. Ezzel szemben 20-25% volt a mobilt használók száma. Ebben a „bizniszben” abszolút nem lehet jósolni, de igazán „visszafelé” sem, pontosan tudjuk, hogy mi miért és hogyan is történt.

6. Hogyan képzeli, mi lesz az ITTK fejlődésének menete, milyen kutatásokkal fognak foglalkozni 5, 10, vagy 15 év múlva?

Forgatókönyvek vannak arra vonatkozóan, hogy hogyan fejlődjön tovább az intézet. Van akadémiai szereplésről szóló forgatókönyv, ami azt jelenti, hogy ez a központ egyre inkább egy akadémiai intézmény lesz, és egyre több kutatással, kutatóval az Európai Uniós kutatási rendszerekbe integrálódik. Kevés piaci jellegű feladatot lát el, a finanszírozása is megoldódik, és végeredményben egy klasszikus tudományos intézménynyé formálódunk.

Ebben az esetben is a többször említett négyes szereplői rendszerben lehet gondolkodni: mi jelenleg a kutatási szférában vagyunk, lehetnek megrendelőink állami oldalról gazdasági oldalról, vagy kooperálhatunk a civil szervezetekkel is. Lehetnek regionális, országos, és nemzetközi kapcsolataink. Stratégiai szempontból nagyon fontos az állammal való együttműködés. Az ITTK munkatársai a magyar információs társadalom stratégiák elkészítésében az elmúlt években minden esetben aktívan részt vettek.

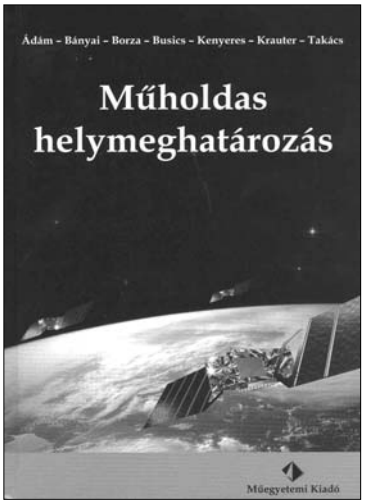
Azt is elképzelhetőnek tartom viszont, hogy bizonyos szempontból inkább a gazdaság felé lépünk, például a piackutatás felé. De még az is lehet, hogy ezek a trendek egyszerre hatnak és az Európai Unió felöl is kapunk bizonyos impulzusokat, de az államtól, gazdaságtól és civil szférától is, végül egy európai uniós kutatói hálóba integrálódunk. Utóbbi azonban inkább egy 10-15 éves forgatókönyv része. Mivel egyre inkább információs társadalomban élünk, ezért hosszú távon nincs értelme egy elkülönült információs társadalmat kutató intézetről beszélni. Akkor már mindenki, aki a jelen folyamatait kutatja, ezekkel a témákkal foglalkozik, így az intézet szakosodása várható a részterületek kutatására.

Könyveket ajánlunk

Műholdas helymeghatározás

Ádám–Bányai–Borza–Busics–Kenyeres–Krauter–Takács
Műegyetemi Kiadó, 2004

A műholdas helymeghatározás már hétköznapjaink részének tekinthető. Számos új technikát, mérési elvet és nagyon összetett teljes rendszert tartalmaz, tehát megérteni, helyesen alkalmazni nem egyszerű dolog. Ebben segít a Műegyetemi Kiadó új egyetemi tankönyve. A 458 oldalas mű elsősorban a geodézia, navigáció és térinformatika területén működő szakembereknek és ezeken a szakokon tanuló hallgatóknak szól. Szélesebb körben is figyelemre méltó az agrár, műszaki és természettudományos szakemberek számára, de felsőfokú szintű, ugyanakkor közérthető stílusa miatt ajánlható szélesebb felhasználói körnek is.



A könyv a műholdas helymeghatározás elméleti és gyakorlati alapjai mellett az alkalmazások, hazai fejlesztések, kapcsolódó kutatási irányzatok területén is eligazítást nyújt.

Az 1. fejezet a globális helymeghatározás történeti áttekintése. A 2. fejezetben az egyes helymeghatározó rendszerek rövid leírása szerepel, majd a GPS műholdak rádiójeleivel foglalkozik. A 4. fejezet a műholdas helymeghatározás elméleti alapjait ismerteti. A következő fejezetek tárgya a mérési módszerek bemutatása és a mérési adatok feldolgozási módszerei. A 7. fejezet a Nemzetközi GPS Szolgálat tevékenységét és szolgáltatásait mutatja be.

Ezek után a geodinamikai és geodéziai alkalmazásokat tekinti át, majd egy külön fejezet a földtani, térinformatikai, fotogrammetriai, építőmérnöki, bányamérnöki, hidrológiai, környezetvédelmi, aeronómiai, meteorológiai, mezőgazdasági, erdészeti, katonai, szabadidős és sportcélú alkalmazások sorát mutatja be. A 10. fejezet a navigációé, az utolsó pedig a fejlődés további irányait tekinti át.

Sajnálatos, hogy kimaradtak a távközlési alkalmazások: például az, hogy a GPS rendszer mindenütt rendelkezésre álló szuperpontos időalapja alkalmas hálózatok szinkronizálására is.

A könyv belső címlapján szerepel egy honlap is, ahol kiegészítő információk, újabb eredmények és a vitafórum aktualitás hozzászólásai találhatók meg. (www.geod.bme.hu/gnss).

Tudományos évfordulóink 2004.

Szerkesztette és kiadta:
Nagy Ferenc

Az évfordulók jegyzékére utaló cím nyomán azt hihetné az olvasó, hogy egy unalmas felsorolást tart a kezében, melyben valamilyen szabály szerint összerendezve évszámokat, dátumokat és eseményeket talál. Egy avatott szerkesztő, a tudománytörténetet elmélyülten ismerő szakember azonban egy ilyen listából is élvezetes olvasmányt készíthet. A 100 oldalas könyv számos szempontból lehetővé teszi a keresést és segít az aktuális évfordulók megünneplésében.

Az első rész 2004 minden napjára megadja, hogy akkor milyen jelentős évfordulóra emlékezhetünk. Halhatatlanok születési és halálozási dátumai rendbe szedve, lehetőleg azoké, akiknél ez 50 vagy 100 éves kerek számú évforduló, bár az utolsó két évszázadban a 25, 75 és 125 éves évfordulókat is lejegyezte a szerző. Így ha bárki előadását valamilyen visszaemlékezéssel akarja kezdeni, akkor a napi aktualitásokat megtalálja ebben a fejezetben. A személyhez kötött eseményeken túlmenően nagy felfedezések bejelentése, épületek, intézmények felavatása, vagy jeles iratok megjelenése is szerepel a felsorolásban.

A következő rész a Krónika, az évfordulókhoz kapcsolódva 25 évenkénti felbontásban, 500 évre visszamenőleg felsorolja azokat az eseményeket, melyek kiemelkedő jelentőségűek, így például egy-egy államfő, tudós vagy feltaláló életének évfordulóján tömören olvashatjuk eredményeit, betöltött funkcióit és elért sikereit. E munka érdekességét azzal jellemezhetjük, hogy többek között itt találjuk meg Boole algebra leírását tartalmazó első könyv megjelenésének 150. évfordulóját, ekkor született Gerbeaud, a híres cukrász család első tagja, ekkor indult a Vasárnapi Újság, az új képes hetilap és ekkor kezdték meg a Dohány utcai zsinagóga építését. Külön részben találkozhatunk a 100, 75, 50 és 25 éve Nobel-díjat kapott tudósok bemutatásával.

A könyv tartalmaz név- és eseménymutatót is, mindezek mellett pedig előadásrészletek, régi versfordítások és fényképek teszik élvezetessé a lapozgatását. Reméljük, hasonló kiadvány 2005-ben is megjelenik.

BASICS OF THE NEW GENERATION INTERNET
Keywords: addressing, directory, mobility, security, international projects
One of the corner stones of our information society is Internet. The Internet has been based on version 4 of the Internet Protocol. Its dramatic proliferation and the associated new requirements, however, required the extension of IPv4. The real solution is the development of a new version. This article presents Internet Protocol version 6 along with the overview of requirements leading to its development and the outline of the newly created services. Last but not least, the actual situation of IPv6 is addressed, both at international and domestic level.

TRANSITION TO THE USE OF THE NEW GENERATION INTERNET
Keywords: IPv6, transmission techniques, protocol converters
The period of time between the birth of the idea of the new generation Internet and its implementation was much shorter than the time required to the proliferation of the protocol. The most important task is the provision of seamless transition to the new version and the transparency for both IPv4 and IPv6 users. To this end, efficient transition technologies have to support the co-working of the two protocols. IPv6 has several favorable features which will facilitate the transition process. There are, however, certain factors which slow down the world-wide proliferation of the new version protocol.

THE ASN.1 LANGUAGE IN PROTOCOL DESIGN
Keywords: mobile data communications, coding techniques, formal description schema
The ASN.1 language is used for the description of messages between communicating applications and as such, has high-level message description forms which frees protocol designers from addressing communications messages at bit or byte level.
Originally it was used for the description of e-mail messages within the OSI protocol. Now ASN.1 has a much wider field of application, including network management, secure e-mail, mobile telecommunications, air traffic control and VoIP applications. The paper presents this language and its manifold implementation opportunities.

AUTOMATIC TEST GENERATION BASED ON FORMAL PROTOCOL SPECIFICATION
Keywords: conformance testing, mutation analysis, evolutionary and bacterial algorithms
This paper presents a technique for an automatic test generation based on the formal SDL specification of protocol. Protocol testing is an important phase of the development process but the creation of testing sets is a time-consuming task. The automation of this phase reduces implementation time and abolishes a serious source of error. We will show how mutation analysis can be used for matching state-pace algorithms with test criteria. This will be followed by the use of evolutionary algorithms for the selection of an optimum partial set out of an original testing set. Using these techniques a complete test generation process will be created which will produce testing sets from the formal specification of a protocol.

INTRODUCING GPRS DATA COMMUNICATIONS TECHNOLOGY AND GTP PROTOCOL
Keywords: GTP, 2.5G and 3G networks, Internet, data communications technologies
The dramatic proliferation of mobile communications has created the need for availability of Internet-based services on the run. More and more people would like to access their information and entertainment services, to read their e-mails or even their corporate intranet or other data networks. Up to the recent times these needs could have been served by the slow and not really cost-effective circuit-switched data communications technology.
This technology was not suited for accessing the emerging WAP and other web pages. This article presents the 2.5G and 3G associated GPRS technology as well as the GTP protocol which can be used with GPRS and Internet as well.

GENERAL PURPOSE SECURE ANONYMITY ARCHITECTURE
Keywords: anonymity, network architecture, secure communications
Among the requirements for electronic communications anonymity becomes more and more important (typically in applications such as electronic voting, polling, electronic payment). The current network layer architecture in itself does not support anonymity. In this paper a solution is proposed for the above problem. The outlined secure anonymity architecture includes new layers with specific functions for anonymity and determines their place in the current model.

PROCESS ALGEBRAIC DEVICES IN SECURITY ANALYSIS OF SENSOR NETWORKS
Keywords: security protocol testing, sensor network coding, CSP, exchange of keys
The considerable development in communications and networking technologies as well as the increasing level of miniaturization allow for the implementation of wireless sensor systems. Sensor computers form a self-organizing ad hoc network. The security of these networks is more difficult to guarantee than that of traditional systems. This article shows an example of how the proven process algebraic devices of traditional telecommunications networks can be used for checking the security features of these systems.

Summaries • of the papers published in this issue

Contents

Cover: The first Hungarian accelerator originated in Sopron, worked in KFKI and now rests in peace at Eötvös Loránd University

Szerkesztőség
HTE Budapest V., Kossuth L. tér 6-8.
Tel.: 353-1027, Fax: 353-0451, e-mail: hte@mtesz.hu

Hirdetési árak
1/1 (205x290 mm) 4C 120.000 Ft + áfa
Borító 3 (205x290mm) 4 C 180.000 Ft + áfa
Borító 4 (205x290mm) 4 C 240.000 Ft + áfa

Cikkek eljuttathatók az alábbi címre is
BME Szélessávú Hírközlő Rendszerek
Budapest XI., Goldmann Gy. tér 3.
Tel.: 463-1559, Fax: 463-3289,
e-mail: zombory@mht.bme.hu

Előfizetés
HTE Budapest V., Kossuth L. tér 6-8.
Tel.: 353-1027, Fax: 353-0451
e-mail: hte@mtesz.hu

2004-es előfizetési díjak
Hazai közületi előfizetők részére:
1 évre bruttó 31.200 Ft
Hazai egyéni előfizetők részére:
1 évre bruttó 7.000 Ft

Subscription rates for foreign subscribers:
12 issues 150 USD,
single copies 15 USD

www.hte.hu
Felelős kiadó: MÁTÉ MÁRIA
Lapmenedzser: Dankó András

HU ISSN 0018-2028

Layout: MATT DTP Bt. • Printed by: Regiszter Kft.