

# Curs 12

## Servicii pentru dezvoltatori

Gestiunea Serviciilor de Rețea

12 ianuarie 2012

*Before software can be reusable it first has to be usable.*

*Ralph Johnson*

Git

Servicii Git

Redmine

Înceiere

Întrebări

Git

Servicii Git

Redmine

Înceiere

Întrebări

- ▶ version control systems/revision control systems
- ▶ source code management
- ▶ repository, clone/checkout/working copy, commit, HEAD, branch, sincronizare, conflict, revert, revision/version, merge, tag, trunk/mainline
- ▶ centralizate: CVS, SVN, Perforce
- ▶ distribuite: Darcs, Monotone, Git, Mercurial, Bazaar
- ▶ repository-ul identificat prin URL
  - ▶ `svn+ssh://gigi@local.net/super-project`
  - ▶ `http://github.com/vmchecker/vmchecker.git`

- ▶ SCM distribuit
- ▶ creat de Linus Torvalds
- ▶ folosit în proiecte mari (Linux, Gnome, KDE, Perl, Ruby on Rails, Android etc.)
- ▶ `apt-get install git` (`git-core` pe sisteme mai vechi)
- ▶ clientul – `git`
- ▶ serverul/daemon-ul – `git-daemon`
- ▶ URL-uri
  - ▶ local – `/var/cache/git/test-repo.git`
  - ▶ SSH – `fkf@home.com:benga.git`
  - ▶ HTTP – `http://home.com/benga.git`
  - ▶ Git – `git://home.com/benga.git`

- ▶ clonare – `git clone URL`
- ▶ configurare – `git config --global user.name "Gogu Sclipici"`
- ▶ commit – `git add ., git commit -m 'add pretty printer'`
- ▶ actualizare către server – `git push origin master`
- ▶ actualizare de la server – `git pull, git pull --rebase`
- ▶ reconfigurare URL
  - ▶ `git remote rm origin`
  - ▶ `git remote add origin URL`

- ▶ creare repository (eventual în /var/cache/git/)
  - ▶ `mkdir repo.git && cd repo.git`
  - ▶ `git init --bare`
- ▶ populare repository (eventual pe alt sistem)
  - ▶ `mkdir local.git && cd local.git`
  - ▶ `git init .`
  - ▶ `touch test.txt && git add .`
  - ▶ `git commit -m 'initial commit'`
  - ▶ `git remote add origin john@ivory.net:repo.git`
  - ▶ `git push origin master`



- ▶ configurare în home-ul unui utilizator
- ▶ `cd ~student`
- ▶ `cd repos; mkdir project.git; cd project.git; git init --bare`
- ▶ URL – `student@test.cs.pub.ro:repos/project.git`
- ▶ populare repository

- ▶ daemon ce servește repository-uri Git (portul 9418)
- ▶ `apt-get install git-daemon-run`
  - ▶ folosește `runit`, `sv` – interacțiune cu servicii
- ▶ `sv start git-daemon`, `sv stop git-daemon`
- ▶ configurat în `/etc/sv/git-daemon/run`
- ▶ log-uri în `/var/log/git-daemon/`
- ▶ pentru a exporta un repository, trebuie să conțină fișierul `git-daemon-export-ok`
- ▶ implicit configurat doar `read-only`

- ▶ HTTP – dumb protocol
- ▶ fiecare repository trebuie să activeze hook-ul de post-update
  - ▶ `git update-server-info`
- ▶ configurare Apache

```
Alias /git /var/cache/git/
```

- ▶ `git clone http://gamma.org/git/rango-repo.git`

- ▶ pentru a se putea scrie e nevoie de
  - ▶ suport DAV (a2enmod dav\_fs)
  - ▶ utilizatorul aferent serverului web (www-data) să aibă drept de scriere
- ▶ configurație naivă

```
Alias /git /var/cache/git/  
<Directory /var/cache/git/>  
    DAV On  
</Directory>
```

▶ configurație strictă

```
Alias /git /var/cache/git/  
<Directory /var/cache/git/>  
    DAV On  
    AuthType Basic  
    AuthName "Git"  
    AuthUserFile /etc/apache2/git-repos.htpasswd  
    Require valid-user  
</Directory>
```

▶ configurație “comună”

```
Alias /git /var/cache/git/  
<Directory /var/cache/git/>  
    DAV On  
    AuthType Basic  
    AuthName "Git"  
    AuthUserFile /etc/apache2/git-repos.htpasswd  
    <LimitExcept GET HEAD PROPFIND OPTIONS REPORT>  
        Require valid-user  
    </LimitExcept>  
</Directory>
```

▶ configurație non-DAV pentru HTTP (read-only) + configurație strictă pentru HTTPS

Git

Servicii Git

Redmine

Înceiere

Întrebări

- ▶ folosirea opțiunii `command` în fața cheii publice SSH (directorul `~/.ssh/authorized_keys`)
- ▶ folosit de Gitis sau Gitolite
- ▶ `command="/usr/share/gitolite/gl-auth-command razvan.deaconescu"`

- ▶ “access control layer on top of Git”
- ▶ gestiune facilă/centralizată a mai multor repository-uri Git
  - ▶ acces pe bază de chei SSH
  - ▶ autorizare – ce utilizator/cheie publică are acces la ce repository
- ▶ permite controlul accesului la nivel de branch (Gitolite nu oferă aceste facilități)



- ▶ `apt-get install gitolite`
- ▶ creează și configurează utilizatorul gitolite  
`ixlabs:~# ls ~gitolite`  
`projects.list repositories`
- ▶ `gl-setup /path/to/ssh/public/key.pub -`  
“administratorul” Gitolite
- ▶ `git clone gitolite@tron.net:gitolite-admin`
  - ▶ repository-ul administrativ  
`.../admin/ixlabs-gitolite-admin$ tree`  
`.`  
`|-- conf`  
`| '-- gitolite.conf`  
`'-- keydir`  
 `|-- razvan.deaconescu.pub`  
 `'-- tavi.purdila.pub`

```
.../admin/ixlabs-gitolite-admin$ cat conf/gitolite.conf
repo gitolite-admin
    RW+ = razvan.deaconescu

@ixlabs      = inspectos vmgen sshsync lxc valgrind robocheck
@them        = razvan.deaconescu tavi.purdila

repo @ixlabs
    RW+ = @them
    R   = @all
```

- ▶ R – read (poate clona)
- ▶ W – write (poate face push)
- ▶ + – rewind (poate reseta branch-ul)
- ▶ configurări la nivel de branch

- ▶ acces vizual web la repository-uri
- ▶ `apt-get install gitweb`
- ▶ configurație implicită în `/etc/apache2/conf.d/gitweb`
- ▶ repository-urile trebuie să fie accesibile serverului web
- ▶ configurare în `/etc/gitweb.conf`
  - ▶ `@git_base_url_list`
- ▶ documentație în `/usr/share/doc/gitweb/`

- ▶ accesibile serverului web
- ▶ editare fișier description din repository
- ▶ adăugare configurare de ownership în fișierul config

```
gitolite@ixlabs:~/repositories$ cat lxc.git/config
[core]
    repositoryformatversion = 0
    filemode = true
    bare = true

[gitweb]
    owner = IxLabs
```

Git

Servicii Git

Redmine

Înceiere

Întrebări

- ▶ sisteme pentru coordonarea/gestiunea proiectelor software
- ▶ integrarea mai multor funcționalități
  - ▶ wiki
  - ▶ issue/request/ticket tracker
  - ▶ repository browser
  - ▶ timeline/roadmap
  - ▶ feed-uri RSS
  - ▶ controlul accesului
- ▶ hosted: Sourceforge, Google Code, Assembla, GNU Savannah, BerliOS
- ▶ client-server: Trac, Redmine, JIRA, Launchpad

- ▶ influențat de Trac
- ▶ Ruby on Rails
- ▶ cross-platform & cross-database (MySQL, PGSQL, SQLite)
- ▶ suport pentru SVN, Git, Mercurial, Bazaar, Darcs
- ▶ suport pentru mai multe proiecte în aceeași instalare
- ▶ roluri pentru membrii proiectului
- ▶ calendar, diagramă Gantt
- ▶ forumuri, știri
- ▶ extensibil prin module



- ▶ `apt-get install redmine` – pe versiunile mai noi de Ubuntu și Debian
- ▶ `/usr/share/redmine/`
- ▶ `dpkg-reconfigure redmine` – pentru adăugarea unei noi instanțe (implicit default) sau configurare
- ▶ `/etc/redmine/*/` – un director aferent unui proiect
  - ▶ `database.yml`
  - ▶ `email.yml`
  - ▶ `session.yml` – generat automat
- ▶ fișiere `/var/lib/redmine/*/files/`
- ▶ log-uri `/var/log/redmine/*/*.log`
- ▶ pentru SQLite  
`/var/lib/dbconfig-common/sqlite3/redmine/instances/`
- ▶ documentație în `/usr/share/doc/redmine/`

- ▶ apt-get install mongrel-cluster
- ▶ /etc/mongrel\_cluster

```
<Proxy balancer://mongrel_cluster>
```

```
    Order allow,deny
```

```
    Allow from all
```

```
    BalancerMember http://127.0.0.1:8000
```

```
    BalancerMember http://127.0.0.1:8001
```

```
    BalancerMember http://127.0.0.1:8002
```

```
</Proxy>
```

```
<Location /redmine>
```

```
    RewriteEngine On
```

```
    RewriteCond %{DOCUMENT_ROOT}/%{REQUEST_FILENAME} !-f
```

```
    RewriteRule ^{(.*)}$ balancer://mongrel_cluster%{REQUEST_URI}
```

```
</Location>
```

▶ a2enmod passenger

```
RailsEnv production
```

```
SetEnv X_DEBIAN_SITEID "projects.rosedu.org"
```

```
Alias "/plugin_assets/" /var/cache/redmine/default/plugin_a
```

```
DocumentRoot /usr/share/redmine/public
```

```
<Directory "/usr/share/redmine/public">
```

```
    Order allow,deny
```

```
    Allow from all
```

```
</Directory>
```

- ▶ a2enmod fcgid
    - ▶ /etc/apache2/mods-enabled/fcgid.conf – SocketPath
- ```
DefaultInitEnv X_DEBIAN_SITEID "default"

DocumentRoot /usr/share/redmine/public
<Directory "/usr/share/redmine/public">
    Order allow,deny
    Allow from all
</Directory>
```

- ▶ `http://www.redmine.org/plugins`
- ▶ copiere + dezarhivare sau clonare în `${REDMINE_ROOT}/vendor/plugin`
- ▶ trebuie repornit Redmine
- ▶ vizibile în meniul de Administrare, opțiunea Plugins

- ▶ gestiune utilizatori
  - ▶ permitere utilizatori înregistrați
  - ▶ creare utilizatori
  - ▶ autentificare prin LDAP
  - ▶ adăugare utilizatori la proiecte
- ▶ gestiune proiecte
  - ▶ creare proiecte
  - ▶ componente/module folosite (wiki, tracker, Gantt, boards)
  - ▶ configurare repository (cale către repository pentru repository browser)
  - ▶ roluri și utilizatori
- ▶ opțiuni, enumerări (trackers, user roles, issue states)

- ▶ custom issue queries – clasificare după diverse criterii
- ▶ RSS feeds (activități și issue-uri)

Git

Servicii Git

Redmine

Înceiere

Întrebări



- ▶ versioning
- ▶ Git
- ▶ clone
- ▶ SSH, HTTP
- ▶ git-daemon
- ▶ DAV (write)
- ▶ Gitolite
- ▶ Gitweb
- ▶ project management
- ▶ Redmine
- ▶ mongrel-cluster
- ▶ passenger
- ▶ FCGID

- ▶ <http://gitimmersion.com/>
- ▶ <http://git-scm.com/>
- ▶ <https://git.wiki.kernel.org/index.php/Gitweb>
- ▶ <https://github.com/sitaramc/gitolite>
- ▶ <http://www.redmine.org/>
- ▶ <http://www.redmine.org/plugins>

Git

Servicii Git

Redmine

Înceiere

Întrebări