

WebDAV

Dokument k prezentaci na předmět X36MTI



Filip Jandejsek

29. listopadu 2007

Obsah

1.	Úvod.....	3
2.	Vlastnosti protokolu.....	3
3.	Softwarová podpora protokolu.....	3
3.1.	Klientská strana.....	3
3.2.	Serverová strana.....	4
3.3.	Rozšíření a odnože protokolu WebDAV.....	4
4.	Komunikace.....	5
4.1.	HTTP.....	5
4.2.	XML.....	5
4.3.	Příklad komunikace protokolu WebDAV.....	6
5.	Zámky.....	7
5.1.	Popis použití zámků.....	7
5.2.	Příklad práce se zámký.....	8
6.	Kolekce.....	9
6.1.	Popis kolekcí.....	9
6.2.	Příklad práce s kolekce.....	10
7.	Závěr.....	10
8.	Zdroje.....	11

Seznam příkladů

Příklad 1 – Aktualizace vlastností souboru.....	6
Příklad 2 – Použití příkazu LOCK.....	8
Příklad 3 – Smazání uzamčeného souboru bez uvedení klíče.....	9
Příklad 4 – Vytvoření nové kolekce.....	10
Příklad 5 – Kopie souboru.....	10
Příklad 6 – Přesun souboru.....	10
Příklad 7 – Smazání souboru.....	10

1. Úvod

Protokol WebDAV byl navržen jako protokol pro podporu vzdálené správy souborů a víceuživatelské práce na dokumentech uložených na vzdálených serverech.

WebDAV začal vznikat v roce 1996. Impulsem pro zahájení prací na novém protokolu byla dvě setkání vedená Jimem Whiteheadem pod záštitou organizace W3C o problémech distribuovaného autorství na webu. Aby bylo možné nejen prohlížet obsah webu, ale i jej editovat, bylo třeba rozšířit možnosti protokolu HTTP. Protokolem HTTP se právě zabývala organizace IETF (*Internet Engineering Task Force*), a tak bylo rozhodnuto o vytvoření pracovní skupiny v rámci IETF. V průběhu prací došlo k oddělení podpory verzování souborů, které bylo později přidáno jako rozšíření. Výsledkem návrhu byl standard pro vzdálenou správu souborů z února 1999 popsáný v RFC 2518.

V červnu roku 2007 bylo vydáno RFC 4918, které přináší drobné změny a aktualizace ke standardu z roku 1999.

2. Vlastnosti protokolu

Protokol WebDAV umožňuje vytvářet, odstraňovat, kopírovat a přesouvat na vzdáleném serveru

- dokumenty a soubory obecně
- informace o dokumentech (např. autor nebo datum vzniku)
- kolekce dokumentů
- kolekce vlastností souborů a dokumentů

Protokol WebDAV umožňuje také dotazování se na tyto soubory, informace, vlastnosti a kolekce. Pro podporu spolupráce uživatelů pro editaci více souborů poskytuje WebDAV možnost uzamknout editovaný zdroj. Zámky jsou dvou typů, výlučné (*exclusive*) a sdílené (*shared*). Více informací o zámcích naleznete v kapitole Zámky.

Kolekci si lze představit jako obdobu adresáře, kolekce má stejně jako adresář stromovou strukturu. Od typického adresáře se však kolekce odlišuje možností umístění částí kolekce na různých místech file-systému nebo dokonce na různých fyzických strojích.

Standard protokolu definuje několik druhů informací, které lze k dokumentu (souboru) připojit. Lze však také vytvářet a připojovat vlastní druhy informací.

3. Softwarová podpora protokolu

V dnešní době je použití protokolu WebDAV podporováno širokým spektrem aplikací a operačních systémů.

3.1. Klientská strana

Ve světě operačních systémů společnosti Microsoft existuje celá řada aplikací, Microsoft však ve svých operačních systémech přímo nabízí zabudovanou podporu použití protokolu WebDAV a to od verze Windows 2000 a novější. Pro Windows 98 nabízel

Microsoft rozšíření pro Windows Explorer nazvané *Web Folders*. Toto rozšíření již bylo přítomno ve Windows 2000. Od verze Windows XP je WebDAV podporován pomocí nového klienta *WebDAV mini-redirector*, který pracuje na úrovni file-systému, dovoluje připojit WebDAV zdroj jako síťovou jednotku a tak zprostředkovat její využití i dalším aplikacím. Bohužel všechny verze redirectoru obsahují závažné chyby např. při HTTP autentifikaci nebo při využití HTTPS jako komunikačního média.

Pro linuxové operační systémy je k dispozici plugin *davfs2*, který dovoluje namountovat WebDAVový sdílený prostor do linuxového file-systému. Tento plugin je modernizací svého předchůdce *davfs*.

Samozřejmě je možné pro výše zmíněné platformy Microsoftu i linuxu využít i některou z dostupných aplikací, jako například:

- SkunkDAV
- DAVManager
- jEdit WebDAV plugin
- Cadaver
- Jakarta Slide Project

3.2. Serverová strana

V současné době již WebDAV podporuje drtivá většina významných serverů. Jako příklad lze uvést:

- Internet Information Server
- Apache
- Apache Tomcat
- PyDAV
- Jigsaw
- WebDAV Project

3.3. Rozšíření a odnože protokolu WebDAV

Jak již bylo řečeno v úvodu, při vývoji protokolu se oddělila větev zabývající se podporou správy verzí. Produktem skupiny *Web Versioning and Configuration Management working group* bylo rozšíření nazvané *Delta-V* pro podporu správy verzí vzdálených dokumentů.

Skupina *DAV Searching and Locating working group (DASL)* se zabývala návrhem standardu pro vyhledávání a lokalizování ve službách poskytovaných pomocí WebDAV. Tato skupina však nikdy nevytvořila žádný oficiální standard, ale podle jejího posledního návrhu vzniklo několik různých implementací.

Protokol *CalDAV* je protokol dovolující přístup ke kalendáři a jeho funkcím prostřednictvím WebDAV. Události kalendáře jsou pojaty jako HTTP zdroje ve formátu *iCalendar*. Tyto události jsou pak poskládány do WebDAV kolekce a vytváří tak kalendář.

GroupDAV je varianta WebDAVu pro skupinové systémy (*groupware*), který je velmi podobný *CalDAVu*, ale je jednodušší. Dovoluje například ukládat a stahovat kalendářové a kontaktní záznamy.

4. Komunikace

Protokol WebDAV je postaven nad dvěma známými technologiemi, HTTP a XML. Komunikace je vždy složená ze žádosti klienta o provedení nějaké činnosti a z odpovědi serveru týkající se žádosti.

4.1. HTTP

Protokol HTTP je používán k samotnému zajištění způsobu komunikace. WebDAV je postaven nad protokolem HTTP 1.1 a využívá jeho metody

- GET – používá se k obdržení dokumentu (stránky) ze serveru
- POST – jako GET, za hlavičkami lze odeslat data
- HEAD – jako GET, ale vrátí pouze hlavičky (ne tělo)
- OPTIONS – používá se k dotazu na možnosti serveru
- PUT – jako GET, uchovává tělo požadavku na místě daném URL
- DELETE – odstraňuje dokument ze serveru
- TRACE – používá se ke sledování cesty požadavku

Aby bylo možné s dokumenty plně manipulovat, přidává WebDAV k příkazům HTTP 1.1 ještě další sadu příkazů

- PROPFIND – používá se k získání vlastnosti (hodnoty nebo názvu) vzdáleného dokumentu
- PROPPATCH – slouží k připojení nové, změně nebo odebrání stávající vlastnosti
- MKCOL – používá se k vytvoření nové kolekce
- COPY – používá se ke kopírování kolekcí a souborů
- MOVE – slouží k přesunu kolekcí a souborů
- LOCK – slouží k uzamčení souboru nebo kolekce
- UNLOCK – používá se k odstranění zámku uzamčené kolekce nebo souboru

K identifikaci úspěchu popř. neúspěchu požadavku klienta k provedení nějaké operace na serveru se využívají stavové kódy HTTP převážně řady 2xx a 4xx (200 – OK, 207 – multistatus, 204 – no content, 404 – not found apod.).

4.2. XML

Technologii XML využívá WebDAV jako nositele strukturovaných informací dotazu / požadavku klienta i odpovědi serveru. V XML jsou odesílány parametry příkazů, informace o souborech a kolekcích i výsledky zpracování příkazů. Struktura přenášených informací je definována v hlavním DTD pojmenovaném *DAV:*, kde jsou uvedeny potřebné

elementy a atributy pro základní použití WebDAVu. Pro definování vlastních atributů je možné využít jmenné prostory definované ve vlastním DTD. WebDAV však nekontroluje ani existenci určeného DTD ani správnost použití definovaných elementů. Přenášený XML dokument musí být správně strukturovaný (*well formed*), jinak jej server odmítne zpracovat a vrátí HTTP stavový kód 400 *Bad Request*.

4.3. Příklad komunikace protokolu WebDAV

Tento příklad znázorňuje požadavek na vytvoření nebo změnu vlastností *Autor* a *Popis* a smazání vlastnosti *Archivni* dokumentu *dokument.pdf* umístěném na serveru *www.webdavserver.cz*:

Příklad 1 – Aktualizace vlastností souboru

```
PROPPATCH /dokument.pdf HTTP/1.1
Host: www.webdavserver.cz
Content-Type: text/xml; charset="iso-8859-1"
Content-Length: xxxx

<?xml version="1.0" encoding="iso-8859-1" ?>
<D:propertyupdate xmlns:D="DAV:"
xmlns:Z="http://www.webdavserver.cz/dtd/dtd-dok">
  <D:set>
    <D:prop>
      <Z:Author>Bc. Filip Jandejsek</Z:Author>
    </D:prop>
    <D:prop>
      <Z:Popis>Muj dokument na WebDAV serveru</Z:Popis>
    </D:prop>
  </D:set>
  <D:remove>
    <D:prop>
      <Z:Archivni/>
    </D:prop>
  </D:remove>
</D:propertyupdate>
```

Pro nastavení nebo aktualizaci vlastností je použit WebDAV příkaz *PROPPATCH*, za ním následuje identifikátor souboru (nebo kolekce). V hlavičce *Host* je uvedeno WebDAV úložiště. Přenášený obsah je samozřejmě *text/xml*. Na příkladu je vidět použití vlastního jmenného prostoru označeného jako *Z* s DTD umístěném na *http://www.webdavserver.cz/dtd/dtd-dok*. Protože používáme *PROPPATCH*, jsou všechny informace o nastavovaných vlastnostech uvnitř značek *<DAV:propertyupdate>* a *</DAV:propertyupdate>*. Použití jiného HTTP příkazu samozřejmě vyžaduje použití jiného vhodného kořenového elementu. Nastavované vlastnosti jsou uzavřeny ve značkách *<DAV:set>* a *</DAV:set>*. Označení uživatelsky definovaných vlastností je uvnitř značek *<DAV:prop>* a *</DAV:prop>*. Odstraňované vlastnosti jsou uzavřeny uvnitř značek *<DAV:remove>* a *</DAV:remove>*.

Odpověď serveru:

```
HTTP/1.1 207 Multi-Status
Content-Type: text/xml; charset="utf-8"
Content-Length: xxxx

<?xml version="1.0" encoding="utf-8" ?>
<D:multistatus xmlns:D="DAV:" xmlns:Z="http://www.webdavserver.cz/dtd/dtd-
dok">
```

```

<D:response>
  <D:href>http://www.webdavserver.cz/dokument.pdf </D:href>
  <D:propstat>
    <D:prop><Z:Autor/></D:prop>
    <D:status>HTTP/1.1 200 OK</D:status>
  </D:propstat>
  <D:propstat>
    <D:prop>
      <Z:Popis/>
    </D:prop>
    <D:status>HTTP/1.1 200 OK</D:status>
  </D:propstat>
  <D:propstat>
    <D:prop><Z:Archivni/></D:prop>
    <D:status>HTTP/1.1 200 OK</D:status>
  </D:propstat>
</D:response>
</D:multistatus>

```

Protože jsme v rámci jednoho požadavku nastavovali více vlastností, je HTTP návratový kód volání serveru *207 Multistatus*. S tím také souvisí kořenová značka xml dokumentu (pokud nepočítáme značku `<?xml ...>`) `<DAV:multistatus>`. Uvnitř jsou odpovědi na jednotlivé požadavky (značka `<DAV:response>`) a identifikace souboru nebo kolekce, jehož se volání klienta týkalo (značka `<DAV:href>`). Návratové stavy jednotlivých požadavků jsou uzavřeny uvnitř `<D:propstat>`, jedná se o označení vlastnosti uvnitř `<D:prop>` a HTTP status uvnitř `<D:status>`.

5. Zámky

Pro podporu kolektivní práce nabízí WebDAV podporu zámků pro uzamčení editovaného obsahu. Zámky jsou typu *write-lock*, neboli zámky pro uzamčení možnosti zápisu.

5.1. Popis použití zámků

Zámky jsou dvou druhů. Prvním druhem je sdílený zámek (*Shared Lock*), který umožňuje uzamčení zdroje pro více uživatelů. Zde se předpokládá, že existuje mezi uživateli domluva, kdy nebude docházet současně práci více uživatelů na stejném úseku dokumentu, tedy že jeden uživatel nebude chtít přepsat tu část dokumentu, kterou právě přepisuje jiný.

Druhým typem zámků je výlučný zámek (*Exclusive Lock*). Ten uzamkne zdroj pro zápis pouze pro jednoho uživatele, ostatní smějí zdroj pouze číst.

Pokud je vyslán příkaz *LOCK*, tak WebDAV server vygeneruje klíč, *Lock Token*. Ten je vrácen v odpovědi na příkaz *LOCK*. Uživatel, který chce k uzamčenému zdroji přistupovat, se musí prokázat platným *Lock Tokenem*. Pokud je zámek sdílený, je pro každého uživatele, který zdroj uzamkne sdíleným zámkem, vygenerován vlastní klíč. Klíč je možné zjistit v rámci dotazu na vlastnost *lockdiscovery*. Zde je nutné říci, že klíče sami o sobě ještě nezaručují nějakou bezpečnost, protože je možné odeslat dotaz *PROPFIND* na *locktoken*. Autentifikaci a oprávnění k přístupu k uzamčenému zdroji totiž WebDAV přenechává na serveru, potažmo na autentifikačních mechanismech protokolu HTTP (například použitím hlavičky *Authorization* – více viz. RFC 2616 k protokolu HTTP).

Aby nedocházelo k zablokování při uzamčení a zapomenutí uživatele odeslat na zdroj příkaz *UNLOCK*, tak se při odesílání příkazu *LOCK* odesílá hlavička *Timeout*, která určuje čas, po jehož uplynutí vyprší platnost zámku.

Protože nemusí všechny servery implementující WebDAV zámky podporovat, připouští norma dva typy implementace – jednoduchou, kdy zámky nejsou podporovány, a úplnou, kde jsou výše zmíněné zámky obsaženy. Zda přístupovaný server obsahuje podporu zámků lze zjistit přečtením vlastnosti *DAV:supportedlock* nebo odesláním HTTP příkazu *OPTIONS*. Většina serverů podporujících WebDAV dnes splňuje úplnou specifikaci mechanismu zámků.

5.2. Příklad práce se zámky

Tento příklad ilustruje použití příkazu *LOCK* v žádosti klienta o uzamčení zdroje výlučným zámekem:

Příklad 2 – Použití příkazu LOCK

```
LOCK /dokument.pdf HTTP/1.1
Host: www.webdavserver.cz
Timeout: Infinite, Second-86400
Content-Type: text/xml; charset="iso-8859-1"
Content-Length: xxxx
Authorization: ...
```

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<D:lockinfo xmlns:D="DAV:">
  <D:lockscope><D:exclusive/></D:lockscope>
  <D:locktype><D:write/></D:locktype>
  <D:owner>
    <D:href>http://www.webdavserver.cz/users/jandef1/kontakt.html</D:href>
  </D:owner>
</D:lockinfo>
```

Odpověď serveru na žádost o uzamknutí zdroje:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset="utf-8"
Content-Length: xxxx
```

```
<?xml version="1.0" encoding="utf-8" ?>
<D:prop xmlns:D="DAV:">
  <D:lockdiscovery>
    <D:activelock>
      <D:locktype><D:write/></D:locktype>
      <D:lockscope><D:exclusive/></D:lockscope>
      <D:depth>Infinity</D:depth>
      <D:owner>
        <D:href>http://www.webdavserver.cz/users/jandef1/kontakt.html</D:href>
      </D:owner>
      <D:timeout>Second-3600</D:timeout>
      <D:locktoken>
        <D:href>opaquelocktoken:e71d4fae-5dec-22d6-fea5-00a0c91e6be4</D:href>
      </D:locktoken>
    </D:activelock>
  </D:lockdiscovery>
</D:prop>
```

Následující příklad zobrazuje pokud klienta smazat uzamčenou kolekci bez uvedení *locktokenu*:

Příklad 3 – Smazání uzamčeného souboru bez uvedení klíče

```
DELETE /locked/dokumenty HTTP/1.1
Host: www.webdavserver.cz
```

Odpověď serveru:

```
HTTP/1.1 423 Locked
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx
```

```
<?xml version="1.0" encoding="utf-8" ?>
<D:error xmlns:D="DAV:">
  <D:lock-token-submitted>
    <D:href>/locked/</D:href>
  </D:lock-token-submitted>
</D:error>
```

6. Kolekce

Kolekce jsou obdobou adresářů, mohou obsahovat jiné kolekce, soubory nebo vlastnosti. Jsou jakýmsi zdrojem, který se skládá ze seznamu interních členů a souboru vlastností.

6.1. Popis kolekcí

Jak bylo uvedeno v kapitole Vlastnosti protokolu, prvky kolekce (se stejným URI) nemusí být umístěny fyzicky na stejném místě. K manipulaci s kolekcemi slouží příkazy

- MKCOL – slouží vytvoření nové kolekce
- COPY – slouží ke kopii kolekce
 - hlavičky
 - Depth – určuje hloubku kopírování
 - 0 – jen tento zdroj
 - 1 – tento zdroj a jeho přímé potomky
 - Infinity – zdroj i všechny potomky
 - Overwrite – zda přepsat cílovou kolekci při kopírování
 - T – přepsat
 - F – nepřepisovat
 - Destination (URI) – cíl kopírování
- MOVE – slouží k přesunu kolekce, parametry jako COPY
- DELETE – smazání kolekce

6.2. Příklad práce s kolekcemi

Tento příklad zobrazuje požadavek klienta na vytvoření nové kolekce:

Příklad 4 – Vytvoření nové kolekce

```
MKCOL /novy_adr/ HTTP/1.1
Host: www.webdavserver.cz
```

Odpověď serveru:

```
HTTP/1.1 201 Created
```

V následujícím příkladu chce klient zkopírovat soubor *dokument.pdf* do nově vzniklé kolekce:

Příklad 5 – Kopie souboru

```
COPY /dokument.pdf HTTP/1.1
Host: www.webdavserver.cz
Destination: http://www.webdavserver.cz/novy_adr/dokument.pdf
Depth: infinity
```

Odpověď serveru:

```
HTTP/1.1 204 No Content
```

HTTP kódy řady 2xx oznamují úspěšné provedení operace, kopie souboru se tedy zdařila.

Další příklad zobrazuje příkaz k přesunu souboru:

Příklad 6 – Přesun souboru

```
MOVE /dokument.pdf HTTP/1.1
Host: www.webdavserver.cz
Destination: http://www.webdavserver.cz/smlouvy/dokument.pdf
```

Odpověď serveru:

```
HTTP/1.1 201 Created
Location: http://www.webdavserver.cz/smlouvy/dokument.pdf
```

Poslední příklad znázorňuje žádost klienta o smazání souboru:

Příklad 7 – Smazání souboru

```
DELETE /novy_adr/dokument.pdf HTTP/1.1
Host: www.webdavserver.cz
```

Odpověď serveru:

```
HTTP/1.1 204 No Content
```

7. Závěr

Protokol WebDAV je vhodný pro správu dokumentů na vzdáleném serveru a v jistém smyslu může nahrazovat např. FTP přístup. Jako příklad lze uvést použití WebDAVu na serveru *service.felk.cvut.cz*, kde se využívá pro odevzdávání semestrálních prací.

V dnešní době však WebDAVu vyrostla silná konkurence ve formě nejrůznějších *Content Management Systémů (CMS)*, které nabízí podobnou funkcionalitu. Stále nelze opomenout protokol FTP a jeho deriváty. Podpora správy verzí a uzamykání je také součástí

systemů jako *CVS* a *Subversion*. Možnost spolupráce při tvorbě dokumentů s využitím protokolu HTTP také podporují stále více rozmáhající se systémy Wiki.

Vzhledem k velkému množství konkurentů nelze říci, jaká bude budoucnost protokolu WebDAV. Že se tento protokol zatím v záplavě konkurentů neztratil, o tom svědčí i letos vydané RFC 4918 aktualizující původní popis protokolu.

8. Zdroje

- www.webdav.org
- [interval.cz \(interval.cz/clanky/zaklinadlo-jmenem-webdav\)](http://interval.cz/clanky/zaklinadlo-jmenem-webdav)
- RFC 2518 (tools.ietf.org/html/rfc2518)
- RFC 4918 (tools.ietf.org/html/rfc4918)
- RFC 2616 (tools.ietf.org/html/rfc2616)
- en.wikipedia.org/wiki/WebDAV