ÚVT MU
zpravodaj

Bulletin pro zájemce o výpočetní techniku na Masarykově univerzitě • červen 2005 • roč. XV • č. 5

EIS Magion na MU

Petr Vokřínek, ÚVT MU

Tématika Ekonomických informačních systémů (EIS) není na stránkách Zpravodaje ÚVT příliš frekventovaná. Při listování řadou předchozích ročníků jsem narazil pouze na sérii článků o ASŘ (Automatizované systémy řízení) z roku 1997, z nichž poslední je věnován počátkům aplikace EIS Magion [1]. V pozdějších ročnících se obecnějším pohledem na integraci EIS do celouniverzitního informačního systému zabýval např. článek [2] z června 2001.

Vzhledem k významu EIS pro chod univerzity je proto zřejmě na místě nabídnout čtenářům přehlednou stručnou informaci o dění v této oblasti od roku „nula“ (1997) do současnosti. Nejprve uvedeme několik vybraných průhledů do minulosti z různé perspektivy, dále se krátce zmíníme o lidech v centru EIS i v jeho okolí a v závěrečné části příspěvku uvedeme alespoň ve stručnosti nejbližší systémové okolí EIS Magion v podmínkách MU.

1 EIS na univerzitách v ČR a na MU

Na prvních místech mezi ekonomickými IS provozovanými na veřejných vysokých školách (VVŠ) ČR – z pohledu rozsahu a komplexnosti poskytovaných služeb – se v posledních deseti letech objevily zejména tyto tři:

- *aplikace EkonFis firmy Pragodata*. Aplikace byla ukončena – nikoliv z důvodů nespokojenosti uživatelů, ale proto, že se její nový vlastník IDS Scheer ČR orientuje na jiné aktivity. U většiny VVŠ byla aplikace nahrazena systémem SAP VVŠ – jedná se vesměs o moravské univerzity, konkrétně VUT, JAMU a MZLU v Brně, Univerzitu T. Bati ve Zlíně, Univerzitu Palackého v Olomouci a VŠB-TUO v Ostravě. Pro zajímavost – IDS Scheer je expertním partnerem SAPu;
- *aplikace FIS* (nyní ve verzi iFIS) *firmy BBM*. Tato aplikace se dále rozvíjí zejména v českých zemích. Písecká firma BBM se mezitím stala dceřinou společností firmy Logis;
- *aplikace EIS Magion firmy Magion*. Cesta vývoje a rozvoje aplikace je v podstatě zakázkového charakteru. Původně byla vyvíjena pro naši univerzitu a v úzké součinnosti s ní, později úspěšně rozšířena na několik dalších VVŠ – na ZČU v Plzni, ČZU v Praze, Univerzitu Hradec Králové, Ostravskou univerzitu a Slezskou univerzitu v Opavě.

Firma Magion Vsetín, se kterou Masarykova univerzita v oblasti ekonomických IS spolupracuje od roku 1997, byla založena v roce 1990 jako spol. s r.o. a zpočátku se soustředila právě na zakázkový vývoj informačních systémů. V roce 2001 byla transformována do právní formy akciové společnosti a v roce 2003 fúzí s firmou Sowa, s.r.o. významně posílila v oboru informačních

technologií. V současné době kromě zakázkové činnosti dodává komplexní ekonomické systémy obecných organizací a provádí jejich servis.

Za téměř 10 let spolupráce mezi MU a firmou Magion prodělala instalace EIS Magion v podmínkách MU vývoj jak po stránce organizačního uspořádání, tak po stránce rozsahu poskytované funkcionality i po stránce použitých technologií.

Aplikace EIS Magion byly na MU uvedeny do rutinního provozu k 1. lednu 1998, zpočátku na 7 fakultách a 3 dalších součástech MU. V roce 2005 má MU 18 hospodářských středisek (9 fakult a 9 dalších součástí) plně zapojených do EIS s rozhodující mírou jednotnosti metodiky i technologie zpracování ekonomických dat. V roce 1998 byly s různou mírou podrobnosti aplikovány uživatelské moduly *Manuální účetnictví, Podvojné účetnictví, Pohledávky, Závazky, Cestovní náhrady, Pokladny* a *Banka*, v dalších letech byly kromě prohloubení komplexnosti zpracování a integrace uvedených modulů aplikovány i moduly Majetek, Sklady, Objednávky a *Rozpočty*.

Po stránce použitých databázových a aplikačních technologií (mj. určujících formu aplikace z pohledu uživatelů) začínal EIS Magion v roce 1998 v tzv. *textové verzi* známé uživatelům z MU pod názvem „bombur“, což je název serveru provozovaného v ÚVT MU, na němž byla (a dosud jsou) uložena veškerá data EIS Magion a také všechny aplikace textové verze (data jsou uložena v databázi Informix a textová verze je naprogramována v jazyce 4GL, což je nativní jazyk databáze Informix). V pozdějších letech se firma Magion zaměřila na vývoj grafické verze svého systému v technologii PowerBuilder. Protože klíčový zákazník, MU v Brně, se nechtěl vzdát sobě na míru vyvinuté textové verze, udržovala a rozvíjela firma původní moduly systému jak v textové verzi (pro MU) tak v grafické verzi (pro ostatní zákaznické VVŠ); nové moduly (zejména modul Majetek) však již výlučně v grafické verzi. Touto cestou se na MU dostal tzv. *grafický klient*, známý uživatelům pod názvem „rumbur“ (resp. „cvibur“ pro testovací – cvičnou variantu), což je název terminálového serveru provozovaného v ÚVT MU (přesněji klastru terminálových serverů, viz článek J. Ocelky v tomto čísle Zpravodaje), na němž běží aplikace grafické verze, které jsou již (na rozdíl od textové verze) nezávislé na použité databázi. V loňském roce, v souvislosti se změnou zákona o DPH, se na MU skokem rozšířilo používání grafické verze, protože nutné úpravy související se změnou legislativy stihla firma Magion realizovat pouze v grafické verzi (ze zcela objektivních časových důvodů). Úplné ukončení provozu textové verze EIS Magion je plánováno na 31. květen letošního roku jako nezbytná podmínka pro to, aby o několik týdnů později, v červenci, mohla být ekonomická data přenesena ze stávajícího, výkonově již nedostačujícího serveru bombur s Informixem na nový, výkonnější datový klastr amber s databází Oracle. Analýza pro přenesení nepostradatelných funkcí je ukončena a grafická verze EIS Magion, resp. aplikace v Inetu MU jsou připraveny uživatelům plně nahradit původní textovou verzi.

2 EIS Magion a lidé kolem

Uživateli EIS Magion na MU jsou zejména zaměstnanci odborných útvarů MU, kteří zabezpečují hospodářskou správu či administrativu chodu univerzity prostřednictvím funkcí EIS a pracují s účetními doklady (účetní, správci majetku, evidenti, pokladní apod.). V současné době má EIS Magion na MU více než 300 takovýchto aktivních uživatelů (loginů) – současně jich zpravidla pracuje 50–100, ve špičkách i více.

Všichni tito uživatelé jsou „spřízněni Magionem“ a přitom rozděleni různými přístupy (právy) k modulům (komponentám) EIS. Tato práva mají dvojí charakter:

- jednak jsou vztažena k místu zpracování (pracoviště, více pracovišť, sekce, fakulta, celá MU apod.) a k roli či úloze uživatele (které doklady, části modulu, modul, více modulů apod. zpracovává);
- a dále mají různě nastavitelnou „sílu“ (zda příslušný doklad uživatel pouze čte nebo i vkládá, ruší, mění, předúčtuje či účtuje).

Potřebnou podporu uživatelům poskytují správci EIS, kteří:

- vytvářejí potřebnou infrastrukturu pro chod EIS po stránce metodické (EO RMU – správa

účetních číselníků, předkontací, analytik, dokladových řad apod.), technicko-provozní (ÚVT – administrativa a servis uživatelů, jejich rolí a přístupů, styk s autory) a autorské (Magion);
- pracují na základě subsidiarity: problém (metodický či technický) je zachycen správcem na uživatelské adrese nebo lince, identifikován a postupován zdola až k místu, kde může být vyřešen;
- vytvářejí metodickou a technickou dokumentaci pro uživatele EIS.

Ke kontaktu uživatelů se správci slouží zejména skupinová e-mailová adresa magion@ics.muni.cz, která je určena pro technicko-provozní správu EIS, a telefonní linky pro metodickou a technickou podporu provozu EIS, které jsou uvedeny na stránkách nápovědy v rumburu.

Dalšími formami a nástroji uživatelské podpory EIS Magion na MU jsou:

- informace ekonomického odboru rektorátu MU na webových stránkách http://morwen.rect.muni.cz/web/portaly/ekonomika/popis-ekonomika.htm (zejména důležité normy, interní pokyny, aktuality);
- školení uživatelů EIS. V poslední době můžeme jmenovat (a) základní instruktáž pro práci s grafickou verzí EIS – cca před rokem provedla firma Magion, (b) polodenní specializované semináře pro práci s pokladnami (hlavní i drobného vydání, začátečníci i pokročilí – celkem 4 běhy) – provedl EO RMU v součinnosti s ÚVT v dubnu 2005, (c) připravované školení k modulu Majetek, příp. další dle požadavku uživatelů;
- dokumentace ve formě nápovědy přímo na rumburu, kterou vyvíjí, spravuje a průběžně aktualizuje ÚVT. Týká se obecných informací a postupů při práci se systémem, postupů při zřizování uživatelů a jejich administrativy, aktuálních informací obecných, o nových funkcích EIS apod. Nápověda je otevřená pro sdělování uživatelských relevantních námětů, postřehů a zkušeností a rádi bychom zde publikovali i metodicko-uživatelské příručky a postupy pro práci s jednotlivými moduly EIS;
- skupinové e-mailové adresy (vedeko@ics.muni.cz, dekeko@ics.muni.cz, alleko@ics.muni.cz, majeko@ics.muni.cz, skleko@ics.muni.cz) pro hromadné rozesílání zpráv uživatelům EIS Magion v rámci MU;
- smlouva s dodavatelskou firmou Magion Vsetín o podpoře při provozu EIS zajišťující plný autorský servis. Pro letošní rok byla nově zformulována vzhledem k aktuálním potřebám MU a posílení vyváženosti smluvního vztahu.

Než uzavřeme tuto část, věnovanou lidem kolem EIS Magion, vzpomeňme ještě na pionýrská léta se sálovými počítači: když se technici či systémáci oddávali „hraní karet“, bylo to znamením jejich dobré práce – úlohy běžely celou směnu. Leč běda zákazníkům i programátorům, pokud byli v horečném pohybu. Doba se však mění: při pohledu na náš EIS není neutuchající pohyb kolem znamením, že věci nefungují – díky neustávajícímu přílivu nových technologií a postupů jsme se nikdy dlouho nenudili a troufám si říci, že už ani nikdy nebudeme.

Když se ohlédneme za dosud strávenými léty s Magionem, může každému vytanout na mysli něco jiného – od nejlepších vpravdě tvůrčích let při spoluvytváření systému k obrazu MU až po ojedinělé skřípění zubů při vstřebávání mechanismů systémem či jinými uživateli předkládaných. Ale společně snad můžeme konstatovat, že skutečně velký „průšvih“ s výpadkem systému (řešený na nejvyšší úrovni MU) nenastal, za což patří díky především našim partnerům z Magionu, ale také všem lidem kolem EIS na MU.

3 Systémové okolí EIS Magion v podmínkách MU

Na závěr tohoto článku se ještě alespoň výčtem a hrubou charakteristikou podívejme na nejbližší systémové okolí EIS Magion na MU.

1. *Personální a mzdový systém MU (PaM).* PaM systém MU byl vyvinut vlastními silami MU, dosud je provozován na již zmíněném bomburu (tedy nad databází Informix a s aplikacemi napsanými v jazyce 4GL) a plní podstatné uživatelské funkce. Pro rok 2006 a další je plánováno, zejména z důvodů technologických, ale i kapacitních, pořízení nového

systému dodavatelským způsobem, ale přitom plně integrovaného do současného systémového prostředí MU. V současné době probíhá výběrové řízení na dodavatele systému, k jehož podmínkám samozřejmě patří spolupráce nového PaM systému s EIS Magion.

2. *Intranetový systém Inet MU.* Inet MU zprostředkovává přístup k vybraným datovým celkům a aplikacím, mimo jiné i k ekonomickým datům zpracovávaným v EIS Magion. Inet tvoří tzv. prezentační vrstvu EIS zpracovanou www-technologiemi s adresným distribuovaným poskytováním informací. Zatímco EIS Magion je nástrojem především pro zaměstnance odborných útvarů MU, poskytuje Inet uživatelsky uspořádané a konkrétně směrované informace všem oprávněným osobám.
3. *Další majetkové aplikace v rámci MU.* Dalšími databázemi a aplikacemi, které úzce souvisejí s ekonomickou oblastí, speciálně s oblastí správy majetku, jsou jednak databáze Budov a místností MU pořizovaná v rámci systému Pasportizace a managementu budov MU a dále identifikace a značení majetku MU čárovými kódy, včetně budování informační podpory pro inventarizaci majetku prostřednictvím čteček čárových kódů. O těchto aplikacích, které jsou v současné době na MU v živém pohybu, pojednávaly dva články v minulém čísle Zpravodaje.
4. *Clearing MU.* Clearing MU je systém centrálního uhrazování závazků a pohledávek mezi univerzitou a osobami v ní působícími, zaměřený na minimalizaci vkladů a výběrů v hotovosti a podporující bezhotovostní formu úhrad pohledávek (přednostně cestou inkasa). Rutinně je provozován pro úhrady kolejného a poplatků za další služby související s ubytováním v kolejích SKM. Způsob vzájemného započítávání závazků MU (tj. záloh na služby poskytované univerzitou a plateb za služby poskytované univerzitě) a pohledávek MU (tj. plateb za služby poskytované univerzitou) vztahujících se k téže osobě, v němž hraje podstatnou roli výklad zákonů, včetně již zmíněného zákona o DPH, není dosud metodicky dořešen a je nepochybně zcela samostatným tématem pro některý z příštích Zpravodajů.

V některém z dalších blízkých čísel Zpravodaje MU nabídneme čtenářům jeden z možných polemických úhlů pohledu na budoucí fungování EIS v rámci naší univerzity.

Literatura

[1] J. Šmarda, I. Jedlička. Ekonomický subsystém IS MU. Zpravodaj MU. ISSN 1212-0901, 1997, roč. 8, č. 2, s. 12–13.

[2] J. Kohoutková. Informační infrastruktura na MU. Zpravodaj MU. ISSN 1212-0901, 2001, roč. 11, č. 5, s. 5–8. □

Serverová infrastruktura informačních systémů MU

Jaromír Ocelka, ÚVT MU

Popis serverového zajištění centrálních informačních systémů MU, jemuž je věnován tento příspěvek, začněme upřesněním: v následujících odstavcích se budeme zabývat pouze serverovým vybavením provozovaným v ÚVT pro podporu centrálního řízení, správy a prezentace univerzity. Diskuse o serverovém zajištění informačních systémů, které nejsou ve správě ÚVT (v prvé řadě studijního IS MU zajišťovaného Fakultou informatiky) nebo jsou úžeji specializovány (knihovní systém zajišťovaný Knihovnicko-informačním centrem ÚVT nebo systém Celouniverzitní počítačové studovny), ponecháváme stranou jako náměty pro samostatné články Zpravodaje a povolanější autory.

Informační systémy pro podporu řízení, správy a prezentace univerzity, které vyvíjí a provozuje ÚVT MU, pokrývají několik aplikačních oblastí. Mezi nejdůležitější patří oblast personalistiky a mezd (dále PaM), oblast ekonomiky (EKO) a oblast vnějších vztahů neboli institucionální www prezentace MU (dále WWW). V posledních letech se do této skupinky důležitých dostává i oblast geografických informačních systémů (GIS). V oblasti PaM byla dosud informační podpora vytvářena v ÚVT ve spolupráci s Personálním odborem RMU. V současné době probíhá výběrové řízení na externího dodavatele nové verze PaM systému, rozšířené o nové komponenty a

funkce. Oblast EKO je zajišťována ekonomickým informačním systémem (EIS) externí dodavatelské firmy Magion (viz také článek P. Vokřínka v tomto čísle Zpravodaje). Informační podpora v oblasti WWW je průběžně vyvíjena v ÚVT, ve spolupráci s RMU. V důsledku přibývajících požadavků na PaM a EKO (požadavků na rozšiřování funkcionality a zejména na umožnění přístupů uživatelů z celé MU) a také na budování informační podpory v dalších aplikačních oblastech vznikl v roce 2000 intranetový server Inet (viz [3]).

Provoz těchto systémů zajišťuje řada serverů – datových, aplikačních, terminálových a webových – jimž je v zabezpečené zóně počítačových sálů ÚVT vyhrazen prostor s potřebným technickým zázemím. S rostoucí důležitostí informačních systémů a rostoucími požadavky na jejich bezvýpadkový provoz dochází postupně k duplikování (pomocí network load balancingu a klastrování) jednotlivých serverů s cílem zajistit okamžitou zástupnost v případě hardwarových výpadků. Do budoucna je plánováno umístění redundantních serverů do samostatné lokality.

1 Centrální serverová infrastruktura

V následujících odstavcích stručně představíme hlavní servery, na nichž jsou provozovány centrální informační systémy MU zajišťované z ÚVT. Uvedeme pouze provozní servery, k nimž samozřejmě existují další podpůrné servery – pomocné, vývojové a zálohovací. Výčet všech serverů, včetně podrobnějších technických specifikací, je pro zájemce k dispozici na internetové adrese [1].

1.1 Databázové servery

Data aplikačních oblastí PaM, EKO, WWW a GIS jsou současné době uložena na třech různých databázových serverech.

Prvním ze serverů je server *bombur* (Sun Enterprise 450, 1GB RAM) s instalovaným databázovým strojem Informix, který slouží především pro provoz PaM a EKO. Protože firma Informix byla pohlcena firmou IBM, která vyvíjí vlastní databázový server, je v současnosti budoucnost db Informix nejistá a směřuje spíše k útlumu dalšího vývoje. ÚVT bylo tedy v loňském roce postaveno před nutnost zvolit jinou, stabilnější platformu. Zvolen byl produkt firmy Oracle – Oracle 10g – v neposlední řadě i s ohledem na kompatibilitu a integraci s aplikační oblastí studia a výuky, kterou na MU zajišťuje studijní systém IS MU běžící rovněž nad databází Oracle.

Během loňského roku byl proto pořízen a zprovozněn nový server nazvaný *amber*, provozující databázi Oracle, a to v robustním klastrovém řešení. Amber je tedy ve skutečnosti dvojice totožných serverů (Sun Fire V240, 8GB RAM) využívajících společné diskové pole (Sun Storedge 3510) a případný výpadek jedné hardwarové komponenty neznamená nedostupnost dat. V době letošních letních prázdnin budou z bomburu na ambery přesunuta data EKO, takže na bomburu nadále zůstanou pouze data PaM.

Posledním z trojice centrálních databázových strojů provozovaných v ÚVT je Microsoft SQL Server sloužící oblastem WWW a GIS a provozovaný na serveru nazvaném *pandora* (Dell PowerEdge 2650, 2GB RAM). Z ostatních databázových serverů jsou na pandoru pravidelně jednou denně přenášeny velké objemy dat určené pro internetovou prezentaci. Vzhledem k charakteristice přístupu k datům WWW (pouze čtení) není nutno řešit pandoru jako databázový klastr, ale postačuje řešení záložním databázovým serverem nazvaným *epimetheus* (Dell PowerEdge 750) s podporou automatického přepínání v případě výpadku pandory.

1.2 Aplikační servery

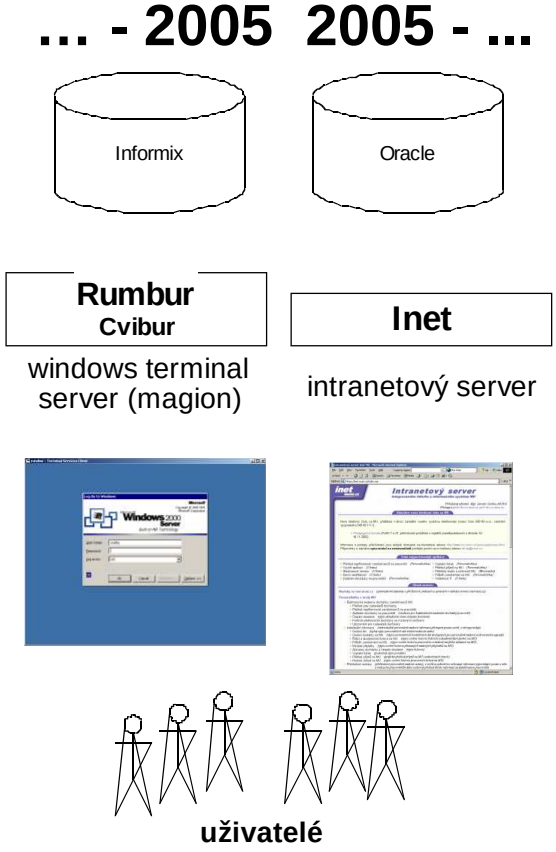
Nad databázovými servery pracují buď přímo úzce specializovaní klienti nebo aplikační servery.

Na základě předchozích zkušeností bylo v ÚVT rozhodnuto o postupném přesunu specializovaných klientů z pracovních stanic uživatelů na terminálové servery a to především z důvodů bezpečnosti a snazší údržby. Terminálové servery jsou provozovány především pro potřeby EKO a rovněž pro potřeby GIS, v obou případech na platformě Microsoft Windows 2000 Server. Pro potřeby EKO je provozován terminálový server *rumbur* a jeho cvičná verze *cvibur*. Z důvodu

počtu uživatelů a potřeby zajistit bezvýpadkový provoz se pod názvem rumbur skrývají dva servery (Dell PowerEdge 1550, 2 GB RAM a HP ProLiant DL360 G3, 2 GB RAM) propojené pomocí network load balancingu. Pro potřeby GIS je provozován terminálový server *tsbaps* (Intel Pentium 4 3GHz, 2 GB RAM).

Pro potřeby EKO a PaM je provozován aplikační server J2EE jimž je produkt Weblogic firmy BEA implementovaný v jazyce Java. Aplikační server je umístěn na serveru *oberon* (Sun Fire 280R) a také tento server je nutno zajistit proti možným výpadkům – na letošní rok je proto plánováno pořízení a implementace jeho klastrové verze.

V oblastech WWW a GIS je využita technologie dynamických www stránek. Aplikační vrstva je tu realizována prostřednictvím ASP a JSP stránek provozovaných na serveru *saturn*. Pro zajištění trvalé dostupnosti je saturn realizován jako dvojice hardwarových serverů (Intel Pentium 4 1,8 GHz) propojených v network load balancingu. GIS oblast navíc využívá specializovaný mapový aplikační server ArcIMS, který je provozován na serveru *razor* (Intel Pentium 4, 2.8 GHz).



Obrázek 1: Serverová infrastruktura ekonomických IS

2 Informační systémy

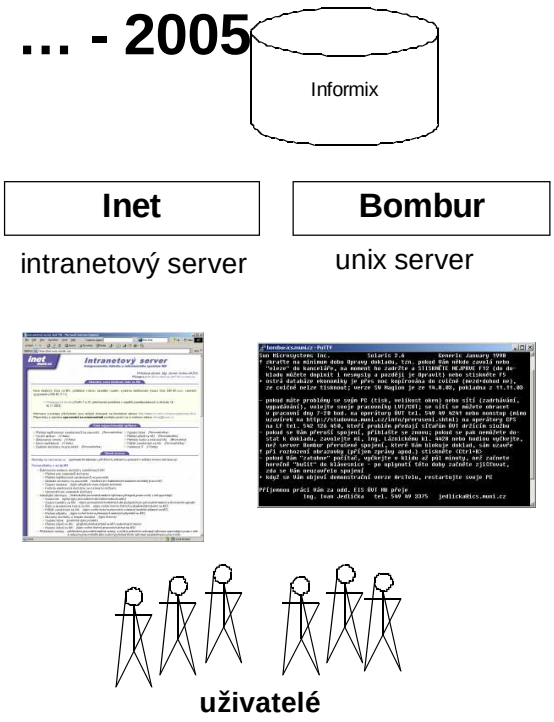
V předchozí části jsme představili hlavní centrální servery a nyní se podívejme, jak tyto servery slouží jednotlivým informačním systémům.

2.1 Ekonomické IS

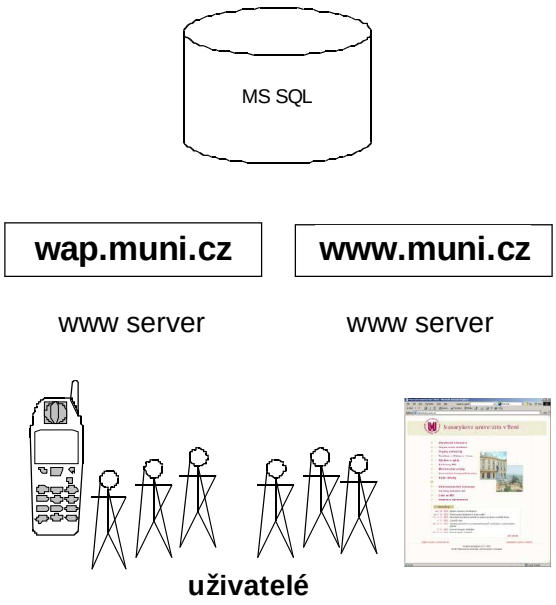
Oblast EKO je obhospodařována především softwarem firmy Magion, který je postaven na dvouvrstvé architektuře – program je určen pro běh na klientské stanici včetně aplikační logiky a přímo přistupuje do databáze (mluvíme tady již jen o grafické verzi EIS Magion, nikoli o původní textové verzi, jejíž provoz na MU v nejbližších dnech skončí, viz článek P. Vokřínka). Pro odstranění nedostatků dvouvrstvé architektury (nutnost instalace na stanicích uživatelů, jichž je dnes na MU více než 300, aj.) je EIS Magion provozován na terminálovém serveru rumbur. Uživatelé tedy mají na své pracovní stanici pouze program pro přístup na vzdálenou plochu serveru a samotný systém Magion běží na vzdáleném serveru. Pro zaškolení nových uživatelů a otestování nových verzí systému je vyhrazen terminálový server cvibur.

Požadavky na zpřístupnění vybraných ekonomických dat (účetních, majetkových nebo clearingových) mají i vedoucí ekonomických zakázek, vedoucí pracovišť, referenti majetku i jednotliví zaměstnanci – tedy uživatelé, kteří s ekonomickým systémem běžně nepracují a pro něž je zbytečné nebo nemožné pořizovat licence pro práci na terminálovém serveru. Těmto uživatelům je vyhověno prostřednictvím www prohlížeče: byly pro ně vybudovány aplikace v intranetovém systému Inet přístupném na adrese `https://inet.muni.cz/`. Inet je postaven na bázi aplikačního serveru J2EE provozovaného na serveru oberon.

Celkové schéma serverové infrastruktury ekonomických IS ukazuje obrázek 1.



Obrázek 2: Serverová infrastruktura personálně-mzdových IS



Obrázek 3: Serverová infrastruktura internetové prezentace MU

2.2 Personálně-mzdové IS

Oblast PaM je provozována nad databázovým serverem Informix. Součástí instalace databáze je i programovací jazyk 4GL, ve kterém byla vytvořena programová podpora oblasti PaM. Uživatelé (personalistky a mzdové referentky) používají pro přístup k aplikacím na serveru bombur běžný telnet klient.

Nový personálně-mzdový systém, k němuž právě probíhá výběrové řízení, bude provozován v případě dvouvrstvé architektury na stávajících terminálových serverech rumbur, jinak bude pro něj pořízen samostatný dedikovaný server. Data nového systému budou uložena v databázovém klastru Oracle na serveru amber.

Jednotliví zaměstnanci a vedoucí pracovišť mají a i nadále budou mít potřebné informace dostupné v intranetovém serveru Inet, obdobně jako mají v Inetu přístupné vybrané ekonomické informace.

Přehledové schéma serverové infrastruktury personálně-mzdových IS zachycuje obrázek 2.

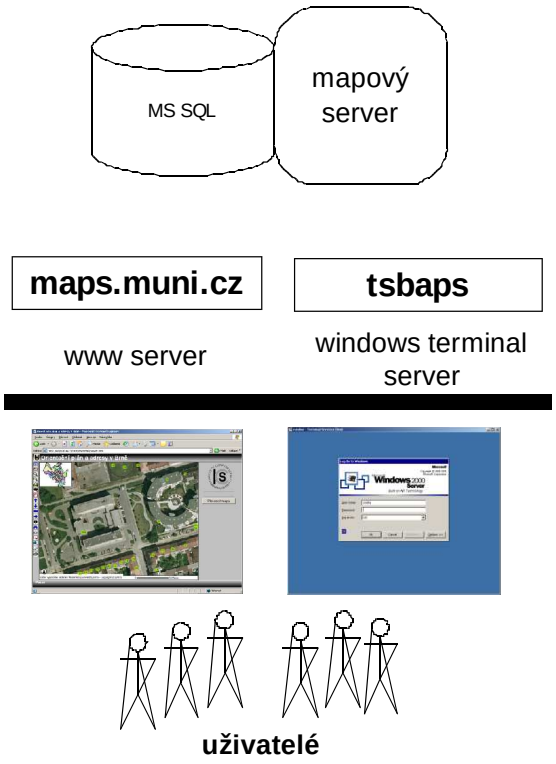
2.3 Internetová prezentace MU

Institucionální veřejné stránky na adrese `http://www.muni.cz/` jsou postaveny na technologiích firmy Microsoft. WWW požadavky si ke zpracování mezi sebou rozdělují dva servery saturn. Aplikační logika používá databázi na serveru pandora s připravenými daty ze všech potřebných univerzitních systémů.

Podrobnější informace o technologickém zázemí www prezentace lze nalézt v [2] a přehledové schéma na obrázku 3.

2.4 Geografické informační systémy

Z původního požadavku mít srozumitelným způsobem evidovánu kabeláž interní sítě MU vznikl informační systém IS BAPS (viz [4]), který je nyní součástí geografického informačního systému. Ten není pro uživatele k dispozici samostatně jako výše popsané systémy, ale prostupuje do ostatních oblastí. Například návštěvníci www prezentace mohou nalézt mapy se zakreslenými budovami. Pro specializované aplikace je provozován terminálový server tsbaps. Jednotlivé aplikace a moduly používají databázi na pandoře, pro vytvoření příslušné mapy z dat je využíván speciální mapový server razor.



Obrázek 4: Serverová infrastruktura geografických IS

2.5 Další aplikační oblasti

Aplikační oblasti, které jsme výše popsali, patří na univerzitě k nejdůležitějším, ale nejsou jediné. Například telefonní ústředna má vybudovánu svou aplikační nadstavbu v Inetu. Tamtéž se plánuje rozsáhlejší podpora V&V. Tyto a případné další nové oblasti využívají již vybudované technologické zázemí – tj. aplikační server J2EE s klastrovým Oraclem.

3 Nejbližší plány do budoucna

Úkolem, který je nyní bezprostředně na řadě, je přesunout data EKO z databázového serveru bombur na ambery a tím na bomburu ponechat pouze data PaM. Bude-li právě probíhajícím výběrovým řízením vybrán dodavatel nové verze personálně-mzdového systému, pobudou data PaM na bomburu již jen několik měsíců, protože jedním z požadavků na nový systém, který by měl být uveden do provozu k 1. lednu 2006, je i uložení dat v databázi Oracle (samozřejmě v klastrovém řešení).

Dalším úkolem, také naplánovaným na letošní rok, je převedení aplikačního a webového serveru oberon, provozujícího SW WebLogic pro Inet MU, na klastrové řešení.

Do třetice bude ještě letos potřeba posílit výkon terminálového serveru cvibur, sloužícího cvičnému a školicímu provozu EIS Magion. Podle nárůstu počtu uživatelů provozního terminálového serveru rumbur bude možná nutné podobně posílit i tento.

A konečně to, co bylo řečeno hned v úvodu – klastrově řešené servery bude nutno rozložit do různých geografických lokalit.

Co a jak se povedlo při řešení těchto a dalších úkolů týkajících se zajišťování provozu a vývoje centrální serverové infrastruktury MU, nejlépe poznají uživatelé informačních systémů MU na své vlastní práci, ale i na stránkách Zpravodaje budeme na toto téma pamatovat.

Literatura

[1] Servery informačních systémů `http://www.ics.muni.cz/25let/technika/is_servery.html`

[2] J.Ocelka. Cluster www-serverů MU. Zpravodaj ÚVT MU. ISSN 1212-0901, 2003, roč. 3, č. , s. 5–8.

[3] J. Měcháček, J. Kohoutková. Intranetový server Informačního systému MU. Zpravodaj ÚVT MU. ISSN 1212-0901, 2000, roč. 11, č. 1, s. 4–7.

[4] P.Glos. IS BAPS – Informační systém Brněnské akademické počítačové sítě. Zpravodaj ÚVT MU. ISSN 1212-0901, 2003, roč. 13, č. 4, s. 13–14. □

Vylaďte si svůj SpamAssassin

Bohuslav Moučka, ÚVT MU

Se spamem, tedy s nevyžádanými zprávami [1], se pravděpodobně již setkala většina uživatelů elektronické pošty. Pro řadu z nich představuje spam vážný problém, který – bez patřičných protiopatření – ohrožuje vůbec použitelnost elektronické pošty jako nástroje efektivní komunikace. Jednou z úspěšných metod boje

proti spamu je používání filtrovacích programů. Ty se snaží „oddělit zrno od plev“, tj. automaticky rozeznat řádné zprávy (kterém mají být uživateli doručeny) od nevyžádaných a obtěžujících zpráv (které lze bez dalšího zkoumání například „zahodit“).

Příkladem úspěšného nástroje pro filtraci spamu je volně dostupný systém SpamAssassin [2], využívaný hojně i na MU. Průměrné hodnoty úspěšnosti filtrace při standardním nastavení se u něj mohou pohybovat kolem 90 – 95%. Tuto úspěšnost lze dále zvýšit speciálním nastavením, které bere do úvahy specifické charakteristiky emailové komunikace u daného konkrétního uživatele. Toto „dolaďování“ filtrace SpamAssassinu však již nemusí být snadnou záležitostí a může vyžadovat poměrně hluboké znalosti z oblasti informačních technologií a systému SpamAssassin samotného.

Cílem tohoto článku je poskytnout základní informace o fungování programu SpamAssassin a nabídnout několik tipů, které zkušeným uživatelům mohou pomoci doladit filtrování jejich osobní pošty.

Jak zapnout filtr SpamAssassin

Systém SpamAssassin musí být nejprve nainstalován na vašem poštovním serveru. Chcete-li jej následně použít pro filtraci osobní pošty, je třeba do souboru `.forward` ve vašem domovském adresáři na poštovním serveru zapsat příkaz pro spuštění programu procmail, například:

```
|/packages/run/links/bin/procmail
```

a do souboru `.procmailrc` přidat řádky pro spuštění SpamAssassinu, např.:

```
:0fw
| /usr/local/spamassassin/spamassassin \
  -c /usr/local/spamassassin/rules
:0:
* ^X-Spam-Status: Yes
Mail/spam
```

Toto nastavení aktivuje spamový filtr a současně udává, že rozpoznané spamy budou ukládány do souboru (poštovní složky) `Mail/spam`. [1]

[1] Pro konkrétní informace o aktuálních nastaveních a možnostech využívání filtru SpamAssassin se obraťte na fakultního správce elektronické pošty.

Po příchodu prvního dopisu se v domovském adresáři uživatele vytvoří podadresář `.spamassassin` a v něm mimo jiné i parametrizační soubor `user_prefs`, do něhož můžete zapisovat vlastní parametry a pravidla programu.

Jak již bylo uvedeno výše, při standardním nastavení rozezná SpamAssassin správně kolem 90% spamů. Jeho úspěšnost můžeme zvýšit třemi způsoby: úpravou některých parametrů, doplněním vlastních pravidel a učením programu.

Skóre zpráv

Program SpamAssassin obsahuje pravidla pro vyhledávání textových řetězců typických pro spamy, a řadu předdefinovaných testů. Pokud se některé pravidlo nebo test při zkoumání dané příchozí zprávy uplatní, je zprávě připočten nebo odečten stanovený počet bodů. Dosáhne-li celkové *skóre zprávy* určené hranice (přednastavené na hodnotu 5), je považována za spam. Tuto hodnotu můžete změnit nastavením příslušného parametru. Například

```
required_hits 4
```

snižuje hranici filtrace, takže více zpráv bude považováno za spamy (může tím ale vzrůst počet selhání, kdy za spam je považována i korektní zpráva).

Síťové testy

Kromě hledání vzorů ve zprávách může SpamAssassin také spolupracovat s několika servery, které shromažďují zprávy označené jako spamy některými z mnoha tisíců uživatelů po celém světě. Program zašle serveru kontrolní součet zprávy a dostane odpověď, zda jde o známý spam. SpamAssassin může takto spolupracovat se 3 servery: Vipul's Razor (`http://razor.sourceforge.net`), Pyzor (`http://pyzor.sourceforge.net`) a DCC (`http://www.rhyolite.com/anti-spam/dcc`). Každý server používá vlastní klientský program, který musí být nainstalován na poštovním serveru před spuštěním SpamAssassinu. Následující parametry pak určují, zda SpamAssassin bude tyto servery pro identifikaci spamu využívat (1 značí, že server bude využíván, 0 značí, že nebude):

```
use_pyzor 0
use_razor2 1
use_dcc 1
```

Důvěryhodné sítě

Při analýze zprávy prohlíží SpamAssassin hlavičky "Received" od poslední, zapsané poštovním serverem (relay) na němž běží SpamAssassin, směrem zpět a určuje, zda příslušná adresa je důvěryhodná. Důvěryhodná je poslední relay, celá podsíť typu B (o rozsahu 65 tisíc adres) v níž tato relay leží a privátní sítě (neveřejné adresy). Seznam důvěryhodných adres můžete rozšířit; následující příkaz doplní do seznamu důvěryhodných síti celou síť 147.229.*.* :

```
trusted_networks 147.229/16
```

Nedůvěryhodné adresy jsou hledány na serverech černých listin (black-lists). Je-li adresa nalezena, skóre zprávy bude zvýšeno.

Analýza zpráv

SpamAssassin používá testy pro kontrolu jednotlivých částí zprávy. Testuje hlavičky (header), tělo zprávy bez HTML značek (body), tělo s HTML značkami (rawbody), tělo zprávy bez dekódování MIME částí (full), URI v těle zprávy (uri) a adresy v URI (uridnsbl). Uživatel si může vytvářet i vlastní testy a zapisovat je do souboru `.spamassassin/user_prefs`.

Ukažme si příklad, jak vytvořit pravidlo, které hledá v těle zprávy text „Wysak Petroleum“[2]:

```
body  WYSAK /Wysak Petroleum/
describe WYSAK Includes Wysak Petroleum
score WYSAK 3.5 3.2 2.8 2.5
```

První řádek pravidla uvádí, kde se bude hledat, název pravidla a hledaný výraz. Druhý řádek obsahuje popis, který bude uveden v popisu zprávy, pokud se dané pravidlo uplatní. Třetí řádek obsahuje skóre přičtené zprávě při nalezení výrazu; zde může být uvedena jedna hodnota pro všechny případy nebo 4 pro následující možnosti:

- Bayesovské ani síťové testy se nepoužívají,
- Baysovské testy se nepoužívají, ale síťové ano,
- Baysovské testy se používají, ale síťové ne,
- Baysovské i síťové testy se používají.

V hodnocení zprávy se při uplatnění tohoto pravidla objeví text:

```
*    20 WYSAK BODY:
     Includes Wysak Petroleum
```

Některé základní zásady při sestavování pravidel:

- jméno pravidla má délku maximalně 22 znaků (písmen, číslic a „_“),
- jména začínající „T_“ jsou vyhrazena pro testovací pravidla,
- jména začínající „__“ jsou vyhrazena pro subtesty metatestů.

Metatest je pravidlo, které kombinuje výsledky několika dalších testů pomocí logických operátorů. Testy začínající „__“ jsou subtesty, které nemají skóre a nejsou uvedeny v seznamu pravidel při testování zprávy.

Příklad metatestu[3]:

```
body CLICK_BELOW_CAPS
     /CLICK\s.{0,30}(?:HERE|BELOW)/s
describe CLICK_BELOW_CAPS
    Asks you to click below
body __CLICK_BELOW
    /click\s.{0,30}(?:here|below)/is
meta CLICK_BELOW
    (__CLICK_BELOW && !CLICK_BELOW_CAPS)
describe CLICK_BELOW
    Asks you to click below
```

CLICK_BELOW_CAPS je standardní pravidlo, které je pravdivé, pokud se ve zprávě vyskytují slova CLICK HERE nebo CLICK BELOW zapsána velkými písmeny. __CLICK_BELOW je podtest bez skóre, který je pravdivý, když jsou ve zprávě výše uvedená slova v libovolné kombinaci velkých a malých písmen. Metatest CLICK_BELOW je pravdivý, pokud __CLICK_BELOW je pravdivý a CLICK_BELOW_CAPS nepravdivý, tedy pokud jsou hledaná slova v libovolné kombinaci písmen, kromě všech velkých. Vedle logických operátorů je možné v metatestech používat i operátory aritmetické a porovnávací.

[2] Takovýto či obdobný text se vyskytoval delší dobu v řadě spamů orientovaných na nabídku nákupu akcií.

[3] řádky v příkladu jsou v dvousloupcové sazbě rozděleny (pozn. editora)

Černé a bílé listiny

SpamAssassin používá černé a bílé listiny adres. Pokud je odesílatel na některé listině, je jeho zprávě zvýšeno nebo sníženo skóre. Přidat do listiny adresu odesílatele, jehož dopisy nepovažujeme nikdy za spam, můžeme příkazem

```
whitelist_from certs@cesnet.cz
```

Tato metoda ovšem není příliš bezpečná, protože adresy odesílatelů jsou ve spamech často podvržené. SpamAssassin však umožňuje spojit adresu odesílatele s důvěryhodnou relay. Chceme-li na listinu přidat všechny adresy, ve tvaru *@muni.cz, které přijdou ze stroje v doméně muni.cz, zapíšeme příkaz

```
whitelist_from_rcvd *@muni.cz muni.cz
```

Učení programu

Kromě statických testů se SpamAssassin učí ze všech zpráv, které zpracoval a svoje chování přizpůsobuje, aby maximalizoval přesnost rozeznávání spamů. Používá dvě metody učení: první jsou automatické bílé listiny, druhou jsou Bayesovské filtry.

Automatické bílé listiny

Na rozdíl od „ručních“ černých a bílých listin uvedených výše, jsou automatické bílé listiny (AWL, Auto-Whitelists) založeny na průměrování: pokud Vám někdo pošle dopis, který po vyhodnocení SpamAssassinem získá skóre 20, a následně Vám pošle druhý dopis, který získá skóre 2, pak AWL tyto dvě hodnoty zprůměruje, takže výsledné skóre druhého dopisu je zvýšeno na 11 (tzv. auto blacklisting, založený na „spamovské“ historii). Funguje to i naopak: jestliže tentýž odesilatel pošle dopis se skórem 0 a následně dopis se skórem 7, pak druhému dopisu je skóre sníženo na 3.5 (auto whitelisting, založený na nespamovské historii).

SpamAssasin využívá v systému AWL automatického učení následujícím způsobem: po každé přijaté zprávě je její skóre přičteno k celkovému skóre odesílatele a je zvýšen čítač jeho zpráv. Průměrné skóre je použito k modifikaci aktuální zprávy. Rozdíl průměrného skóre a skóre aktuální zprávy je vynásoben váhou a přičten ke skóre aktuální zprávy. Hodnota váhy je nastavitelná v rozsahu 0 až 1 příkazem

```
auto_whitelist_factor 0.7
```

Defaultní hodnota je 0.5. Nastavíme-li vyšší hodnotu, bude mít větší význam historické skóre, hodnota 1 znamená, že výsledné skóre zprávy se bude rovnat historickému skóre. Hodnota 0 způsobí, že historické skóre bude ignorováno.

Bayesovské filtry

Druhou metodou učení jsou bayesovské filtry. Zde je již zapotřebí osobní zásah konkrétního uživatele; pro využití učení je třeba programu předložit velké množství zpráv obou druhů – nechtěných zpráv (spam) i dobrých normálních dopisů (ham) a tím ho „doučovat“ pro správné rozpoznávání. Aby program pracoval efektivně, měla by každá z jeho databází (spam i ham) obsahovat alespoň tisíc zpráv (čím více, tím lépe). Minimální množství, po němž program začne využívat bayesovskou databázi, je 200 zpráv typu spam a 200 zpráv typu ham (důležité je trénovat SpamAssassin na obou typech zpráv).

Program si podle předložených výukových zpráv vytvoří databázi symbolů (řetězců délky 3–15 znaků) nalezených ve zprávách. Ke každému symbolu si zapíše počet jeho výskytů ve spamu a hamu a čas posledního použití při vyhodnocení zprávy. Symboly, které nebyly použity dlouhou dobu, jsou z databáze vymazány, aby se zvýšila efektivita. V druhé databázi je seznam zpráv, z kterých se program učil. Program učíme příkazy:

```
sa-learn --mbox --spam reklamy
sa-learn --mbox --ham  mojedopisy
```

V prvním příkaze předkládáme programu soubor reklamy obsahující spamy, ve druhém soubor mojedopisy obsahující hamy. Parametr `-mbox`, označuje, že předkládána je poštovní schránka s více zprávami (tyto schránky jsou obvykle uloženy u uživatele v podadresáři Mail nebo mail). Při kontrole je zpráva rozdělena na symboly, které jsou hledány v databázi. Podle výsledku je zprávě přiřazena pravděpodobnost, že jde o spam, což je v hlavičce vyznačeno tím, že zpráva vyhovuje pravidlu např. BAYES_80, tedy, že s pravděpodobností 0.8 - 0.9 jde o spam. Bayesovským pravidlům označujícím pravděpodobnost

menší než 0.5 je přiřazeno záporné skóre, pro pravděpodobnost větší než 0.5 kladné. Po vybudování počáteční databáze symbolů je třeba ji aktualizovat. Buď to můžeme programu předložit všechny zprávy nebo jen zprávy, které chybně označil.

Používání bayesovských filtrů můžeme ovlivnit několika parametry. Pokud bychom chtěli bayesovské filtry zcela zakázat, nastavíme parametr:

```
use_bayes 0
```

Následujícím parametrem určíme, že program se bude automaticky učit (což je implicitní nastavení, 0 znamená, že se učit nebude)

```
bayes_auto_learn 1
```

Nastavíme, že program se bude automaticky učit ze spamů s vyšším skóre než 8 a z hamů s nižším skóre než -2. (Do tohoto skóre se nepočítá skóre z bayesovských pravidel a z bílých a černých listin.)

```
bayes_auto_learn_threshold_spam 8.0
bayes_auto_learn_threshold_nonspam -2
```

Program může vyloučit ze zpracování zadané pole záhlaví zprávy, které by mohlo být při učení zavádějící (zpravidla jde o pole generované jiným antispamovým nebo antivirovým programem), například:

```
bayes_ignore_header X-Muni-Spam-TestIP
```

Možností, jak dolaďovat přesnost filtrace SpamAssassin u své osobní pošty, je samozřejmě mnohem více, než bylo možné popsat v tomto krátkém informačním článku. Zájemce o další podrobnosti můžeme odkázat na domovské stránky systému SpamAssassin [2] a na čtivou dokumentaci na SpamAssassin-Wiki [3].

Literatura

[1] M. Kolaja, M. Bartošek. Jemný úvod do (anti)spamové problematiky. Zpravodaj ÚVT MU. ISSN 1212-0901, 2002, roč.12, č.5, s.1-6

[2] Domovská stránka projektu SpamAssassin. http://spamassassin.apache.org/

[3] SpamAssassin-Wiki. http://wiki.apache.org/spamassassin/ □

Správa soukromých klíčů pomocí hardwarových tokenů

Daniel Kouřil, ÚVT MU

PKI (*Public Key Infrastructure*) nabízí velmi bohaté možnosti pro realizaci silných autentizačních mechanismů a je také základem řady rozšířených protokolů používaných pro zabezpečení síťové komunikace (jako je např. SSL). Vedle svých silných stránek však také model PKI trpí neduhy, které plynou z principů, na kterých je založen a mohou velmi často snižovat celkovou bezpečnost systémů založených na tomto modelu. Tento článek se zabývá problémy, které se týkají správy klíčů v PKI a popisuje jejich možné řešení pomocí specializovaných HW zařízení.

Základním principem PKI je použití dvojice soukromého a veřejného klíče, kterou každý zúčastněný uživatel či služba vlastní a spravuje. Soukromý klíč se používá pro vytváření elektronických podpisů, které lze následně pomocí odpovídajícího veřejného klíče ověřit a zkontrolovat tak, že data skutečně pocházejí od majitele dvojice klíčů a že nebyla nijak modifikována při přenosu. Naopak pomocí veřejného klíče je možné zašifrovat data tak, že je lze dešifrovat pouze pomocí odpovídajícího klíče a je tak zaručeno, že je může přečíst opravdu pouze určený adresát, tj. majitel dotyčného páru klíčů. Jak plyne z názvu i popisu, soukromý klíč je určen pouze pro použití svým majitelem, který jej musí držet v tajnosti. Naproti tomu veřejný klíč je distribuován v rámci celé komunity a je volně k dispozici komukoliv, kdo má zájem navázat bezpečné spojení s majitelem tohoto klíče. Veřejné klíče se nejčastěji distribuují ve formě certifikátů veřejného klíče vydávaných certifikačními autoritami. Certifikáty obsahují identifikaci majitele klíče a další informace, které jsou důležité pro komunikaci, jako je doba platnosti klíče, adresa seznamu revokovaných certifikátů, emailová adresa uživatele či síťová adresa služby ad. Samotný proces vybudování kvalitní a důvěryhodné certifikační autority, definování certifikačního procesu a vytvoření systému důvěry mezi certifikačními autoritami a uživateli je zcela klíčový pro nasazení PKI, ale je mimo rozsah tohoto článku. Více informací o současných trendech v oblasti certifi-

kačních autorit lze nalézt např. v minulém čísle Zpravodaje [1].

Správa soukromých klíčů je jedním ze základních problémů, se kterými se oblast PKI potýká. Klíče používané v PKI jsou velmi dlouhé řetězce znaků a na rozdíl od hesel si je člověk nemůže zapamatovat. Musí být proto uloženy v nějaké elektronické podobě tak, aby jej mohla přečíst aplikace, kterou uživatel používá. Nejčastěji se dnes klíče ukládají na lokální disk počítače, buď ve formě samostatného souboru nebo ve specializovaném úložišti, které poskytuje aplikace, příp. operační systém. Vždy se ale jedná o data, která jsou uložena na disku počítače a tudíž čitelná komukoliv, kdo má oprávnění číst příslušnou část disku. Pro ochranu před neoprávněným přístupem k těmto datům se používá šifrování souborů heslem, které musí uživatel zadat při přístupu k soukromému klíči. Navíc, pokud to použitý souborový systém dovoluje, bývá přístup k datům na disku chráněn proti přístupu jiných uživatelů na úrovni operačního systému. Úroveň takové ochrany soukromé klíče je ale velmi křehká, zejména pokud se případnému útočníkovi podaří získat práva majitele soukromého klíče nebo administrátora příslušného systému. Technik jak získat příslušná data z lokálního disku je celá řada, od použití počítačových virů, přes zneužití různých chyb v aplikacích běžících na daném počítači, až k technikám sociálního inženýrství, které zřejmě právě zažívají renesanci v podobě tzv. *rhybaření*[1]. Pokud se útočníkovi povede získat soubor se soukromým klíčem, může se pokusit najít správné heslo k rozšifrování tohoto souboru. Jelikož má data plně pod kontrolou a může je libovolně zpracovávat, např. nasadit klasické techniky pro lámání hesel, které známe z jiných oblastí – jako je hádání hesel hrubou silou nebo slovníkový útok.

Dostáváme se tak k další problematické oblasti správy klíčů, kterou je fakt, že zabezpečení souboru s klíčem je z velké části v rukách samotného uživatele. Navíc v oblasti PKI neexistují mechanismy, které by spolehlivě zajistily, že soubor s klíčem je patřičně ochráněn, tj. že použité heslo je dostatečně silné, aby odolalo běžným útokům, že jsou správně nastavena přístupová práva k souboru s klíčem apod. Uživatelé také mohou (a často tak také činí) libovolně manipulovat se souborem s klíčem, např. jej kopírovat na jiné počítače, kde klíč potřebují a při těchto operacích může také dojít k prozrazení obsahu soukromého klíče. Důsledkem zneužití těchto problematických míst pak může být velmi oslabený systém.

Výrazné zvýšení bezpečnosti by přineslo uložení soukromých klíčů na bezpečnější místo tak, aby neležely přímo na disku stroje, ale místo toho byly na nějakém jiném médiu, které se použije pouze v případě potřeby.

Hardwarové tokeny

Pro bezpečnější ukládání soukromých klíčů existuje několik alternativ, které se liší ve způsobu práce s nimi i v zabezpečení, které uloženým klíčům poskytují.

Nejjednodušší možností je použít nějaký typ vyjímatelného média, jako je např. disketa, CD-ROM nebo populární USB flash disk, na které se soukromý klíč uloží místo pevného disku. Po dobu práce je médium s klíčem zapojeno do počítače a aplikace s klíčem pracují stejně, jako by byl přímo na pevném disku počítače, tj. přistupují k souboru na výměnném médiu. Po ukončení práce uživatel vyjme médium z počítače a klíč tak není v počítači nadále dostupný a nemůže se stát předmětem útoku. Tento postup je sice jednoduchý, ale neposkytuje žádnou ochranu pro klíč v okamžiku, kdy je médium s klíčem připojeno k počítači. Navíc se zvyšuje riziko prozrazení klíče, protože médium je přenosné a může se ztratit nebo být ukradeno. Naopak výhodou tohoto přístupu je fakt, že jej lze začít používat okamžitě a nejsou potřeba žádné změny v aplikacích.

Další možností je použití *čipových karet* a příbuzných technologií, které obsahují jak chráněný prostor, do kterého lze uložit soukromý klíč s certifikátem, tak i samostatný procesor, který je schopen s těmito klíči pracovat a provádět s nimi základní kryptografické operace. Karta

[1] rhybaření (*phishing*) je technika, kterou se útočníci snaží z uživatelů vylákat citlivé informace (čísla kreditních karet nebo hesla) pomocí podvržené komunikace tvářící se, že opravdu přichází od oficiální instituce (jako je banka, administrátor systému). Viz také `http://en.wikipedia.org/wiki/Phishing`

je k počítači připojena pomocí čtečky zapojené přes USB nebo sériový port, pomocí které komunikují aplikace s kartou. Aplikace tak nepoužívají přímo soukromý klíč, ale předávají kartě data, která jsou zpracována procesorem na tokenu a výsledek je vrácen zpět aplikaci. Klíč tak nikdy neopustí kartu a není jej možné nijak zkopírovat. Přístup ke kartě je autentizován, tj. aplikace se musí procesoru na kartě nejprve prokázat znalostí příslušného PINu, který zadá uživatel. Je tak zabráněno zneužití informací z karty v případě její ztráty. Většina karet je konstruována tak, že se po zadání určitého počtu chybných PINů zablokuje a jedinou možností jak ji zprovoznit je její nová inicializace, která však nevratně smaže všechny informace na kartě. Vedle čipových karet s čtečkami také existují *čipové tokeny* připojitelné do USB, které kombinují funkcionalitu karty a čtečky v jednom kusu hardware. Vzhledem se podobají USB flash diskům, ale vnitřní architektura je totožná s čipovými kartami, tj. obsahují vlastní procesor a není možné přistupovat přímo k citlivým datům na tokenu. Výhodou tokenů je jejich vyšší mobilita, protože není potřeba s sebou nosit kartu i čtečku. Další výhodou je jejich tvar, protože vzhledem k jejich malé velikosti je lze připojit např. ke svazku klíčů, takže se snižuje riziko, že zůstanou zapomenuté v počítači.

Technologie čipových karet a tokenů výrazně zvyšují ochranu soukromých klíčů, protože umožňuje jejich bezpečné uložení a přístup k nim. Zavádí pojem tzv. dvoufaktorové autentizace (*two-factor authentication*), kdy uživatel musí prokázat znalost nějakého tajného kódu (tj. PINu k tokenu) a také fyzické držení tokenu.

Praktické nasazení hardwarových tokenů

Tato kapitola popisuje prostředí vytvořené v průběhu řešení projektu „Univerzální autentizace pomocí hardwarových tokenů“, jehož cílem je nasazení tokenů v *META Centru*. *META Centrum*[2] je aktivita sdružení CESNET, která buduje a provozuje gridovou infrastrukturu v akademické síti CESNET2. Bezpečnostní infrastruktura *META Centra* je založena na autentizačním mechanismu Kerberos, kde se pro iniciální autentizaci uživatelů používá heslo. Vzhledem k popularitě a možnostem, které skýtá PKI jsme se rozhodli podporovat i tento mechanismus a zároveň vytvořit prostředí, které umožní vyřešit jednu z největších slabin PKI, kterou je správa soukromých klíčů. Vytvoření této podpory a vybavení aktivních uživatelů *META Centra* hardwarovými tokeny je cílem výše zmíněného projektu, který je řešen pod hlavičkou Fondu rozvoje sdružení CESNET a jehož řešiteli jsou MU v Brně, UK v Praze a ZČU v Plzni.

META Centrum vytváří virtuální organizaci, všichni jeho uživatelé jsou rozprostření po celé ČR a jsou primárně zapojeni v infrastruktuře svých domovských institucí. *META Centrum* nemá žádné nástroje (ani ambice), jak ovlivňovat lokální nastavení jednotlivých institucí. Naší hlavní snahou při řešení projektu HW tokenů proto bylo vybrat taková zařízení a programová vybavení, která všem uživatelům umožní hladké zapojení tokenů do stávajícího prostředí. Jednou z priorit proto bylo umožnit práci s tokeny na více platformách, nezbytnou podmínkou byla práce jak na MS Windows, tak i Linuxu. Vzhledem k tomu, že uživatelé i migrují mezi různými systémy, snažili jsme se najít řešení, které umožní přecházet mezi těmito platformami při zachování plné funkčnosti tokenu. Jelikož nezbytnou součástí projektu byly změny naší současné infrastruktury, zaměřovali jsme se především na použití open-source aplikací, které v případě potřeby umožňují provádět snadné zásahy do kódu.

Ve fázi výběru nejvhodnějšího typu tokenu jsme testovali několik vzorků jak čipových karet a čteček, tak i USB tokenů. Od začátku jsme sice preferovali spíše USB tokeny, zejména pro jejich snadnější fyzickou přenositelnost, ale chtěli jsme ověřit, že USB zařízení jsou skutečně ekvivalentní klasickým čipovým kartám. Mezi kritéria, která jsme vyhodnocovali, patřila zejména schopnost tokenu či karty provádět kryptografické operace a chránit soukromý klíč. Dále jsme se zaměřili na podporu příslušného typu tokenu v současných open-source produktech a možnosti používat zařízení na různých OS. Ověřovali jsme také podporu pro běžně používané standardy PKCS11 a PKCS15, které umožňují vyví-

[2] `http://meta.cesnet.cz/`

jet a používat aplikace nezávisle na konkrétním typu zařízení. Vyhodnocení ukázalo, že testované USB tokeny jsou skutečně funkčně zcela ekvivalentní čipovým kartám.

K dalšímu testovaní a pilotnímu provozu jsme vybrali USB token iKey 3000 od firmy Rainbow (nyní SafeNet). Každý token se dodává s ovladači a základním programovým vybavením pro OS Windows a Linux. Instalace na OS Windows byla bezproblémová, pro použití na OS Linux jsme se rozhodli nepoužívat dodávané ovladače a software, které byly dodávány pouze v binární verzi, nefungovaly ve všech Linuxových distribucích a zejména se nepodařilo je bezproblémově zaintegrovat do middleware *META Centra*. Pro tokeny iKey 3000 však existují kvalitní alternativní ovladače pro Linux i další open-source software. Pomocí těchto alternativních ovladačů lze bez problémů používat token, který byl inicializován originálním ovladačem a softwarem, takže jej lze snadno přenášet mezi různými systémy. Bohužel opačná možnost nefunguje, open-source nástroje nejsou schopny inicializovat token ve formátu, který by byl čitelný originálním softwarem. Nepovažujeme to však za významnou obtíž, protože uživatelé, kteří přechází mezi více systémy, mají vždy možnost inicializovat token v prostředí Windows pomocí originálního vybavení.

Bez výraznějších problémů se povedlo zprovoznit podporu tokenů ve frekventovaných aplikacích. V prostředí MS Windows jsme token testovali s aplikacemi Internet Explorer, Outlook, Mozilla Thunderbird a Mozilla FireFox. Lze tak snadno používat klientskou autentizaci při přístupu na chráněné www stránky, podepisovat, resp. dešifrovat emailovou komunikaci. Poslední tři aplikace spolupracují s tokeny i v prostředí Linuxu. Pro vzdálené přihlašování lze použít aplikace PuTTY nebo OpenSSH, které mají podporu pro hardwarové tokeny a lze je používat jak v prostředí MS Windows, tak i v Linuxu. V prostředí obou systémů také funguje balík OpenSSL, který umožňuje provádět kryptografické operace s tokenem. Obecně lze říci, že podporu tokenů lze zprovoznit ve většině aplikací, které podporují rozhraní PKCS11.

Pro programování vlastních aplikací, které používají token, jsme použili open-source knihovnu OpenSC[3], která je dostupná ve verzích pro Linux i MS Windows. Tato knihovna nám umožnila vytvořit aplikace pro tokeny nezávisle na konkrétním operačním systému. Pro správnou funkci této knihovny je potřeba mít instalované ovladače konkrétního tokenu, ale protože knihovna umí komunikovat jak s originálními ovladači (na MS Windows), tak s alternativními ovladači (na Linuxu), mohli jsme se soustředit na vývoj vlastní aplikace a nezatěžovat se nižšími detaily komunikace s tokenem.

Nedílnou součástí projektu byla úprava stávající infrastruktury *META Centra* tak, aby umožnila hladké nasazení hardwarových tokenů. Vedle zprovozňování a konfigurace dodaného vybavení jsme se soustředili na vývoj vlastního software a potřebných nástrojů. Nejdůležitějším úkolem bylo připravit současnou autentizační infrastrukturu tak, aby umožňovala použití PKI autentizace pomocí hardwarových tokenů. Bezpečnostní infrastruktura *META Centra* je založena na mechanismu Kerberos, který podporuje autentizaci pomocí hesla a symetrické kryptografie. Jedním z prvních úkolů projektu tedy bylo upravit protokol Kerberos tak, aby umožňoval i autentizaci pomocí PKI certifikátů. Využili jsme aktivit standardizační organizace IETF a převzali tehdejší návrh úprav protokolu a podle tohoto návrhu jsme navrhli a realizovali změny do implementace protokolu Kerberos. Jednalo se o velmi komplexní zásah do kódu, ale výsledkem byla funkční a kompatibilní realizace tohoto rozšíření. Výsledná verze byla přejata do standardní distribuce a v současné době je dále vyvíjena a používána i dalšími organizacemi ve světě, které nasazují čipové technologie v prostředí s protokolem Kerberos.

Vedle zapojení tokenů do stávající infrastruktury *META Centra* jsme nachystali nástroje, které umožní použití tokenů v mezinárodním gridovém prostředí, které používá PKI a speciální tzv. proxy certifikáty (viz také [1]). Využili jsme vlastností, které pro manipulaci s proxy certifikáty nabízí knihovna OpenSSL a pomocí ní implementovali program, který pomocí tokenu generuje proxy certifikáty.

[3] `http://www.opensc.org/`

META Centrum vždy úzce spolupracovalo s certifikační autoritou sdružení CESNET. Certifikáty vydané touto CA jsou uznávány širokou komunitou v rámci celé Evropy a držitelé těchto certifikátů se tedy mohou snadno zapojovat do mezinárodních projektů. Pro podporu uživatelů *META Centra* jsme dohodli zřízení Registrační autority pro CESNET CA, která bude provozována na ÚVT MU a bude usnadňovat získání certifikátu jak pro uživatele *META Centra*, tak i MU.

V současné době je infrastruktura *META Centra* připravena k nasazení tokenů. Administrátoři byli vybaveni vybranými USB tokeny již dříve a v současné době je používají k administrátorským účelům. Připravili jsme nákup většího množství tokenů a připravujeme v brzké době jejich distribuci mezi cca 150 aktivních uživatelů.

Vedle přípravy technického zázemí bylo úkolem projektu vyřešit i logistické problémy s distribucí tokenů. Situace *META Centra* je odlišná od jiných prostředí, protože máme velmi distribuované uživatele, což velmi komplikuje fyzické předání tokenů. Nakonec jsme se rozhodli uspořádat krátké semináře s jednotlivými skupinami uživatelů, které se budou konat přímo v jejich lokálních institucích. V rámci těchto seminářů uživatelům předáme tokeny, předvedeme jejich funkcionalitu, jejich zapojení do *META Centra* a příp. gridových aktivit a zejména vyřešíme na místě otázky spojené s žádostí o certifikát od CESNET CA, protože k tomuto kroku je nutný fyzický kontakt s pracovníkem CESNET RA, který ověří identitu žadatele. Věříme, že výsledkem bude hladké zapojení tokenů do infrastruktury a tím zvýšení celkové bezpečnosti *META Centra*. Navíc výsledky a zkušenosti s touto netechnickou fází budou cenné do budoucna, protože řada velkých mezinárodních projektů se začíná zabývat myšlenkami na použití hardwarových tokenů a jejich distribuce rozsáhlém prostředí je samozřejmě klíčová.

Literatura

[1] D. Kouřil. „Bezpečnost v distribuovaném prostředí.“ *Zpravodaj ÚVT MU*. 2005, roč. 15, č. 4, s. 2–6. □

Obecná veřejná licence GNU

Ladislav Lhotka, CESNET, z.s.p.o.

Čtvrté pokračování seriálu o open source softwaru [1] věnoval Luděk Matyska problematice softwarových patentů. Je docela dobře možné, že právě neblahé prolínání patentové ochrany do sféry autorského práva bude velkou překážkou rozvoje open source softwaru, a nejen jeho. Souboj na evropském patentovém poli v poslední době připomíná houpačku: Evropská komise a Rada EU se nadále snaží obejít odmítavé stanovisko Evropského parlamentu a prosadit co nejdříve svou verzi směrnice „o patentovatelnosti vynálezů realizovaných počítačem“, která by otevřela značné možnosti pro skryté patentování počítačových programů. Je povzbudivé, že český Senát ve svém usnesení z 31. března 2005 podpořil odmítavé stanovisko EP a doporučil vládě zasadit se o demokratické a neuspěchané projednání této otázky v orgánech EU a trvat na jasném odmítnutí počítačových programů jakožto předmětu patentové ochrany. Velkou iniciativu v boji proti softwarovým patentům vyvíjí také poslankyně EP Zuzana Roithová, viz její webové stránky[1]. V polovině května 2005 se legislativní výbor EP rozhodl požádat o pomoc právní experty a připojit ke směrnici dodatky, které by zcela jednoznačně znemožňovaly patentování „čistých“ počítačových programů. Takže uvidíme ...

V tomto článku se však chci vrátit zpět k problematice softwarových licencí a detailněji analyzovat úhelný kámen svobodného softwaru – Obecnou veřejnou licenci GNU (GNU GPL). Pokusím se vysvětlit její hlavní zásady, problémy její aplikace v našem autorském právu a také pravděpodobný budoucí vývoj, který bude zřejmě – světe div se – mimo jiné velmi významně reflektovat otázku softwarových patentů. Zdůrazňuji, že jako laik obojího práva si nedělám nárok na správnost svých právních interpretací a uvítám v tomto směru kritické vyjádření autorskoprávních expertů.

[1] `http://www.roithova.cz/`

GPL – čím je a čím není

Obecná veřejná licence GNU je na první pohled trochu zvláštní tím, že kromě obvyklých formulací práv a povinností uživatelů obsahuje i vysvětlení záměru a návod k použití. Takové pasáže v jiných softwarových licencích najdeme stěží. Je to především tím, že GNU GPL není určena pouze jako licence k programům Free Software Foundation, nýbrž je nabízena k použití i jiným autorům, kteří píší svobodný software a nepřejí si jeho využití v programech, které by už svobodné nebyly.

Poselství GNU GPL je v tomto směru celkem jasné a v ideálním světě by proto mohlo stačit stručné a jasné vyjádření takového záměru autora. Realita je bohužel jiná, a tak musí GPL nasadit legislativní páky, které mají zabránit všelijakým chytrákům v obcházení autorovy vůle. Hlavním trikem je tzv. *copyleft*, o němž jsem se zmínil již v úvodním článku tohoto seriálu: software se opatří copyrightem (tedy výhradním právem kopírování) tak, jak je v anglosaském právním systému obvyklé, licenční podmínky však zároveň umožní komukoli studovat zdrojový kód a program v podstatě libovolně používat, kopírovat, modifikovat a dále šířit – ovšem za předpokladu, že všechna odvozená díla poskytnou svým uživatelům tatáž výše uvedená práva.

Čeští právníci se vesměs shodují na tom, že v našem právním systému se GPL ve svém původním znění nehodí jako přímý podklad pro licenční smlouvy. Podrobnější argumentaci je možno nalézt například v článku [2]. Sdružení ZASTUDENA.cz připravilo upravený český překlad GNU GPL, který licenci přizpůsobuje platnému autorskému zákonu 121/2000 Sb. Takovýto postup má ovšem podle mého názoru také své nevýhody. Originální anglická verze GNU GPL je podepřena autoritou FSF i vahou desítek tisíc programů, které ji používají (jen na serveru `freshmeat.net` je jich asi dvacet tisíc). A nejedná se přitom zdaleka jen o programy napsané v USA či Velké Británii. Je proto otázka, nakolik je účelné uchylovat se k jinému, téměř neznámému licenčnímu textu, který takovéto zázemí nemá.

Případné nekompatibilní formulace GNU GPL ale podle mého názoru nepředstavují fakticky žádný velký problém. Pokud to totiž jako *autor programu* myslím s GPL opravdu vážně, nemám se v podstatě čeho obávat: uživatel se buď může podřídit GPL, což je mým záměrem, anebo musí respektovat obecná ustanovení autorského zákona, která kopírování a jiné činnosti povolené v GPL zakazují. Teoreticky hůře by na tom mohl být uživatel, pokud by se v dobré víře řídil ustanoveními GPL a proradný autor by posléze prohlásil licenci za neplatnou a žaloval uživatele z porušení autorského zákona. Nedovedu si ovšem představit, jak by takový autor-padouch mohl s podobnou žalobou uspět, když de facto uživatele uvedl neplatnou licencí v omyl. Otázka platnosti či neplatnosti GPL tak může mít zásadní význam jen pro právníky firem, které se pokoušejí GPL obejít anebo balancovat na hraně toho, co GPL povoluje.

O tom, že GNU GPL má svou právní sílu i v zemích s autorským právem vycházejícím z revidované bernské úmluvy z r. 1971 (kam patří i ČR), svědčí zkušenost z Německa: Zemský soud v Mnichově vydal již dvě opatření proti firmám Sitecom a Fortinet, které GPL porušily tím, že ve svých výrobcích užívaly upravený Linux bez toho, že by zveřejnily zdrojový kód svých modifikací. Je třeba říci, že v obou případech dotyčné firmy své „opomenutí“ napravily. Bližší informace k oběma případům lze najít na `www.gpl-violations.org`.

Odvozené dílo

Jednou z dlouhodobě sporných otázek interpretace licence GPL je pojem odvozeného díla (v angličtině *derivative work*). Náš autorský zákon 121/2000 Sb. jej nijak nedefinuje, pouze v §2, odst. 4 uvádí, že

> Předmětem práva autorského je také dílo vzniklé tvůrčím zpracováním díla jiného, včetně překladu díla do jiného jazyka. Tím není dotčeno právo autora zpracovaného nebo přeloženého díla.

Termín „derivative work“ použitý v originálu GNU GPL má naproti tomu přímou oporu v autorském právu USA. US Code uvádí v titulu 17, kapitole 1 a §101 toto (přeložil LL):

> „Odvozené dílo“ je dílo založené na jednom či několika dříve existujících dílech. Může jím být překlad, hudební aranžmá, dramatizace, beletrizace, filmová verze, zvuková nahrávka, umělecká reprodukce, zkrácená nebo zhutněná verze či jakákoli jiná forma, v níž může být dílo prezentováno anebo do níž může být transformováno či adaptováno. Dílo sestávající z edičních úprav, poznámek, rozvinutí či jiných modifikací, které jako celek představují originální autorské dílo, jsou „odvozeným dílem“.

V případě specifických děl, jimiž jsou počítačové programy, nám však ani tato definice příliš nepomůže. Vyplývá z ní ovšem nepochybně, že program vzniklý modifikací *zdrojového kódu* předlohy je odvozeným dílem, ale jak je tomu v jiných případech? Je spojení programu s knihovnou, jež je chráněna GPL, dílem odvozeným z této knihovny? Záleží na tom, je-li toto spojení (linking) statické, dynamické anebo má dokonce formu vzdáleného volání procedury (RPC) po síti? Tyto a podobné otázky si často kladou ti, kdo mají zájem o využití nějakého programu pod GPL, ale přitom nemohou nebo nechtějí zveřejnit svůj zdrojový kód.

Lawrence Rosen [3] uvádí následující orientační kritéria, podle nichž lze rozhodnout, zda je daný program odvozeným dílem:

1. Pokud byl jakkoli aktivně použit nebo dokonce modifikován zdrojový kód předlohy, jde téměř jistě o odvozené dílo.
2. Pouhé *použití* knihovny prostřednictvím aplikačního programového rozhraní (API) *nečiní* z programu dílo odvozené z knihovny.
3. Pokud program nabízí mechanismus pro připojení samostatných modulů (pluginů, ovladačů zařízení apod.), pak tyto moduly nejsou odvozeným dílem.
4. V případě spojování (linkování) programů je potřeba zvážit, zda je předloha svou podstatou takové spojování umožňuje a předpokládá, tj. je-li například prezentována jako knihovna. Technický způsob takového spojení (statické, dynamické apod.) pak není rozhodující.

Zdůrazněme, že výše uvedená kritéria se tak docela nekryjí s názorem Richarda Stallmana a FSF, zejména proto, že otevírají cestu k poměrně snadnému obcházení GPL. Kupříkladu lze vzít program pod GPL, přidat k němu minimální API pro připojování modulů a výsledek publikovat opět pod GPL. O porušení licence nemůže být zatím ani řeči, ovšem modifikovaný program se tím otevírá rozšířením, která by podle Rosenova výkladu nemusela podléhat GPL. Takové a podobné případy tvoří nepříjemnou šedou zónu, v níž se právní názory liší a definitivní stanovisko v každém konkrétním případě by musel zaujmout až soud.

Výše uvedený scénář byl již prokazatelně a opakovaně použit v jádře Linuxu, které jako celek podléhá licenci GPL. Jak je všeobecně známo, Linux umožňuje už drahnou dobu (tuším od verze 1.2) připojovat k jádru za běhu moduly s nejrůznějšími funkcemi, od ovladačů zařízení přes souborové systémy až po základní věci jako je třeba přidělování paměti. Linus Torvalds v minulosti opakovaně vyhlásil, že tyto moduly nedědí automaticky licenci GPL, pokud ovšem využívají pouze „oficiální“ rozhraní jádra, tj. zejména systémová volání. Jenomže: vezmeme-li v úvahu distribuovaný charakter vývoje jádra, kdo je vlastně schopen určit, co je oficiální součástí jádra a co již ne? Známým případem je třeba ovladač `pwc` pro USB kamery Philips, který byl složen ze dvou částí, u jedné z nichž nemohl být z důvodu výrobcových licenčních omezení zveřejněn zdrojový kód. Druhá část ovladače proto (vedle realizace části funkcí) také připravila „háčky“ pro připojení proprietární části ovladače. Ovladač v této podobě existoval poměrně dlouhou dobu, než se prosadil názor – podle mého správný – že tento ovladač porušuje GPL, a proto byl z jádra odstraněn. Jeho autor se tímto krokem cítil velmi dotčen a vyvolal v poštovní konferenci *linux-kernel* poměrně ostrou diskusi. Nakonec se ale ukázalo, že ovladač lze napsat i v licenčně čisté a navíc i technicky lepší podobě.

V hlavní distribuci jádra Linuxu je tedy ještě celkem možné prosadit zásady, které zabrání erozi GPL. Daleko horší je to ale v případě samostatných zařízení, které používají Linux uvnitř a mohou si jej jakkoli upravovat i bez souhlasu Tor-

valdse nebo jiných vývojářů jádra. Byli bychom možná překvapeni, kolik takových zařízení dnes na trhu najdeme: jsou to různé směrovače WiFi, datové projektory atd. V těchto případech často není ani snadné prověřit, v jaké podobě byl program chráněný GPL použit.

Softwarové knihovny a Menší GPL

Pro speciální účely vypracovala FSF jinou licenci, která se nazývá „Menší obecná veřejná licence GNU“ (LGPL, Lesser General Public License). V jejím původním názvu bylo místo „Lesser“ použito slovo „Library“, čímž bylo naznačeno předpokládané hlavní užití této licence pro ochranu softwarových knihoven. Změnou názvu chtěla FSF zdůraznit nejen to, že licence může být použita i pro jiné programy než knihovny, ale také – a možná hlavně – že softwarové knihovny mohou být v mnoha případech chráněny standardní GPL.

Stanovisko FSF vůči programům používajícím knihovny je v LGPL vyjádřeno zhruba takto: Program, který je napsán tak, že vůbec nevyužívá zdrojový text knihovny, není dílem odvozeným z knihovny, ale „dílem využívajícím knihovnu“. Proto se jej netýká (jakákoli) licence, s níž je šířena knihovna. Složitější (ale velmi častý) případ nastane tehdy, používá-li program hlavičkové soubory, které jsou součástí knihovny. V LGPL se uvádí, že program není považován za dílo odvozené z knihovny v případě, že se z hlavičkových souborů používají pouze číselné parametry, schémata a přístupové body datových struktur, malá makra a malé in-line funkce (dlouhé nejvýše deset řádků).

Jakmile se však program „spojí“ s knihovnou do binární formy, výsledek už je dílem odvozeným z knihovny. Je-li tedy knihovna chráněna GPL, lze výsledný binární program šířit pouze za předpokladu splnění podmínek GPL.

Licence LGPL je méně restriktivní v tom smyslu, že umožňuje spojit program s knihovnou a výsledek šířit pod jakoukoli licencí a třeba jen v binární podobě. Musí se ale přitom poskytnout (nebo nabídnout bezplatné poskytnutí) zdrojový kód knihovny, popřípadě použít vhodný mechanismus sdílené knihovny. V posledním případě se předpokládá, že si uživatel knihovnu nainstaluje nezávisle na programu.

Samozřejmě jakékoli úpravy vlastní knihovny již dědí její licenci, tedy LGPL. V textu LGPL je rovněž výslovně uvedeno, že autor modifikované verze může, bude-li chtít, změnit její licenci na GPL.

GPL verze 3

Zřejmě největším nedostatkem Obecné veřejné licence GNU (verze 2) je její citelné zastarávání. Text GPL je, až na drobné kosmetické úpravy, starý téměř patnáct let. Uvědomíme-li si, že v té době neexistoval web ani Linux, je docela obdivuhodné, že GPL si i po tak dlouhé době udržuje své výsadní místo.

Řešení některých nejpalčivějších problémů však nelze dále odkládat, a proto skupina expertů koordinovaných nadací FSF započala práci na přípravě nové verze GPL (zřejmě půjde o verzi 3). Zatím nebyl publikován žádný návrh jejího textu, podle dílčích náznaků některých osob, které se na přípravě podílejí, se ale zdá, že hlavní změny můžeme čekat v těchto oblastech:

- *Vztah k duševnímu vlastnictví a softwarovým patentům:* Jak jsme si už vysvětlili, softwarové patenty jsou dnes pro open source software hlavní hrozbou. Lze očekávat, že nová GPL zde zaujme daleko striktnější pozici.
- *Softwarové komponenty spolupracující po síti:* Tento typ programů a služeb se v poslední době rozvíjí velmi dynamicky, zejména na platformě WWW (Web Services aj.). Soudím, že zde bude proces hledání vhodné rovnováhy velmi bolestivý.
- *Trusted Computing*: Tento koncept předpokládá, že počítače příští generace budou mít hardwarovou pojistku proti spouštění „neautorizovaných“ programů. Autorizací se přitom rozumí nějaká forma digitálního podpisu, který by k programu připojila nějaká uznaná certifikační autorita. I když je tato myšlenka z bezpečnostního hlediska určitě zajímavá, pro volné šíření a modifikace svobodného softwaru by mohla mít poměrně neblahé následky.
- *Přizpůsobení GPL autorskému právu EU*: Toto by byla pro nás velmi významná změna, která by vyřešila zmiňované nesrovnalosti GPL ve vztahu k autorskému právu založenému na

bernské úmluvě. GPL by se pak mohla stát skutečně univerzální licencí svobodného softwaru.

První návrh textu GPL verze 3 se očekává ještě během roku 2005. Záměrem FSF je podrobit novou licenci před jejím přijetím velmi zevrubné diskusi za účasti všech zainteresovaných skupin včetně softwarového průmyslu. Výsledkem by pak měla být licence přijatelná pro daleko širší okruh producentů softwaru a také lépe slučitelná s jinými licencemi svobodného softwaru. Tím by se také mohl eventuálně zredukovat už poměrně nepřehledný a stále rostoucí počet licencí typu FOSS (Free or Open Source Software). Jisté je ovšem také to, že nová GPL rozhodně nepůjde proti přesvědčení Richarda Stallmana, a tak žádné průlomové změny nečekejme.

Místo závěru

V pěti pokračováních seriálu o svobodném a open source softwaru jsme se dotkli většiny hlavních zásad i sporných otázek tohoto stále významnějšího segmentu informační společnosti. Je dobře, že se v druhém pokračování dostalo i na hlas z opačného břehu, přece jen dialog je pro čtenáře jistě zajímavější, a že jsme se neomezili je na otázky licencí a autorských práv a zdůraznili potenciální nebezpečí softwarových patentů.

Nerad bych ale tímto seriál definitivně ukončil. Názory tří zúčastněných autorů jistě ani zdaleka tuto problematiku nevyčerpávají. Kupříkladu zcela chyběl hlas zastánců licencí typu BSD, které mají právě v brněnské akademické komunitě velmi silnou pozici. Nechme proto tomuto seriálu otevřený konec! Určitě bude zajímavé se k tématu po čase vrátit.

Literatura

[1] MATYSKA, L. Softwarové patenty. *Zpravodaj ÚVT MU* 15(4), Brno, 2005.

[2] RAMBOUSKOVÁ, L. Autorský zákon. Sborník semináře *SLT 2001*, Konvoj, Brno, 2001. `http://www.cstug.cz/slt/01/plne_texty/11.pdf`

[3] ROSEN, L. Derivative works. *Linux Journal* 103, January 2003, str. 96.

Obsah

Editor: M. Bartošek • Vydavatel: Masarykova univerzita v Brně • Žerotínovo náměstí 9, 601 77 Brno, CZ
tel.: 549 492 100 • e-mail: zpravodaj@muni.cz • WWW: http://www.ics.muni.cz/zpravodaj/
Sazba systémem LaTeX 2ε písmem LucidaBright • Tisk: Grafex Blansko • *Neprodejné* • ISSN: 1212–0901