



Протоколът WebDAV:  
инсталиране, конфигуриране и различни  
реализации

Версия *Revision* : 80

Иван Иванов - rambius@users.sf.net,  
Петър Тахчиев - paranoiabla@gmail.com

3 юли 2006 г.

[У дома](#)

[Print](#)

[Заглавие](#)

[Съдържание](#)



Стр. 1 от 21

[Назад](#)

[Екран](#)

[Затвори](#)

[Край](#)

[У дома](#)

[Print](#)

[Заглавие](#)

[Съдържание](#)

[«](#) [»](#)

[«](#) [»](#)

Стр. 2 от 21

[Назад](#)

[Екран](#)

[Затвори](#)

[Край](#)

## Съдържание

1	Що е WebDAV?	3
2	Решенията на WebDAV	6
3	WebDAV демонстрация	11
4	WebDAV сървъри и клиенти	14
5	Сигурност при WebDAV	17
6	Заключителни думи	19
7	Достъп до настоящата презентация	21

[У дома](#)[Print](#)[Заглавие](#)[Съдържание](#)[◀](#) [▶](#)[◀](#) [▶](#)[Стр. 3 от 21](#)[Назад](#)[Екран](#)[Затвори](#)[Край](#)

# 1. Що е WebDAV?

**Web-based Distributed Authoring and Versioning** е разширение на HTTP протокола, дефинирано от RFC 2518.

- **Authoring** - процесът на създаване, осъвременяване и управление на съдържанието на един Уеб сървър;
- **Distributed** - даден документ може да има няколко автора, които са в различни географски райони;
- **Web-based** - съдържанието се управлява със същите средства, с които се и разглежда;
- **Versioning** - различни версии на един и същи ресурс могат да се достъпят; изключено е от RFC 2518, поради сложността си;

[У дома](#)[Print](#)[Заглавие](#)[Съдържание](#)[◀](#) [▶](#)[◀](#) [▶](#)[Стр. 4 от 21](#)[Назад](#)[Екран](#)[Затвори](#)[Край](#)

## 1.1. Начини за управление на Уеб ресурси

Начините да бъде променян web ресурс без WebDAV са били няколко.

- HTTP методите PUT и DELETE
- FTP и SSH, NFS дялове или споделени директории
- Нестандартизирани разширения на HTTP (Microsoft FrontPage, Netscape Navigator Gold, ...)

[У дома](#)[Print](#)[Заглавие](#)[Съдържание](#)[◀](#) [▶](#)[◀](#) [▶](#)[Стр. 5 от 21](#)[Назад](#)[Екран](#)[Затвори](#)[Край](#)

Проблеми свързани с тези методи:

- Потребителя трябва да е наясно с начина, по който се свързват йерархиите на сайта с файловата система на сървъра (FTP, SSH);
- Няма как да се създаде директория, премести, копира, или преименува даден ресурс (HTTP PUT и DELETE);
- Няма възможност за асоциирането на метаданни със самия ресурс;
- FrontPage праща командите си и данните си чрез POST заявки, а те могат да бъдат забранени по причини за сигурност;
- Проблемът с изгубената редакция важи за всички.

## 2. Решенията на WebDAV

### 2.1. Променени HTTP методи

**DELETE** - разширен да изтрива не само единични ресурси, а и колекции/ директории;

**PUT** - добавена е възможността да следи дали даден ресурс е заключен;

**GET** - променен е, за да скрива от обикновените браузъри, че даден адрес поддържа WebDAV;

**OPTIONS** - ако даден ресурс поддържа WebDAV, OPTIONS методът описва класа му на съвместимост с RFC 2518;

## 2.2. Нови методи

**MKCOL** - създава колекция/директория;

**MOVE** - премества даден ресурс от даден източник на дадено местоположение, което може е да на друг WebDAV сървър

**COPY** - копира даден ресурс от даден източник на дадено местоположение, което може е да на друг WebDAV сървър

**PROPFIND** - извлича метаданни (property) от даден ресурс

**PROPPATCH** - задаване на стойност на дадено пропърти.

**LOCK** - заключва ресурс

**UNLOCK** - отключва ресурс

[У дома](#)[Print](#)[Заглавие](#)[Съдържание](#)[◀](#)[▶](#)[◀](#)[▶](#)[Стр. 7 от 21](#)[Назад](#)[Екран](#)[Затвори](#)[Край](#)

[У дома](#)[Print](#)[Заглавие](#)[Съдържание](#)[◀◀](#) [▶▶](#)[◀](#) [▶](#)

Стр. 8 от 21

[Назад](#)[Екран](#)[Затвори](#)[Край](#)

## 2.3. Пропъртитата за даден ресурс: какво представляват и как се използват?

- Данни за самите данни (датата на последна промяна, типа на ресурса и т.н) от тип ключ със стойност.
- Биват "пасивни" или "активни".
- Активните (contentLength, lastModified) се контролират от самия сървър.
- Част от активните са единствено за четене, докато други могат да бъдат променяни от потребителите.
- Пасивните се контролират изцяло от клиента - сървърът е отговорен единствено за тяхното съхранение.
- Стойностите им могат да бъдат в **xml** формат.

## 2.4. Нови кодове

Всички HTTP кодове запазват значението си и тук.

**102 Processing** - Ако за изпълнението на една заявка са необходими повече от 20 секунди, сървърът връща този статус код на клиента

**207 Multistatus** - Отговорът съдържа повече от един кода за състояния

**422 Unprocessable Entity** - Заявката не може да бъде обработена от сървъра - тялото на заявката е валиден XML документ, но смислово не е верен

**423 Locked** - Ресурсът е заключен, и заявката не може да бъде осъществена.

**424 Failed Dependencies** -Необходимите изисквания не са налице

**507 Insufficient Storage** - Няма достатъчно дисково пространство

## 2.5. Нови хедъри въведени от WebDAV

**DAV** - Връща се в отговор на OPTIONS заявка и съдържа класовете на съвместимост на WebDAV ресурса с RFC 2518;

**Depth** - Указва дълбочината, с която трябва да се прилагат методи като: COPY, MOVE, ...

**Timeout** - Указва след колко време един заключен ресурс трябва автоматично да се отключи;

**Lock-Token** - Съдържа символа, който ни позволява да модифицираме заключен ресурс

**Overwrite** - Указва дали даден ресурси да бъдат презаписани или не; използва се в COPY и MOVE заявки;

**Destination** - Съдържа местоположението където трябва да бъде преместен или копиран даден ресурс; използва се в COPY и MOVE заявки;

## 3. WebDAV демонстрация

### 3.1. Настройване на Apache Httpd сървъра

- Модулите `mod_dav` и `mod_dav_fs` са включени по подразбиране в дистрибуциите на Apache.
- Глобални директивите в `httpd.conf`  
`LoadModule dav_module modules/mod_dav.so`  
`LoadModule dav_fs_module modules/mod_dav_fs.so`

```
<IfModule mod_dav.c>  
    DAVLockDB /usr/local/etc/apache22/dav/lockdb  
    DAVMinTimeout 600  
</IfModule>
```

```
<Location /firstdav>  
    DAV On  
</Location>
```

- Настройване на правата на WebDAV директорията:  
`# chown www:www firstdav/`  
`# chmod 770 firstdav/`

[У дома](#)[Print](#)[Заглавие](#)[Съдържание](#)[◀](#) [▶](#)[◀](#) [▶](#)[Стр. 12 от 21](#)[Назад](#)[Екран](#)[Затвори](#)[Край](#)

## 3.2. Настройки на клиента

- Настройване на прокси сървъра (например чрез на HTTP\_PROXY системната променлива) ако е нужно
- Сесия с клиента cadaver

```
$ cd /tmp/
$ echo "This is the first WebDAV test" > sample.txt
$ cadaver
dav:.!> open http://localhost/firstdav/
dav:./firstdav/> put sample.txt
Uploading sample.txt to '/firstdav/sample.txt'
Progress: [=====>]
  100.0% of 192 bytes succeeded
dav:./firstdav/> close
Connection to 'localhost' closed.
dav:.!> exit
$
```

### 3.3. Настройване за сигурност

В горния пример WebDAV сървърът е достъпен от всеки за всичко. Използвайки дайджест автентикация, нивото му на сигурност може да се повиши:

```
LoadModule
```

```
auth_digest_module modules/mod_auth_digest.so
```

Дайджест автентикацията се включва чрез следните директиви

```
<Location /digestdav>
```

```
DAV On
```

```
AuthType Digest
```

```
AuthName "WebDAV Repository"
```

```
AuthUserFile /usr/local/etc/apache22/.digestdav
```

```
AuthDigestDomain /digestdav
```

```
<LimitExcept GET HEAD OPTIONS>
```

```
Require valid-user
```

```
</LimitExcept>
```

```
</Location>
```

[У дома](#)

[Print](#)

[Заглавие](#)

[Съдържание](#)



Стр. 14 от 21

[Назад](#)

[Екран](#)

[Затвори](#)

[Край](#)

## 4. WebDAV сървъри и клиенти

[У дома](#)

[Print](#)

[Заглавие](#)

[Съдържание](#)



Стр. 15 от 21

[Назад](#)

[Екран](#)

[Затвори](#)

[Край](#)

## 4.1. Клиенти

- cadaver
- Nd
- Emacs с плъгина eIDAV
- JEdit
- Jakarta Slide
- Nautilus

[У дома](#)

[Print](#)

[Заглавие](#)

[Съдържание](#)



Стр. 16 от 21

[Назад](#)

[Екран](#)

[Затвори](#)

[Край](#)

## 4.2. Сървъри

- Apache 2.2, или 1.3 с mod\_dav
- Zope
- Magi
- Jakarta Slide с Tomcat

## 5. Сигурност при WebDAV

- HTTP Challenge/Response Authentication Framework;
- Два вида протоколи за автентикация: basic и digest authentication;
- Basic протокола се счита за несигурен;
  - При него името и паролата на потребителя се пращат почти некодирани.
  - Зложелател би могъл да декодира паролата на потребителя не само при неговото влизане в системата, но по всяко време след това - демонстрация.
- Basic протокола е за препоръчване да се използва единствено в строго подсигурени мрежи, където сигурността не е приоритетна, както и в комбинация с https.

## 5.1. Digest автентикация

Digest протокола е създаден за да усъвършенства пропуските на basic автентикацията.

Основни характеристики:

- Никога не изпраща паролата и потребителското име на потребителя в прав текст - за целта ги криптира използвайки еднопосочни digest функции.
- Дори заявката на клиента да бъде прихваната декодирането на паролата е изключително трудно.
- За още по-голяма сигурност сървърът изпраща на клиента nonce-token, който се сменя доста често. Клиента добавя получения nonce-token, към края на паролата си преди да я кодира.

Демонстрация на заявките към сървъра използвайки digest автентикация.

[У дома](#)

[Print](#)

[Заглавие](#)

[Съдържание](#)



Стр. 19 от 21

[Назад](#)

[Екран](#)

[Затвори](#)

[Край](#)

## 6. Заключение думи

[У дома](#)[Print](#)[Заглавие](#)[Съдържание](#)[Стр. 20 от 21](#)[Назад](#)[Екран](#)[Затвори](#)[Край](#)

## 6.1. Потенциални ползи от WebDAV

- Превръща интернет в място за разработка, в единна файлова система.
- Ползите за потребителите
- Ползите за авторите на ресурси
- Ползите за администраторите

## 6.2. Бъдещето пред WebDAV

[У дома](#)[Print](#)[Заглавие](#)[Съдържание](#)[Стр. 21 от 21](#)[Назад](#)[Екран](#)[Затвори](#)[Край](#)

## 7. Достъп до настоящата презентация

Изходният код на презентацията може да се изтегли от Subversion хранилището на SF.net чрез командата:

```
svn co
```

```
https://svn.sourceforge.net/svnroot/ralitsa/trunk ralitsa
```

Подробни бележки по презентацията могат да се видят на <http://ralitsa.sourceforge.net/wt2006/> (обновяват се по SSH чрез rsync) и на <http://sharemation.com/rambius/wt2006/> (обновяват се по WebDAV чрез sitecopy).