Testing PC2 Shadow Mode with an Existing Contest

1. Introduction

There is a need to be able to test the "Shadow Mode" capability of a PC2 system. In principle this could be done by running one instance of PC2 acting as a "remote CCS" implementing the CLICS Contest API, and then running a second instance of PC2 which connects to the "remote PC2 CCS" (via the API) and obtains submissions which it then judges (saving the results for shadow-comparisons with the results of the "remote CCS").

However, in addition to the fact that it's not ideal testing procedure to test a PC2 Shadow system (which is supposed to be interfacing with an arbitrary remote system implementation of the CLICS Contest API) by connecting it to another instance of PC2, there are two substantial issues with trying to use PC2 to act as a "remote CCS" for Shadow testing. The first is that the PC2 implementation of the CLICS Contest API is not fully completed and has not even been thoroughly vetted to be completely correct in those CLICS API endpoints that it does implement. The second is that even if PC2 did fully and correctly implement the CLICS Contest API, there is currently no simple mechanism for having the "remote PC2 CCS" stagger runs over time (that is, there's no simple and flexible "contest replay" mechanism).

This article describes how to take an existing (previously-run) PC2 contest and feed it into a PC2 Shadow CCS. In other words, it describes how to convert the PC2 archive for a previously-run contest and make it appear as if that contest is currently running on a "Remote CCS" feeding data to a PC2 Shadow CCS. It does this without the need to actually run a PC2 Remote CCS, without the need to have PC2 provide a "CLICS Contest API implementation", and with support for a simple mechanism for adjusting the "replay" time (that is, the rate at which the "remote system" feeds submissions to the PC2 Shadow system.

There are three basic steps involved in feeding an existing PC2 contest into a PC2 Shadow system as if the existing contest were coming from a remote CLICS Contest API: (1) generate a valid CLICS Event Feed from the existing contest; (2) load the CLICS Event Feed into the Shadow Mode "MockContestAPIAdapter"; and (3) run a PC2 Shadow Mode system, connecting it to the "remote CCS" provided by the Mock Adapter.

2. Generating a CLICS-compatible Event Feed from an Existing Contest Archive

We assume that a contest has been previously run using some build of PC2 Version 9.6 or higher; we'll call this the "original contest". We also assume that the original contest has been completed; i.e. it has been "finalized" in PC2 prior to saving the archive. This means that the archive for that original contest will have a "profiles" folder containing all the information for that contest. However, the Event Feed generated by PC2 is not complete or correct with respect to the CLICS Contest API which is expected by the PC2 Shadow Mode. The steps for generating a CLICS Event Feed from this original contest archive are as follows.

2.1. Install a copy of PC2 which is the same as the version which was used to run the original contest.

2.2. Unzip the archive from the original contest into its own folder.

2.3. In the unzipped archive folder, start a PC2 server (using the installed PC2 system which is the same version as what was used to run the original contest and to create the archive). This should load the profile for the original contest and then prompt for the original contest master password. Enter the original contest master password when prompted.

2.4. Start a PC2 Admin in the unzipped archive folder.

2.4.1. Verify that the Admin “Configure” tab contains the proper configuration for the original contest: same accounts, groups, languages, judgements, contest problems, etc. The “Times” tab should show that the contest is “Stopped” and has no more remaining time (because as stated above, the assumption is that the contest was finished and finalized).

2.4.2. Verify that the Admin “Runs” tab shows the submissions which occurred during the contest.

2.5. Using the Admin “Reports” tab, select the “Event Feed JSON” report and then click “View Report” then “Save”. This will save a copy of the report under the “profiles/reports” folder using file extension “.txt”. You can also save it under another name/extension (e.g. “.json”).

2.6. Using the **Server** “Reports” tab, select the “Extract Replay Runs” report, click “View Report” and then “Save”. This will save a copy of the Extract Replay Runs report (again, under “profiles/reports”), *and will also construct a subfolder containing all of the extracted runs* (again, under “profiles/reports”.

2.7. Merge the Event Feed JSON report and the “Extract Replay Runs” report into a single CLICS-compatible Event Feed (that is, a single event feed which contains not only the data in the PC2 Event Feed JSON output but *also* contains all of the Submissions and Judgements found in the Extract Replay Runs. To do this merger, run the tool “edu.csus.ecs.pc2.tools.EventFeedReplayRunsMerger”. The usage for this tool is:

```
java  EventFeedReplayRunsMerger  <pc2EventFeedJSONFile>  <pc2ExtractReplayRunsFile>
```

with an optional third argument `<mergedEventFeedOutputFile>` (defaults to stdout).

The output produced by this tool will be CLICS-compatible Event Feed.

3. Loading the CLICS Event Feed into the Mock Adapter

3.1. Edit the pc2v9.ini file to override the MockContestAPIAdapter’s Event Feed Input file name to be the name of the CLICS-compatible Event Feed file created in step (2) above. To do this, add a [shadow] section to the pc2v9.ini file as follows:

```
[shadow]
overrideeffilename=name_of_CLICS_Event_Feed_file
```

4. Run a PC2 Shadow Mode system which reads the Event Feed

4.1. Using a version of PC2 with Shadow Mode code, start a PC2 Server using --load to load the CDP for the original contest (so that the new PC2 system has the same configuration as that which existed in the original contest).

4.2. Start a PC2 Admin and verify that the contest configuration is the same as for the original contest. Insure that there is at least one “feeder” account (create one if not).

4.2.1. Edit the feeder account and activate the "Shadow Proxy Team" permission for that account if it is not already activated.

4.3. Using the Admin ConfigureContest>Problems tab, select and edit each problem, as follows:

4.3.1. Check the "Judging Type"; if necessary, *change it so that it is "Computer" with No Manual Review.* (This is necessary to insure that, even if the judging in the original contest was manual, the Shadow can do all judging automatically.)

4.3.2. Check the "Output Validator". If the Output Validator selection is "Use Custom (User-supplied) Validator", insure that the code for the custom validator is available at the specified location (file path). If the custom validator code isn't available, change the Output Validator type to "Use CLICS Validator"; note that this could potentially result in disagreements between the Shadow CCS and the original contest results.

4.4. On the Admin "Auto Judge" tab, insure that each contest problem is identified as being "auto-judged" by at least one judge account.

4.5. On the Admin "Languages" tab, edit each language:

4.5.1. Insure that the language has a CLICS "ID Code" (***java***, ***cpp***, ***c***, ***python2***, or ***python3*** for example).

4.5.2. Insure that all language compile and execute arguments are correct for the platform. (If running the ACPC 2019 contest, for example, the gcc/g++ argument "-fsplit-stack" is invalid on a Windows machine; other language command items may also need updating for the current platform).

4.6. On the Admin "Settings" tab, scroll down to "Remote CCS Settings" and do the following:

4.6.1. Check (enable) the "CCS Test Mode" checkbox

4.6.2. Check (enable) the "Enable Shadow Mode" checkbox

4.6.3. Enter a URL string in the "Primary CCS URL" textbox. The URL should be of the form file://path/to/extracted/runs, where "path/to/extracted/runs" is the full path to the folder where the PC2 ExtractReplayRuns were stored in Step (2) above. Specifying this URL will cause the "Mock Adapter" to return the files stored in the Extract Replay Runs folder as if they were coming from a remote CCS.

4.6.4. Enter a string in the "Primary CCS Login (account)" textbox. It doesn't matter what actual string is entered (since the Mock Adapter is not actually going to login to a remote system) as long as some non-zero-length string is entered.

4.6.5. Enter a string in the "Primary CCS Password" textbox. Again, the actual value doesn't matter (but the code edit-checks the field to make sure something has been entered).

4.7. Note: instead of manually editing the above settings (Judging Type, Output Validator, AutoJudge accounts, languages, and Settings), an alternative is to update the YAML files for the contest prior to starting the PC2 server.)

4.8. On the Admin “Times” tab, click “Start All”.

4.9. Start one autojudge for each autojudging account specified in the Admin “Auto Judge” tab.

4.10. Login to an Event Feeder account. Select “Shadow Mode” and verify that the Shadow Mode settings enter in the Admin (above) are present in the Feeder. (Note: the “Enable Shadow Mode” checkbox may not be checked; if so, check (enable) it.

4.11. Click “Start Shadowing”.