Webové služby. Identifikácia, autentizácia a autorizácia

Martin Stančík <stancik@fi.muni.cz>

Centrum výpočetní techniky
Fakulta informatiky
Masarykova univerzita

27. október 2010

Obsah

Úvod

- Všetko je na webe
- Záplava rôznych aplikácií a informácií
- Nechcem všetko zverejňovať celému svetu
- Nutnosť autentizácie a autorizácie
- Veľa aplikácií, veľa prihlasovacích údajov
- Snaha integrovať aplikácie a komunikovať cez viaceré systémy

Čo je to služba?

- Činnosť, ktorá uspokojí našu potrebu
- Jej výsledkom je určitý efekt a nie hmotný produkt
- Je nehmotná a ďalej nedeliteľná
- Služby v bežnom živote
  - predajca
  - zákazník
  - upratovacia služba, kaderníctvo, banka, polícia
- Webové služby
  - Rovnaký význam ako obyčajná služba
  - Chce ju použiť viacero úžívateľov
  - Dostupná takmer odkiaľkoľvek
  - Na svoju požiadavku chcem konkrétnu odpoveď

Motivácia

- Typická vlastnosť programátorov - lenivosť
- Niektoré časti sa chcú použiť na viacerých miestach
- Reakcia na zmenu správania
- Žiaden veľký systém kvôli drobnosti
- Podobný princíp ako http požiadavok
- Nezaujíma ma formát, ale obsah
- Aké sú kurzy mien, aké bude počasie?
- Web 2.0 `http://www.google.cz/ig`

Ako to funguje?
Účastníci komunikácie

Pomocou sieťových protokolov

- Poskytovateľ služby
- Klient
- Register služieb

- Publikovanie
- Vyhľadávanie
- Prepojenie

Ako to funguje?
Rozdelenie

- Veľké webové služby na princípe SOA
  - Sú v registri služieb
  - Výmena XML správ
  - SOAP, WSDL
- Vzdialené volanie služieb (RPC)
  - Vopred dohodnutý formát
  - XML-RPC
- Representational State Transfer (REST)
  - Orientovaný dátovo
  - CRUD
  - Rôzne reprezentácie zdrojov: XML, HTML, PDF

Servisne orientovaná architektúra SOA

- architektonický koncept, realizácia v podobe protokolu SOAP
- zasielanie správ
- vhodná na business prístup a zachytávanie procesov vo firme
- vytvorenie rozhrania nad aplikáciami
- zmena aplikácie neovplyvní funkčnosť systému
- flexibilita pri pridávaní nových služieb
- ESB - Enterprise service bus

Volanie vzdialených služieb RPC

- technologia umožňujúca vykonať vzdialené volanie procedúry
- nie je také volné ako SOA, orientované na metódy
- Postup:
  - Zabalenie parametrov do vhodnej formy na strane klienta (marshalling)
  - Odoslanie balíčka serveru
  - Rozbalenie parametrov (unmarshalling) na serveri
  - Spustenie samotnej metódy
  - Výsledok sa opätovne zabalí a pošle klientovi
  - Klient rozbalí výsledok a ďalej ho spracuje
- proces výpočtu je pre klienta tzv. čierna skrinka
- proces neprebieha u klienta, čo je výhoda i nevýhoda
- XML-RPC základ pre SOAP

Volanie vzdialených služieb RPC

Príklad zaslania požiadavku

```
POST /server HTTP/1.0
User-Agent: Mozilla/5.0
Host: xmlrpc.pocasi.cz
Content-Type: text/xml
Content-length: 314

 <?xml version="1.0"?>
 <methodCall>
  <methodName>Weather.temperature</methodName>
     <params>
       <param>
         <value><dateTime.iso8601>20101027T16:00:00</dateTime.iso8
       </param>
       <param>
         <value><string>Brno</string></value>
       </param>
   </params>
 </methodCall>
```

Volanie vzdialených služieb RPC

Príklad správy výsledku

```
HTTP/1.1 200 OK
Connection: close
Content-Length: 12
Content-Type: text/xml
Date: Ut, 27 October 2010 16:01:15 GMT
Server: xmlrpc.pocasi.cz

<?xml version="1.0"?>
<methodResponse>
 <params>
 <param>
 <value><string>7 C</string></value>
 </param>
 </params>
</methodResponse>
```

REST

- je dátovo zameraný, klient-server architektúra
- Zdroje majú jednoznačne definované URI
- 4 metódy CRUD, beží na protokole HTTP(GET/POST)
- umožňuje cache, je bezstavový(potreba zaslať všetky parametre)
- tzv. Query parametre pre filtrovanie výsledku
- možnosť použitia metód XML, JSON, RSS a ATOM
- `http://developers.facebook.com/docs/reference/rest/status.get`
- `https://api.facebook.com/method/status.get?uid=609152564&limit=1`

GET /statuses/user/timeline/uid.xml Host: twitter.com

Web 2.0

- webové služby v praxi
- vytváranie obsahu
- štítkovanie
- komentáre
- agregácia
- otvorenosť
- paranoia z možného sledovania užívateľa

RSS

- Really Simple Syndication
- nemožnosť sledovať všetko
- pravidelný odber obsahu (v praxi 5-10 min. po zverejnení)
- RSS vie čítať už takmer všetko (REST), dektopové i webové aplikácie
- `http://www.google.com/reader/`

Na webe nie je možné byť anonymný

- dnes sa nepracuje len z jedného miesta
- aplikácie uchovávajú citlivé údaje
- užívatelia chcú mať vopred prispôsobený obsah, napr. lunchtime.cz
- webový "desktop"
- ako server zistí kto som

Identifikácia

- neznáma entita sa stáva známou
- užívateľ sa voči autorite preukazuje svoju identitu, tá ju však neoveruje
- identifikačné preukazy, OP, pas, číslo zdravotného poistenia...
- na webe je to meno, prezývka, IP adresa, session

Autentizácia

- overenie identity, či sa jedná naozaj o ňu
- overí sa platnosť OP a držiteľ porovná s osobou, ktorá ju predkladá
- v počítačovom prostredí je možností overenia identity viacero
  - nejaká forma hesla (PIN, SMS kód, tajný kľúč)
  - čítačka kariet
  - biometrické údaje
  - čo by identita mala vedieť, captcha, fotky priateľov, diplom???

Autorizácia

- overenie, či identita má oprávnenie uskutočniť nejakú akciu
- ak je človeku menej ako 18 rokov nemožno mu predať alkohol
- študent nemá oprávnenie si zadať známku do svojho predmetu
- aplikácie majú rôzne stupne ochrany, a užívatelia rôzne oprávnenia

Zhrnutie

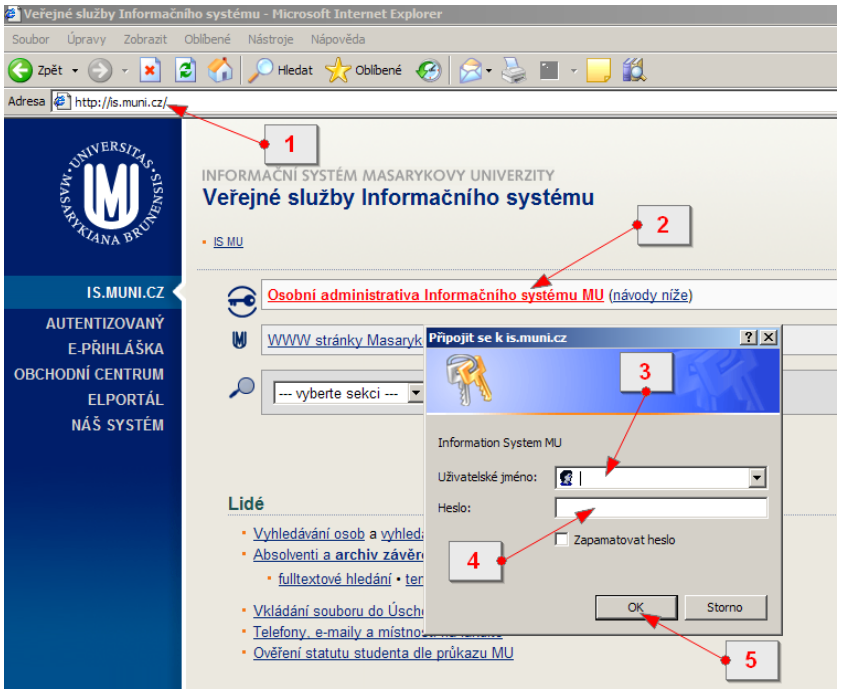
- keď chcem používať aplikáciu musím mať na ňu patričné oprávnenie
- oprávnenie nie je trvalé a môže sa časom meniť
- na niektoré úkony potrebujem viacúrovňový stupeň ochrany
- autentizácia sa overuje na základe dát od užívateľa
- autorizáciu zväčša overuje samotná služba, aplikácia

Session autentizácia

- server si uloží poznatok, že je užívateľ prihlasený do cookie
- server následne už len kontroluje platnosť cookie
- každá cookie má nejaký timeout, po ktorej sa treba znova prihlásiť

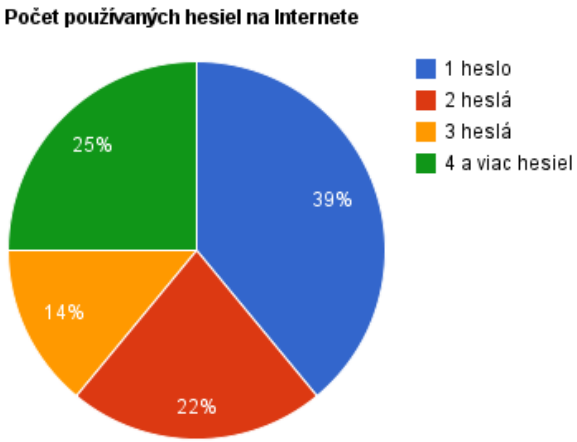
Basic autentizácia

- vyskakujúce okno pri prístupe na chránenú oblasť
- server si uchováva úspešnú autentizáciu a pri ďalšom prístupe nevyžaduje prihlásenie



Moderné spôsoby autentizácie
Motivácia

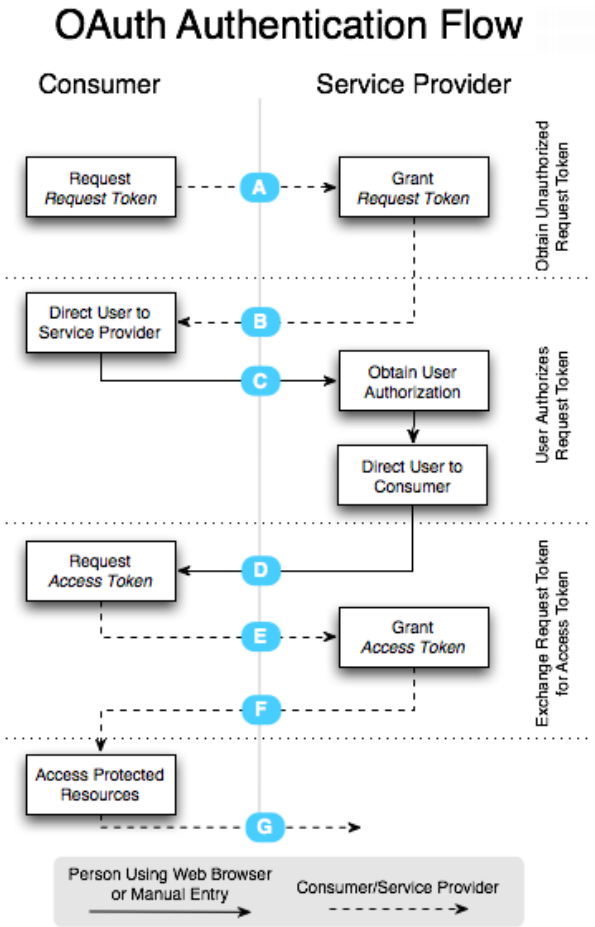
- veľa používaných aplikácii
- veľa identít(užívateľských mien) i hesiel
- nemožnosť si pamätať hesla vedie k ich "ukladaniu"
- obmedzenie aplikácie na konkrétnu osobu je takmer nemožné, lebo 1 osoba má viac mailov



OAuth

- protokol pre bezpečné zabezpečenie API autentizace
- citlivé prihlasovacie údaje sa predávajú len medzi užívateľom a poskytovateľom služby (SP)
- tretie strany pristupujú k dátam užívateľa bez vedomia hesla
- aplikácia si vyžiada od poskytvoateľa (service providera) dočasný token a zašle ho s užívatelom sa k SP prihlásiť
- po prihlásení u SP je užívateľ presmerovaný späť k danej aplikácií
- aplikácia už môže pristupovať k užívateľským dátam na základe API poskytovateľa

OAuth
Ako protokol funguje



OpenID

- užívateľ je registrovaný u tzv. poskytovateľa identit (IdP)
- univerzálny -¿ akákoľvek služba požadujúca prihlásenie
- otvorený -¿ zadarmo plne dostupná špecifikácia
- decentralizovaný -¿ možnosť registrácie u viacerých IdP
- len overenie užívateľa a poskytnutie schválenej informácie o ňom
- užívateľ zadá ako prihlasovací údaj svoje OpenID, na základe ktorého je presmerovaný na svojho IdP s požiadavkou prihlásenia a poskytnutia informácií o užívateľovi
- po prihlasení u IdP je opäť presmerovaný k poskytovateľovi služieb SP

Rozdiely medzi OpenID vs. OAuth

OpenID

- definuje protokol, výmenu a rozsah informacií
- distribuovaný systém, nezaleží na poskytovateľovi identít
- dôraz je na užívateľovi
- overuje užívateľa a poskytuje o ňom informácie

OAuth

- definuje len protokol, výmena informacií je už cez API poskytovateľa
- autentizácia len voči konkrétnej službe pre prístup k API
- dôraz na službe
- overuje užívateľa a pristupuje cez API v jeho mene

Shibboleth

- Single Sign-On (SSO)
- funguje na podobnom princípe ako OpenID
- projekt konzorcia Internet2
- poskytuje overené údaje, väčšinou z akademického prostredia
- poskytovatelia identít (IdP) i služieb sú často v národných federáciach

Shibboleth

- užívateľ, ktorý chce pristupovať k chránenej časti služby u SP je presmerovaný k svojmu domovskému IdP a po prihlásení je opäť presmerovaný k SP s už požadovanými informáciami o užívateľovi
- SP nemusí vedieť ku ktorému IdP užívateľa presmerovať, preto sa môže obrátiť na službu WAYF (Where Are You from), ktorá udržuje všetkých poskytovateľov identít, z ktorých si užívateľ môže vybrať
- v ČR federáciu eduID.cz prevádzkuje Cesnet. V ostrej prevádzke je od 1. 1. 2009
- `https://odevzdej.cz/shibboleth/`
- `http://www.lupa.cz/clanky/shibboleth/`

mojeID

- nový poskytovateľ OpenId v ČR od 26. 10. 2010
- overovanie užívateľov: e-mail, SMS, klasická pošta
- dokonca fyzické overovanie pri citlivejších dátach
- užívateľ môže určit, ktoré dáta o sebe predá ďalej
- spôsob prihlásenia môže určit aj konkrétna služba
- `http://www.mojeid.cz/`

Koniec