Electgon
Publications Platform

Trac

Quick Installation and Configuration Guide

Contents

1   Trac Installation

To be able to install Trac, you have to install setuptools and Genshi

- Install setuptools by

```
$sudo apt-get install python-setuptools
```

- Then you can install Genshi by

```
$sudo easy_install Genshi
```

- Download Latest version of Trac form the following link:

http://trac.edgewall.org/wiki/TracDownload#LatestStableRelease

- Then, extract it, browse extracted file then type:

```
$sudo python ./setup.py install
```

Trac should be installed now.

2   Creating a Project Environment

A new environment can be created using

```
$trac-admin /path/to/myproject initenv
```

For this tutorial a directory called myspace has been created, for instance a command like the following has been used

```
$trac-admin /path/to/myspace initenv
```

trac-admin will prompt you for the information it needs to create the environment, such as the name of the project and the database connection string. If you're not sure what to specify for one of these options, just press Enter to use default values.

3   Deploying Trac

After having created a Trac environment, you can easily try the web interface by running the standalone server tracd

```
$tracd --port 8000 /path/to/myproject
```

Then, fire up a browser and visit http://localhost:8000/. You should get a simple listing of all environments that tracd knows about. Follow the link to the environment you just created, and you should see Trac in action.

4   Authentication

4.1   Adding Users

So far by issuing previous command "tracd –port 8000 /path/to/myproject" you can access
your environment as a standalone using means of web browsing as stated before. But using
this command you can only navigate through your environment as anonymous. i.e. you can
only view almost welcome page.

To administrate your environment, you should initialize your project with authorized
user. So you have to define first which users can access your environment. Trac has permission
module that can be used for that.

First you have to define user names that yo wish them having specific access to your
environment. The following command is used to define your users

```
$trac-admin /path/to/projenv permission add myUser TRAC_ADMIN
```

In this command you have defined new user "myUser" with role TRAC_ADMIN.

TRAC_ADMIN has access to everything. You can check all available privileges in Trac at
wiki page of Trac (yourprojectspace/wiki/TracPermissions).

You can review all available users for your environment using the following command

```
$trac-admin /path/to/projenv permission list
```

4.2   Create Credentials

Now remaining to know how to set login credentials for that user. If you visited Trac wiki
page (yourprojectspace/wiki/TracStandalone#UsingAuthentication), you will find mainly 3
methods for creating credentials: using normal password file – using apachi .htpasswd file –
using python script. In this tutorial, I preferred to use the last method (using python). Below
is how you can create credentials then.

In Trac wiki, you can find the following python script

```python
from optparse import OptionParser
# The md5 module is deprecated in Python 2.5
try:
    from hashlib import md5
except ImportError:
    from md5 import md5
realm = 'trac'

# build the options
usage = "usage: %prog [options]"
parser = OptionParser(usage=usage)
parser.add_option("-u", "--username",action="store", dest="username", type =  "
    string", help="the username for whom to generate a password")
parser.add_option("-p", "--password",action="store", dest="password", type =  "
    string", help="the password to use")
parser.add_option("-r", "--realm",action="store", dest="realm", type = "string"
    , help="the realm in which to create the digest")
```

```
(options, args) = parser.parse_args()

# check options if (options.username is None) or (options.password is None):
   parser.error("You must supply both the username and password")
if (options.realm is not None):
   realm = options.realm

# Generate the string to enter into the htdigest file
kd = lambda x: md5(':'.join(x)).hexdigest()
print ':'.join((options.username, realm, kd([options.username, realm,  options.
    password])))
```

Take a copy and save it in a file named (for instance) "trac-digest.py". This script is used to create credentials using the following command

```
$python trac-digest.py -u myUser -p mypass12 >> pypass.txt
```

In this command we defined password "mypass12" for the user "myUser" that we added in Trac before. Output of this command is saved in file "pypass.txt" which we will use later to run Trac. If you opened this file you will find its content is written like:

myUser:trac:a44ce2cc60c729e8a9216b15b205fff6

myUser is the user, trac is value for a parameter called realm, it acts like grouping organization of users so it is not important, by default it is defined as trac. But if you want to define it use python command as follows

```
$python trac-digest.py -u myUser -p mypass12 -r mygroup >> pypass.txt
```

You can add as many users as you need in this file by issuing this python command for each user.

Deploy Trac with Authentication

So far we have ready user credentials file "pypass.txt" that we can import into Trac while deploying it. Now running Trac should be done using command like the following

```
$tracd -p 8000 --auth=myspace,/home/path/to/pypass.txt,trac /path/to/myspace/
```

In this command we used tracd (as before) to run at port 8000 then we defined authentication for Trac. myspace is project that we work on. Then we defined path to credentials file that we created in previous section. Then we provided realm name as defined in the credentials file, you can write also any trivial name instead. Finally we defined path to our project. For more information you can check wiki pages of Trac.

Now run your environment again using a web browser, you can access many new sections like Tickets, Admin, etc.

5   Trac Configuration

When you create your project, you will find a folder in project directory called "conf". In this folder there is a configuration file "trac.ini" that is used to control how your environment should look like. For example we can add the following to our environment

5.1 Adding Logo:

- In your project directory you can find a folder called "htdocs", lets save your company logo as a png image in this folder.

- Open "trac.ini" file.

- In [header_logo] section define height for instance to be '50', width to be '200'. src should indicate image name, it should be defined as src = site/company_logo.png where company_logo is my image name.

- In [inherit] section, you have to define path to "htdocs" folder.

- Now check your environment.