AcroTeX.Net

The **eq-pin2corr** package
Apply PIN security to quizzes

D. P. Story

Table of Contents

Part I: Version 1.0

In this part of the manual, we document the features of the eq-pin2corr package as they originally appeared. In addition to the features documented here, version 2.0 adds a number of features, some of which were suggested by Tahir Y. These additional features are documented in of this manual.

1. Introduction

This package is an add-on to the quiz environment of the exerquiz package.

Purpose. This package adds PIN security to a quiz created by the quiz environment. To correct a quiz, the document consumer must press the Correct button of a quiz and successfully enter the correct PIN number. For example,

Solve each, passing is 100%.

1. The sum of 1 and 1 is …

    0   1   2   3   4

2. $\cos(\pi) =$

3. $\dfrac{d}{dx}\sin(x) =$

☛ Take the quiz, when finished, press the Correct button and enter the PIN number to correct the quiz.

(PIN: 5243)

Answers:

This package is designed for the educational sector, for instructors who use the quizzes from the exerquiz package to assess the understanding their students of the course material.

PDF Viewers. Discussion of PDF viewers for the document author and the document consumer.

Instructor: Any PDF viewer may be used as a PDF *previewer*, sumatraPDF, for instance. To verify the newly created quiz is functioning correctly, Adobe Reader *AR* DC (AR) or Acrobat DC (AA) *must be used*.

AA

Document consumers (students): The exerquiz and eq-pin2tocorr extensively use JavaScript to perform many background tasks. For the student to have any *AR* success in this workflow, he/she must use Adobe Reader DC(AR).

Workflow. The package is designed for the following workflow:

create quiz   1. The instructor creates the quiz using the exerquiz and eq-pin2corr packages.

deliver quiz   2. The instructor delivers the "PDF quiz" to each student. (System drive or email)

take quiz   3. The student takes the quiz. The student can press the Correct but, unless s/he knows the PIN, the quiz is not marked up.

save quiz   4. The student saves the PDF quiz in Adobe Reader DC.

return quiz 5. The student returns the PDF to the instructor. (System drive or email)

correct quiz 6. The instructor opens the returned PDF quiz in AA/AR and presses the Correct button to mark up the quiz. The instructor *saves the quiz*. The instructor records the grade of the student.

returns quiz 7. The instructor returns the PDF, at some point, to the student.

happy people 8. Both instructor and student happily go on with their lives.

The implementation details are left to the instructor.

Document creation. Any of the PDF creators current to the LaTeX world can be used: (1) pdflatex, (2) lualatex, (3) xelatex, and my old friend, (4) dvips/Distiller (or dvips/ps2pdf). In the latter case (4), Acrobat is required to import Document JavaScripts.

2. The preamble

The minimal preamble for documents that use the eq-pin2corr package.

```
\documentclass{article}        % or some other class
\usepackage[⟨options⟩]{web}    % recommended but not required
\usepackage[⟨options⟩]{exerquiz}[2021/04/27]
\usepackage[⟨options⟩]{eq-pin2corr}
...
\declPINId{⟨pin-num⟩}{⟨hash-string⟩}
\classPINVar{⟨class-pin-var⟩}
...
\begin{document}
```

A recent version of exerquiz (2021/05/29 or later) is required; eq-pin2corr brings in the eq-save package (2021/04/27 or later).

3. Package options

There are two options for this package: showscore and !showscore. If you took the test on page 3, you will have noticed that when End Quiz control is pressed the phrase "Success! Now save and send to the instructor" appears in the \ScoreField or the

showscore \PointsField, this is the default behavior. Passing showscore in the optional argument list of eq-key2corr causes the actual score to appear in the same box (eg, Score: 2 out of 3), which is historically what always appears in these boxes. The other op-

!showscore tion !showscore is a convenience option to aid in switching from one option to the other without too much cut and paste. The default definition of the text string that appears is,

```
\flJSStr[noquotes]{\SaveAndSendMsg}{Success! %
Now save and send to the instructor}
```

Refer to the eformsman.pdf for a discussion of \flJSStr.

\showScoreOff **Local controls.** The two options can be turned off and on locally with \showScoreOff
\showScoreOn and \showScoreOn commands.

4. Setting the pin-hash values

In the preamble, as indicated above, are two commands, the first is required, the second is optional.

```
\declPINId{⟨pin-num⟩}{⟨hash-string⟩}
\numPINId
\classPINVar{⟨class-pin-var⟩}              (optional)
```

It is through the \declPINId command that the PIN security is set up. The command \numPINId expands to ⟨*pin-num*⟩ is not normally typeset into the document, but is used for documentation or demonstration purposes, such as in this document.

\declPINId{⟨*pin-num*⟩}{⟨*hash-string*⟩} The ⟨*pin-num*⟩ is a number, perhaps four digits, that is used to pass through the security of the Correct button.[1] Once you decide on the PIN number, you need to generate the corresponding *hash-string*. The hash-string is obtained from the demo file get-hash-string.pdf, the contents of that file is reproduced below.

Press the button labeled Push and enter your PIN number into the dialog box (enter, for example, 5243), then press OK, the corresponding hash-string appears in the text field, 02JRVZdRgYgCA-Rtje8VkD. Copy the PIN number (5243) into the first argument of \declPINId and copy the hash-string (02JRVZdRgYgCA-Rtje8VkD) from the text field into the second argument, like so,

```
\declPINId{5243}{02JRVZdRgYgCA-Rtje8VkD}
```

The above is placed in the preamble of the document.

When the user enters a PIN number (right or wrong), it is converted to a hash-string and compared with the hash string of the correct PIN number; if they match, then the quiz is corrected. It is important to note that the PIN number *does not appear anywhere* in the document, so it cannot be discovered. Knowledge of the correct hash-string does not help. Of course both PIN number and hash string appear in the source file of the document.

\classPINVar{⟨*class-pin-var*⟩} From the instructor's viewpoint, it takes a lot of effort to enter the PIN number for every quiz that is submitted (it may be in the hundreds). There is a way of avoiding the requirement of entering the PIN number, and that is through the use of the \classPINVar command. This method will work if you have Adobe Acrobat or if you only have Adobe Reader DC. The steps for doing this are as follows:

1. Decide on a class PIN variable name, it can be a generic one, or one specialize to your class, for example, \classPINVar{Calc2Sprg21}, then declare it in the preamble:

---

[1]The ⟨*pin-num*⟩ does not have to be a number, it can be any password (passcode) that is easy to remember. I prefer a four digit number.

```
\classPINVar{Calc2Spr21}
```

The value of the argument must be a valid JavaScript name.

2. Go to the user config.js file of AA/AR and edit that file by including the following line in it,

```
var Calc2Spr21 = "02JRVZdRgYgCA-Rtje8VkD";
```

where the hash string used here is the same one corresponding to the PIN number for the document (5243 in our example).

AA/AR only reads `config.js` one time when it opens, so `Calc2Spr21` will not be known until the next time AA/AR is opened. The location of the `config.js` file is described in general in the file `install_jsfiles.pdf`, found in the `doc` folder of this distribution.

Once the above line is placed in the `config.js` file, the instructor does not enter the PIN number, pressing Correct immediately corrects the quiz.

5. Turning the PIN security feature on and off

The security PIN feature can be turned on and off through the following two commands.

`\usePINCorrBtn`	(Turn on PIN security)
`\restoreCorrBtn`	(Turn off PIN security)

These can placed anywhere outside a `quiz` environment. The commands take effect beginning at the next quiz in the document.

Part II: Version 2.0

6. Introduction

Version 2 of this package provides additional security options for the document author (instructor or professor).

The source files (TEX) for the working examples in this part of the manual may be found on the AcroTeX Blog web site:

- eq-pin2corr: PIN security with warning and freezing[2]
- eq-pin2corr: PIN security on Begin Quiz and tracking retakes of a quiz[3]

7. Security with warn and freeze on End Quiz

warn and freeze   When the student presses the End Quiz control, an alert dialog box opens which warns the user that quiz will be 'frozen' which means active form fields are made readonly, except for the Ans button. The user has a choice of responding Yes or No, in the later case, the student can continue with the quiz. If the student presses Yes, the quiz is frozen, all the student can do is to save the file and to send it to the instructor.

```
\useWarnEndQuiz % use with \usePINCorrBtn
\restoreEndQuiz
\flJSStr{\EndQuizG@te@Msg}{Warning:
Are you sure you want to end this quiz?\r\r
The quiz will be frozen and no more changes will be allowed.
Click 'Yes' to end the quiz or 'No' to continue working on the quiz.}
```

Expand \useWarnEndQuiz prior to the quiz for which the 'warn and freeze' security is to be employed. After the quiz, optionally expand \restoreEndQuiz to its original definition.

Prior to the following quiz, \usePINCorrBtn\useWarnEndQuiz are expanded.

Solve each, passing is 100%.

1. The sum of 1 and 1 is …

  0   1   2   3   4

2. $\cos(\pi) =$

3. $\dfrac{d}{dx}\sin(x) =$

(PIN: 5243)

Answers:

---

[2] http://www.acrotex.net/blog/?p=1516
[3] http://www.acrotex.net/blog/?p=1519

8. PIN security with freeze on Correct

In the previous section, the quiz is made readonly (frozen) when the student presses the End Quiz control. We can also freeze the quiz when the Correct control is pressed.

This strategy allows the student to take and retake the quiz if the score is not to his liking; assuming the `showscore` option is in effect. Freezing the quiz on the Correct control allows the quiz to be marked up and returned to the student, without fear the student will later modify his answers and complain to the teacher that the score is incorrect.

To freeze the quiz when the Correct control is pressed, expand \FreezeThisQuiz.

`\FreezeThisQuiz`
`\FreezeThisQuizNot`

(use with \usePINCorrBtn)

There are two methods of expanding \FreezeThisQuiz: (1) expand prior to the quiz (and expand \FreezeThisQuizNot following the quiz); or (2) pass \FreezeThisQuiz through the optional argument of the \CorrButton (the Correct control) using the syntax \CorrButton[\cmd{\FreezeThisQuiz}]{\currQuiz}. The latter method make the change local, hence \FreezeThisQuizNot is not needed following the quiz.

The following quiz uses \CorrButton[\cmd{\FreezeThisQuiz}]{\currQuiz} at the end of the quiz. Prior to this quiz, \showScoreOn and \restoreEndQuiz are expanded, the later to recover from 'warn and freeze' of the previous quiz.

Solve each, passing is 100%.

1. The sum of 1 and 1 is …

$\qquad$ 0 $\qquad$ 1 $\qquad$ 2 $\qquad$ 3 $\qquad$ 4

2. $\cos(\pi) =$

3. $\dfrac{d}{dx}\sin(x) =$

(PIN: 5243)

Answers:

9. Tallying the number of retakes of a quiz

One problem with digital PDF quizzes (à la AcroTeX) is a student takes and retakes a quiz until a desired score of 100% is attained, assuming no PIN security is on the Correct control. This problem is partially mitigated by the PIN security on the Correct control, but still, teachers, when administering an exam for credit, do not like to see student retaking the quiz multiple times. The eq-pin2corr package now offers the following commands:

```
\useBeginQuizCnt          (optionally, use with \usePINCorrBtn)
\restorBeginQuiz
\qzResetTally[⟨options⟩]
```

Expanding \useBeginQuizCnt modifies the action of the Begin Quiz control to count the number of times the student as retaken the same quiz. The count show up in the readonly text field created by \qzResetTally. Restore the original action of the Begin Quiz control by expanding \restorBeginQuiz.

This next quiz has PIN security with freeze on the Correct control. The Begin Quiz controls tracks the number of times the student *re-takes the quiz.* Prior to the quiz we expand \showScoreOn\useBeginQuizPIN\useBeginQuizCnt. The \qzResetTally field is place to the right of the \CorrButton command. Speaking of the \CorrButton, \cmd{\FreezeThisQuiz} is passed to this command through its optional argument so that the quiz is frozen when the instructor presses the Correct control and successfully enters the PIN.

Instructions: Take and retake this quiz until you obtain 100%.

1. The sum of 1 and 1 is …

　　　0　　　　1　　　　2　　　　3　　　　4

2. $\cos(\pi) =$

3. $\dfrac{d}{dx}\sin(x) =$

　　　　　　　　　　　　　　　　　(PIN: 5243)

Answers:

10. Setting the maximum number of retakes

This quiz has PIN security on the Correct control. The Begin Quiz controls tracks the number of times the student *re-takes the quiz*. It also sets the maximum number of times the student retake the quiz.

```
\setMaxRetakes{⟨qz-name⟩}{⟨num⟩}
\nMaxRetakes{⟨qz-name⟩}
```

When declared prior to the quiz whose name is ⟨qz-name⟩, \setMaxRetakes sets the maximum of times a student can *retake the same quiz* to ⟨num⟩, where ⟨num⟩ is a non-negative integer. Declaring \setMaxRetakes{⟨qz-name⟩}{0} means the student may only take the quiz once (no retakes allowed); \setMaxRetakes{⟨qz-name⟩}{2} means he can retake the quiz twice (for a total of three times).

The command \nMaxRetakes{⟨qz-name⟩} is a way of typesetting the number ⟨num⟩ into the document as part of the instructions for the quiz, for example.

The following quiz has PIN security with freeze under the Correct control and it allows the student to retake the quiz at most 2 times.

Solve each, passing is 100%. Be aware that you will be allowed to *retake* this quiz at most 2 times.

1. The sum of 1 and 1 is …

　　　0　　　　　1　　　　　2　　　　　3　　　　　4

2. $\cos(\pi) =$

3. $\dfrac{d}{dx}\sin(x) =$

(PIN: 5243)

Answers:

11. PIN Security for Begin Quiz and Correct

For this final example, the student can see his quiz score (which can be optionally changed by expanding \showScoreOff); however, to retake the quiz a PIN must be entered when the Begin Quiz button is pressed. The PIN under the Begin Quiz control is (usually) different from the PIN under the Correct button.

```
\declRePINId{⟨pin-num⟩}{⟨hash-string⟩}
\numRePINId
\flJSStr{\BeginQuizG@te@Msgi}{Enter the PIN number
to retake this quiz}
\flJSStr{\BeginQuizG@te@Msgii}{Press the Begin Quiz
control to begin the quiz again}
```

\declRePINId is used to declare the PIN number for retaking the quiz, as well as the corresponding hash string. Refer to \declPINId above (Section 4) for information of how to acquire the hash string for the PIN.

The command \numREPINId expands to ⟨*pin-num*⟩ is not normally typeset into the document, but is used for documentation or demonstration purposes, such as in this document.

The final two, \BeginQuizG@te@Msgi and \BeginQuizG@te@Msgii expand to the messages the respondent reads. These may be redefined as desired.

The PIN for the Begin Quiz button is 1234. The Begin Quiz button does not need a PIN for the first time it is pressed. It requires a PIN after the first press.

Solve each, passing is 100%.

1. The sum of 1 and 1 is …

　　　0　　　　　1　　　　　2　　　　　3　　　　　4

2. $\cos(\pi) =$

3. $\dfrac{d}{dx}\sin(x) =$

(PIN: 5243)

Answers:

This kind of security is best when the students are taking a quiz in a computer lab with a proctor in the room. The student can ask the proctor to reset the quiz. Note that we keep a tally on the number of requests as the proctor may not write it down or remember.

Now, I really must get back to retirement. $\mathcal{DS}$