

# Taskforce: Test/Assignment Submission Grading and Reporting Tool

Igor Wojnicki

April 18, 2006

## 1 Introduction

Taskforce is a set of scripts which enable automatic assignment or test submission, supports grade calculation, feedback and grade reports. *Teacher* refers to a user which gives and grades tests or assignments. *Student* refers to a user which takes tests and assignments. Name *test* refers to test or assignment.

## 2 Teacher's Perspective

Scripts for teachers:

**install** installs configuration file in `/.taskforce` – keep in mind that each *student* needs *read access* to this very file!

**tfprep** prepares appropriate entries in the directory structure to accommodate a new test/assignment

**tfend** disables submission for a given test

**tfsend** sends back grades

**tfsun** auxiliary script for partial grade calculation

**tfgrades** generates grade report

### 2.1 Configuration

All scripts must be in a directory on `$PATH`. Before starting using Taskforce, you need to run **tfinstall** script. Two questions are asked during the installation process:

- what is the Taskforce root directory (`$TFROOT` variable in `~/.taskforce`, it is `~/taskforce` by default) – it is where all data is going to be stored; this directory is referred to as `$TFROOT` later on,
- what is the group the teacher belongs to (`$TFGROUP` variable in `~/.taskforce`), simultaneously no student is allowed to be assigned to this group (provides authorization mechanisms for the submission process)?

The **tfinstall** script creates `~/.taskforce` configuration file which is subsequently used by other Taskforce scripts. Important: `~/.taskforce` must be world readable (or at least student readable) since it is used by all Taskforce scripts.

### 2.2 Preparation

To prepare a structure for a test the teacher should issue:

```
tfprep name
```

Where **name** is the name of the test. The test name might include course name or number and test name or number for easy recognition, i.e.: first test for Operating Systems course could be created by issuing:

```
tfprep os1
```

Subsequent Operating Systems tests could be named **os2**, **os3** etc. The process creates appropriate directory named after the test name in **\$TFROOT** (see Section 4). From now on, students are allowed to submit their course work. There is a certain security issue related to the test preparation process, see Section 5.

## 2.3 Deadline

To disallow students to submit their course work (when the test is due) the teacher should issue:

```
tfend name
```

where **name** is the test name. This process can be automated by other means i.e.. using **at** or **crontab**. There is a certain security issue related to the test preparation process, see Section 5.

## 2.4 Grading

Student course work (a set of files of any type or contents) for a particular test and student is saved in **\$TFROOT/test/student/in** directory upon its submission (regarding the submission process see Section 3.1). Where **test** is the test name, **student** is the student name. The directory contains original files which should not be altered by the teacher (to have the original course work at hand). In addition there is also **tflog** file there which contains extra information about the submitted files: their flags and submission time-stamp.

To grade a test the teacher should create a file in **\$TFROOT/test/student** directory. Every entry in any file created in this directory which starts with **#** followed by an optional **-**, followed by a real number is treated as a partial score for this test. For example if there is a file which contains:

```
#2 algorithm
#3.5 programming
Unfortunately you have received some penalty
#-1 cheating penalty
```

It means that the student receives  $2 + 3.5 - 1$  points in total for the test. If there is more than one file partial points from all files are added up.

It might be convenient (especially for programming assignments) to include marks in the student course work context. In order to do this original student files from **in** directory can be copied to **\$TFROOT/test/student** and marks can be incorporated in them.

Any file created in **\$TFROOT/test/student** directory will be send back to the student as a feedback from the test.

## 2.5 Feedback

The feedback is sent by email. To send a feedback regarding particular test to all the students which took it issue:

```
tfsend test subject file
```

where **test** is the test name, **subject** is the subject of an email message the students are going to receive, and **file** is a file name which contents is going to be included at the beginning of the email (contents of this file can explain for example the grading policy).

Each student taking the test receives a separate email message. The body of the message consists of the contents of the file specified by the teacher (as mentioned above), the sum of all partial points, and the contents of all files which are in **\$TFROOT/test/student** directory which is: partial points and the feedback as created by the teacher.

## 2.6 Grade Report

To display all grades for all students issue:

```
tfgrades test
```

where `test` is the test name, wild-cards are welcome, i.e.: to see grades from all tests which names start with "os":

```
tfgrades os*
```

To see grades for particular students:

```
tfgrades os* wojnicki dubaj
```

The grades are displayed as *newline* separated records. Each record consists of comma separated fields. The first record is a header which describes fields in subsequent records:

```
NAME,test_1,test_2,...,test_n,TOTAL
```

where `test_1..test_n` are the test names. Subsequent records are as follows:

```
student_name,test_1_grade,test_2_grade,...,test_n_grade,total_score
```

The test grades and the total score are calculated automatically from partial grades entered by the teacher during the grading process. For example, if there are two tests registered in the system: `os1` and `os2` and there are two students: `smith` and `white`, the output will look like this:

```
NAME,os1,os2,TOTAL
smith,21.5,0.5,22
white,5,-0,6
```

The partial grade of -0 means that the student didn't take this particular test or the test has not been graded yet.

## 3 Student's Perspective

Scripts for students:

```
tfsubmit test submission
```

### 3.1 Submission

To submit some files regarding a particular test the student should issue:

```
$TROOT/testname/tfsubmit file
```

where `testname` is the test name, and `file` is the file to send. Multiple files can be specified as subsequent arguments, directories are not permitted. For example the following command will submit files: `file1`, `file2`, and multiple files which names start with `other` from current directory as `test01`.

```
$TROOT/test01/tfsubmit file1 file2 other*
```

There is a certain security issue related to the test preparation process, see Section 5.

## 4 Directory Hierarchy

```
$TFROOT          Taskforce root directory (default: ~/taksforce)
test_name        test directory
  student_name   student directory
    in           contains submitted files and submission log
      tflog      submission log (file permissions, timestamp)
```

## 5 Security

The Taskforce configuration file, which is `~/.taskforce` `availability` must be at least student and teacher readable. All Taskforce scripts rely on it.

Upon preparing a test (calling `tfprep`) a directory is created with permissions set to: `drwxrwsrwt`. The directory has its group owner set to `$TFGROUP` as defined in the configuration file. Such permissions make the submission process possible. Furthermore `tfsubmit` script is generated and placed in it.

Upon calling `tfend` the permissions on the test directory are set to: `drwx-----` to prevent students from resubmitting files or tampering with them.

### 5.1 Security Threats

After submitting a file the student is free to alter permissions on the student directory within the Taskforce directory hierarchy. In particular he or she can alter permissions or group ownership on this directory which might deny the teacher access to it. Similarly any user is allowed to save any file at `$ROOT/test` directory before the test is due. To prevent such a tampering a different submission method based on a *suided* application can be developed.

## 6 Scripts

The Taskforce scripts require the following tools: `bash`, `ls`, `bc`, `basename`, `mkdir`, `chown`, `chmod`, `chgrp`, `cat`, `mail`, `awk`.

## 7 Why? – Pros and Cons

To simplify grading/grade submission process and increase its effectiveness. Partial grades are added up automatically on demand. There is no need of having separate grade sheets. Students receive decent feedback with detailed remarks and partial grades/points. Simple interface. The files containing feedback and partial grades *must* be regular text files – otherwise the grade calculation process can be disrupted.