Přístupová práva (nejen) v IS BAPS a jak na ně

Filip Procházka, Petr Glos, ÚVT MU

Ze zkušeností z provozu první verze IS BAPSu (viz [4]) vyplynulo, že použitý model přístupových práv je příliš hrubý a pro druhou verzi systému se jeví nedostatečný z více hledisek. Proto jsme začali hledat přístup či nástroj pro realizaci mechanizmu přístupových práv, který by vyhovoval zvýšeným nárokům. Hlavními požadavky bylo, aby:

- Byl dostatečně obecný a umožnil definovat přístupová práva s takovou podrobností, jaká bude v konkrétním případě potřeba.
  Jako u obdobných informačních systémů se ukazuje, že řada požadavků vyvstane až při využívání systému, a proto potřebujeme dostatečně obecný model přístupových práv. Tím se vyhneme obtížně řešitelné situaci, kdy realizace nově vznesených požadavků znamená často razantní zásah do fungujícího systému jak v datovém modelu, tak v aplikační logice.
- Účinně podpořil proces modelování přístupových práv.
  Jinými slovy řečeno, chceme aby se nástroj nestavěl k definici přístupových práv způsobem: *Někde bokem si to ujasni a přesně nadefinuj a až to budeš mít, tak to do mě zaznamenej a já pak podle toho budu pracovat.* Nástroj musí poskytovat prostředky, jak definovat přístupová práva rychle a kvalitně. Navíc musí být vytvořený model snadno udržovatelný a rychle modifikovatelný.

1 Jak dosáhnout obecnosti při specifikaci přístupových práv

Při zkoumání společného obecného jádra různých přístupů ke správě přístupových práv se ukázalo, že takové jádro skutečně existuje a lze popsat takto:

Cílem je vědět, KDO smí (nebo nesmí) CO s ČÍM KDY provádět. Musíme být schopni odpovídat na autorizační dotazy typu: „*Smí daný aktér (KDO) provést danou úlohu (CO) s daným objektem (S ČÍM) v daném čase (KDY)?*“, „*Se kterými objekty (S ČÍM) smí daný aktér (KDO) provádět danou úlohu (CO)*“ a podobně.

Sloveso „smí“ respektive „nesmí“ vyjadřuje tzv. MODALITU. Pro oblast definice přístupových práv nás zajímají především modality „smí“ a „nesmí“.

Jádrem modelu přístupových práv je tedy seznam pětic (KDO, MODALITA, CO, S ČÍM, KDY). Nad touto strukturou je pak definován příslušný proces pro vyhodnocování autorizačních dotazů.

Jednotlivé bezpečnostní modely se liší tím, jakým způsobem je možné vymezovat prvky jednotlivých dimenzí a svým vyhodnocovacím procesem. Uveďme příklady omezení jednotlivých modelů:

- vymezování dimenze KDO je vázáno na konkrétní uživatele a není možné použít pro vymezení role uživatelů
- při vymezování CO se používají pouze základní úlohy (operace) nad daty read, delete, update
- MODALITA je většinou *smí*, pak jde o politiku „co není povoleno je zakázáno“ a není možné jednoduše specifikovat dočasné zákazy
- při vymezování S ČÍM je možno používat pouze přímé ukázání nebo vymezení pomocí třídy objektů, ale už není možné vymezit S ČÍM pomocí vztahů k jiným objektům. Například nelze S ČÍM vymezit jako všechny konektory propojovacích kabelů v dané místnosti.
- KDY je možné vymezit pouze konkrétním časovým údajem, ale už ne rolemi časových údajů např. „po dobu trvání etapy zkušební provoz“

Z těchto příkladů plyne, že pokud chceme budovat obecný nástroj pro definici přístupových práv, musíme především disponovat obecným aparátem pro vymezování prvků jednotlivých dimenzí. Pro tento účel byly použity tzv. *identifikační výrazy* (zkratka idex – identification expression), které umožňují libovolně vymezovat (resp. dotazovat se na) objekty zájmu v modelem popsané oblasti. Identifikační výrazy tedy lze chápat jako dotazovací jazyk nad daty. Od dotazovacího jazyka SQL se liší v následujících bodech:

– Identifikační výrazy specifikují CO chci získat za informace, kdežto SQL příkazy vyjadřují JAK informace získat z databáze, jejíž fyzickou strukturu musíme znát (jak se jmenují příslušné tabulky, jak se jmenují atributy). Identifikační výrazy od fyzického uložení dat abstrahují. Vykonání identifikačního výrazu nad konkrétní databází je možné díky překladači, který na základě modelu dané databáze dokáže identifikační výrazy automaticky překládat na SQL příkazy. To už je však pro uživatele skryto.
– Díky abstrahování od fyzického uložení dat je dotazovací aparát identifikačních výrazů přístupnější pro běžného uživatele, pro kterého je jazyk SQL většinou „těžko stravitelný“.

Ukažme si příklad práva definovaného pomocí identifikačních výrazů:

```
právo (
 správce_lokality_1_patro,
 smí, editovat,
 zařízení_v_dané_místnosti (
   místnosti_v_lokalitě(1_patro)
   filtruj nepatřící do
   kategorie(spec_režim_správy)
  ),
 doba_rekonstrukce_1_patra
)
```

Tímto právem jsme specifikovali, že *správce prvního patra* (KDO) *smí* (MODALITA) *editovat* (CO) *zařízení nacházející se v místnostech prvního patra, která nejsou ve speciálním režimu správy* (S ČÍM) *po dobu rekonstrukce_prvního_patra* (KDY). Jednotlivé elementy použité v identifikačních výrazech (např. vztah *zařízení_v_dané_místnosti*, kategorie *zařízení_ve_speciálním_režimu_správy* pochází z modelu IS BAPS).

Prvky jednotlivých dimenzí práv vychází z analýzy a specifikace systému. KDO jsou role uživatelů, CO jsou úlohy, které aplikace umožňuje. Při objektově orientovaném návrhu jsou tyto informace zachyceny v případech užití (use-case). S ČÍM vychází z konceptuálního modelu, stavů a vlastností používaných objektů. V objektově orientovaném návrhu jsou tyto informace získány z modelů tříd a stavových diagramů.

2 Jak si usnadnit práci při specifikaci a údržbě přístupových práv

V situaci, kdy jsou přístupová práva složitější (což je případ IS BAPS), okamžitě vyvstává otázka, jak se v nadefinovaných právech orientovat a neztratit při údržbě práv přehled. Jedním ze způsobů je použít metody odvozování práv pomocí pravidel. Specifikujeme generickou množinu práv a poté pravidla, jak z ní odvodit práva další. Pomocí pravidel je možné odvozovat nová práva na základě jednotlivých dimenzí či jejich kombinací. My si zde pro ilustraci popíšeme tři typické druhy pravidel využívající dimenze KDO, CO a S ČÍM:

2.1 Odvozování na základě dimenze CO

První typickou situací je, že díky právu uživatele vykonávat úlohu A získává uživatel právo vykonávat i úlohu B, pokud B je využívána při vykonávání A. Příklad z IS BAPSu – pokud někdo může editovat lokality, může je i prohlížet.

Pravidlo, které na základě vztahů mezi úlohami „navyrábí“ z existujících práv práva nová, je zapsáno takto:

```
pro každé
 právo(P),
 úloha(U) patřící do
  úlohy_využívané_úlohou(co(P))
vykonej:
 definuj právo
  kdo(P) může provádět
  úlohu U
  s s_čím(P)
```

Klíčová slova jazyka pro specifikaci pravidel jsou podtržena. Příkaz co(P) vrací složku CO k právu P, analogicky příkaz kdo(P) vrací složku KDO od práva P.

Pravidlo je vyhodnocováno následujícím způsobem: prochází se postupně všechna práva, u nich je zjištěna složka CO a od ní všechny úlohy, které jsou úlohou ve složce CO využívány. Pro každou možnou instanciaci proměnných P a U je vykonáno to, co následuje za klíčovým slovem vykonej zde definice nového práva.

2.2 Odvozování na základě dimenze S ČÍM

Častá je také situace, kdy uživatel na základě toho, že získal právo manipulovat s nějakými objekty A, automaticky získává toto právo i na objekty B, které jsou s A v určité vazbě.

```
pro každé
 právo(P) patřící do
  práva_se_složkou
   (co = prohlížení_atributů),
 idex(S) patřící do
   s_čím
   (
     P
     filtruj
     výstup_idexu( lokalita )
   ),
vykonej:
 vytvoř idex
   I=místnosti_v_lokalitě(S),
 definuj právo
   kdo(P) může provádět
   úlohu prohlížení_atributů
   s I
```

Toto pravidlo říká, že pokud někdo může prohlížet atributy nějakých lokalit, přidělí se mu i právo na prohlížení atributů všech místností v těchto lokalitách. Za klíčovým slovem pro každé jsou vymezena taková práva, která mají ve složce CO úlohu prohlížení atributů a ve složce S ČÍM vymezují nějaké lokality.

Na pátém řádku vytváříme nový identifikační výraz I, který vznikne tak, že z toho, co je identifikováno identifikačním výrazem S (nějaká množina lokalit) navíc přejdeme od prvků této množiny po typu vztahu *místnosti_v_dané_lokalitě*.

2.3 Odvozování nového KDO a S ČÍM specializace rolí

Příklad z BAPSu: většina rolí, které se v přístupových právech vyskytují, je vázána na sítě (např. síť MU, síť VUT) – například role *správce sítě*. Správce sítě MU může provádět příslušné úlohy (CO) s příslušnými objekty (S ČÍM), které ale musí patřit do sítě MU. Správce sítě VUT může dělat to samé, příslušné objekty však musí patřit do sítě VUT. Práci si usnadníme zavedením obecné role *správce sítě* a k ní specifikujeme práva, která mohou správcové sítí dělat. Práva pro správce sítě omezená na určitou síť pak jsou odvozena pravidly (viz příklad níže). Pokud se poté nějak změní kompetence správců sítí, stačí provést úpravu na jednom místě – u definice práv pro obecnou roli *správce sítě* a aktualizaci kompetencí správců konkrétních sítí již zajistí automaticky příslušné pravidlo.

Tento případ vyřešíme pomocí dvou pravidel. První vytvoří specializací nové role pro sítě, které máme v systému zavedeny a druhé pravidlo těmto rolím přidělí příslušná práva:

```
pro každý objekt
 role(R) patřící do
  kategorie
   (specializovatelná_role_na_sítě),
 síť(T)
vykonej:
 vytvoř objekt typu role s vlastnostmi:
  patri do
   role_vytvořené_specializací_role(R),
  patri do
   role_omezené_na_práci_s_objekty_z(T)
```

Toto pravidlo pro každou roli, kterou lze specializovat na síť a pro každou síť vytvoří novou roli. Tuto roli spojí příslušnými typy vztahů s rolí, ze které je odvozena a se sítí na jejichž působnost je hraní role omezeno. Tyto vztahy pak jsou využity v následujícím pravidle:

```
pro každé
 právo(P), role(R) patřící do
  role_specializované_z_role(kdo(P)),
 síť(T) patřící do
  objekty_na_které_je_omezena_role(R)
vykonej:
 vytvoř idex
   I_new=s_čím(P) a prvky_sítě(T),
 definuj právo
   R může provádět úlohu co(P) s I_new
```

Toto pravidlo probírá všechna práva, od KDO najde všechny specializované role, od nich si zjistí sítě, na které jsou specializované, vezme S ČÍM tohoto práva a doplní podmínku patření prvku do sítě.

Vyhodnocování pravidel je řešeno principem pevného bodu: pravidla jsou umístěna do sady a postupně spouštěna jedno za druhým. Inference končí, pokud již nelze uplatnit žádné pravidlo ze sady – vyhodnocování je „nasycené“. Proto nezáleží na pořadí výše uvedených pravidel.

3 Jak správu přístupových práv zrealizovat

Potřebujeme podpořit dva základní módy práce s přístupovými právy:

- Specifikace práv (modelování) a
- zodpovídání autorizačních dotazů nad naspecifikovanými právy (uživání).

Je zřejmé, že pro modelování práv pomoci pravidel a používání aparátu identifikačních výrazů potřebujeme poměrně silný modelovací nástroj. V našem řešení je použit nástroj UIR (Universal Information Recorder). Jde o nástroj vyvíjený brněnskou firmou eTrium s.r.o. Lze ho zařadit do oblasti tzv. Business Rule Engines a nástrojích postavených na Model Based Architecture.

Literatura

[1] Castano, S., Fugini, M. G., Martella, G., Samarati, P., Database security, ACM Press, 1996

[2] Procházka, F.: eTrium-ISMS Univerzální nástroj pro správu přístupových práv, Sborník konference Datakon 2003

[3] Staníček Z., Procházka F. Technologie eTrium a její aplikace. Zvaná přednáška Datakon 2002

[4] Zpravodaj ÚVT MU, roč. XIII, č.4 – `http://www.ics.muni.cz/bulletin/issues/vol13num04` □