PageFocus: Using paradata to detect and prevent cheating on online achievement tests

Birk Diedenhofen¹ · Jochen Musch¹

Published online: 29 August 2016
© Psychonomic Society, Inc. 2016

Abstract Cheating threatens the validity of unproctored online achievement tests. To address this problem, we developed PageFocus, a JavaScript that detects when participants abandon test pages by switching to another window or browser tab. In a first study, we aimed at testing whether PageFocus could detect and prevent cheating. We asked 115 lab and 186 online participants to complete a knowledge test comprising items that were difficult to answer but easy to look up on the Internet. Half of the participants were invited to look up the solutions, which significantly increased their test scores. The PageFocus script detected test takers who abandoned the test page with very high sensitivity and specificity, and successfully reduced cheating by generating a popup message that asked participants not to cheat. In a second study, 510 online participants completed a knowledge test comprising items that could easily be looked up and a reasoning task involving matrices that were impossible to look up. In a first group, a performance-related monetary reward was promised to the top scorers; in a second group, participants took part in a lottery that provided performance-unrelated rewards; and in a third group, no incentive was offered. PageFocus revealed that participants cheated more when performance-related incentives were offered. As expected, however, this effect was limited to items that could easily be looked up. We recommend that PageFocus be routinely employed to detect and prevent cheating on online achievement tests.

Electronic supplementary material The online version of this article (doi:10.3758/s13428-016-0800-7) contains supplementary material, which is available to authorized users.

✉ Birk Diedenhofen
birk.diedenhofen@uni-duesseldorf.de

¹ Department of Experimental Psychology, University of Düsseldorf, Universitätsstrasse 1, Building 23.03, 40225 Düsseldorf, Germany

Keywords Paradata · Cheating detection · Achievement tests · Unproctored online testing · Data quality · JavaScript

Conducting research online has become increasingly popular as an inexpensive and efficient way to collect responses from people all over the world (Couper, 2008; Reips, 2008). Potential domains for online assessment include both surveys and online ability tests. Typically, online tests are unproctored; that is, no human proctor supervises the test session. Unproctored Internet testing, however, has been criticized for a lack of control over the testing process (e.g., Tippins et al., 2006). One major criticism is that deliberate cheating can distort the results of online ability tests. Some test takers may simply look up the solutions on the Internet, but such behavior may depend on the format of the online test. Cheating threatens the validity of performance tests, especially if the stakes are high, and may reduce the benefits of Internet-based testing. In an online survey by Jensen and Thomsen (2014), about 22 % of the participants indicated that they had used the Internet to identify the correct solution to at least one of the four political knowledge questions that had been presented. Looking up solutions on the Internet—the largest collection of human knowledge ever created—is being made easy by online encyclopedias such as Wikipedia and search engines such as Google. For this reason, “tests without a proctor often come without knowledge questions because the answers can be looked up quickly” (Kersting & Wottawa, 2014, p. 39). The problem with look-ups is further aggravated by test security issues, which are inherent to online assessment. The Internet is a copying machine, and test administrators ultimately cannot prevent participants from copying test material (e.g., by capturing the screen). However, copied test material available online makes it even easier for test takers to look up solutions. The problem of cheating is thus made even worse.

There is strong evidence that cheating is indeed a problem for unproctored Internet testing. Harmon and Lambrinos (2008) found that the grade point average of their student participants

predicted the students' results on an exam better if the exam was proctored rather than unproctored. Arguably, the predictive power of the grade point average was lowered when participants cheated (Harmon & Lambrinos, 2008). In a study by Carstairs and Myers (2009), scores on a high-stakes test were inflated if the test was administered online without a proctor rather than under formal supervised examination conditions. Tippins et al. (2006) argued that unproctored Internet tests may be acceptable in low-stakes contexts but should not be used as the sole source of evidence in high-stakes testing. In recent years, the focus of the debate has gradually shifted from discussing the feasibility of unproctored online achievement tests to investigations of how reliability and validity can best be protected in online assessment (Lievens & Burke, 2011).

Detecting and preventing cheating

Because cheating poses a considerable threat to the validity of unproctored Internet testing, various approaches have been suggested for detecting and preventing cheating. One approach that can be applied to detect cheating on online tests is the administration of an additional verification test that has to be taken offline under proctored conditions (Guo & Drasgow, 2010; Tendeiro, Meijer, Schakel, & Maij-de Meij, 2013). Test takers achieving a lower score on the verification test than on the preceding online test are suspected of cheating. A disadvantage of this method is that additional testing is necessary, and thus, the cost efficiency of unproctored Internet testing is thereby reduced. Furthermore, practice effects that may occur due to repeated measurements may make it difficult to identify cheaters because practice effects can help to increase the score on the verification test (Lievens & Burke, 2011; Nye, Do, Drasgow, & Fine, 2008). To discourage cheating in the first place, the International Test Commission (2006) recommended that test takers be informed in advance about planned follow-up verification tests. Some test administrators have also asked test takers to sign an honesty agreement before they take an online test (Lievens & Burke, 2011).

Another approach that can be used to identify cheaters is the use of person fit indices. Several item response models have been proposed to detect aberrant responding by comparing actual test responses with expected responses that are based on a test model (Karabatsos, 2003). If a test model that fits the response data can be identified, a person fit index can be calculated for each test taker. Persons whose response pattern deviates strongly from the test model (e.g., participants solving very difficult but failing relatively easy items) are suspected of cheating (e.g., Armstrong & Shi, 2009). A major problem of this approach is that person misfits are not necessarily indicative of cheating but may also be the result of lucky guesses or careless errors.

An alternative to unproctored offline testing that shares many of its advantages while offering more control over the test situation is remotely proctored testing, an administration mode in

which a test administrator supervises the test session over the Internet, for example, via a webcam (Karim, Kaminsky, & Behrend, 2014). Karim et al. (2014) found that remote proctoring reduced cheating but also produced negative reactions in the test takers. An even more invasive countermeasure against cheating is a strong lockdown of the browser or the entire operating system by employing a monitoring program (Foster, 2009). In this way, the test administrator can supervise all activities on the test taker's computer and, if necessary, restrict illegitimate actions. However, people may consider the installation of a monitoring program that grants the test administrator full access to their computer a strong violation of their privacy. Moreover, remotely proctored testing is costly; the need for a proctor and special software and hardware reduces the desired efficiency of online testing.

Another popular approach that can be used to prevent test takers from cheating is speeded testing. Restricting the time a test taker has available to respond to an item may be an effective way to keep people from looking up solutions. Arthur, Glaze, Villado, and Taylor (2010) administered a speeded unproctored Internet test of cognitive ability under high-stakes and low-stakes conditions and found scant evidence for cheating. Nye et al. (2008) compared the results of a perceptual speed test that was administered in either an unproctored online environment or a proctored offline context. They found no difference between the results obtained in the two administration modes. However, the measurement of some constructs may be incompatible with time constraints, and the use of time limits may discriminate against people less skilled in operating a computer. A recent study by Jensen and Thomsen (2014) casts additional doubt on whether time restrictions are effective countermeasures against cheating on factual knowledge tests because they found that cheaters actually responded more quickly than noncheaters on a political knowledge test.

To summarize, the methods currently available to detect and prevent cheating on unproctored Internet tests are either limited to certain testing contexts or result in substantial and prohibitive additional costs.

Paradata

When conducting online studies, paradata are useful kinds of auxiliary information that may help to improve data quality (Couper, 2005). Paradata are data that participants generate in the process of answering test questions (Kreuter, 2013); they have already been used early on in Internet-based research for verification purposes (Reips, 2000; Schmidt, 1997). In online studies, paradata are recorded from either the server that delivers the survey pages (server-side paradata) or the participants' computer (client-side paradata; Heerwegh, 2003). Two types of paradata may be distinguished (Callegaro, 2013):

1. *Device-type paradata* contain all information that is available about the device a participant uses to take part in an

online study. This includes, for example, the device model (e.g., desktop, tablet, or smartphone), operating system, screen resolution, browser, and available browser plugins. IP addresses are also device-type paradata that can be used to preclude multiple participations (Aust, Diedenhofen, Ullrich, & Musch, 2013) or to estimate the geographic location of a participant (Rand, 2012).

2. *Questionnaire navigation paradata* provide more detailed information about the participant's response process. Paradata of this category include, for example, mouse clicks, mouse movements, keystrokes, nonresponses, response times, and changes in input elements. For analysis, such actions and events may be aggregated at the respondent, page, or survey level (Kaczmirek, 2008).

Paradata augment the information obtained from the participants' responses and may help to shed light on the cognitive response process (Olson & Parkhurst, 2013). For example, Kieslich and Hilbig (2014) used the curvature of mouse movements as an indicator of cognitive conflict in social dilemmas. In a similar vein, Heerwegh (2003) reported that respondents holding unstable attitudes needed more time to respond to opinion questions than respondents with stable attitudes. Less knowledgeable respondents also required more time to answer knowledge questions and tended to change their answers more often than knowledgeable respondents (Heerwegh, 2003).

Paradata have also proven to be useful for assessing measurement error and for identifying and eliminating sources of error, thereby improving data quality (Yan & Olson, 2013). In an online survey, Malhotra (2008) found short completion times of respondents with a low level of education to be an indicator of satisficing behavior, resulting in poor data quality. Stern (2008) investigated whether the visual layout of survey questions influenced if and how participants changed their initial answers on a Web survey. By analyzing answer changes, he could identify the question layouts that were most prone to errors. Stieger and Reips (2010) also provided evidence that paradata may be linked to data quality. They found that participants showing behavior that is potentially detrimental to data quality (e.g., longer periods of inactivity and excessive mouse movements) exhibited a larger number of incorrect demographic entries.

Technically, the test page is always the active window that has the so-called *focus* when a test taker responds to an online test. This focus determines which window the keyboard input is being sent to. Registering whether or not a survey or test page is currently the page of focus has been mentioned as an additional type of paradata in Olson and Parkhurst (2013) and Callegaro (2013). Although using the focus offers an interesting approach for combating cheating, its usefulness for improving the validity of online testing has not yet been investigated. In this article, we therefore introduce PageFocus, a paradata tool aimed at detecting and possibly preventing cheating on unproctored Internet tests.

PageFocus

PageFocus is a JavaScript that can be added to test pages to register page-focusing events. If a test taker switches to an interface element outside of the test page (e.g., another window or browser tab), the previous test page loses the focus. If the PageFocus script has been added to a webpage, the script runs in the background, and it detects page-focus losses and registers them as *defocusing events*. When the test taker switches back to the original test page, PageFocus detects that the test page has regained the focus and registers a *refocusing event*. Thus, the PageFocus script captures when and how frequently defocusing and refocusing events occur on any given webpage. These data can be used to determine the length of time for which a test taker has abandoned a test page by switching to another window or browser tab. The PageFocus script is based on JavaScript events that are triggered when a website loses or gains the test taker's focus, respectively. Because PageFocus runs within the JavaScript sandbox, it cannot identify what a test taker does outside of the test page. Thus, the use of PageFocus does not violate the test taker's privacy. PageFocus is easy to implement and needs only JavaScript to run, a requirement that is met by 99 % of Web users (Kaczmirek, 2008, p. 87). We developed PageFocus to be compatible with all popular browsers including older as well as the most recent versions of Firefox, Chrome, Safari, and Internet Explorer. A complete list of compatible browsers is given in Table 4 in the [Appendix](#). If an incompatible browser is used, the PageFocus script will most likely detect no page-focusing events, but will not interfere with the experiment. Test takers using an incompatible browser may however be prompted to switch to a compatible browser. In the [supplementary material](#) for this article, the latest version 1.3 of PageFocus is included in an HTML page demonstrating the functionality of the script. PageFocus can also be obtained from GitHub (<https://github.com/deboerk/PageFocus/>).

Study 1

In Study 1, we conducted an experiment to validate PageFocus in which participants completed a general knowledge test. We aimed at testing whether the PageFocus script could be used (a) to reveal when participants looked up a solution by detecting page-focusing events and (b) to prevent look-ups by presenting a popup warning whenever a test taker abandoned the test page. Our knowledge test consisted of 16 multiple-choice questions that were difficult but could be answered easily by consulting the Internet. We experimentally manipulated the instructions that we presented to the test takers. In the experimental condition, participants were invited to simulate cheating by looking up the correct answer on the Internet if they were unable to identify the solution. In the control condition, participants were simply instructed to choose the most plausible answer. We

expected participants who were invited to cheat to look up solutions more frequently, and we expected PageFocus to detect more page-defocusing events for these participants as well. We also expected cheaters to obtain higher scores. To test whether the PageFocus script could also be used to prevent cheating, we presented a popup warning on the second half of the test whenever a participant caused a page-defocusing event by switching to another window or browser tab. This popup kindly asked participants not to look up the solution. We expected the popup warning to reduce cheating and to reduce the number of defocusing events on the second half of the test. This effect was predicted to be stronger in the group that was invited to cheat; no such effect was expected for the participants in the control group who were not invited to cheat. As a result of a reduced tendency to cheat, lower test scores were expected on the second half of the test. Again, we expected this decrease to be larger for the cheating group that we predicted would look up solutions more frequently on the first half of the test than on the second half during which a popup warning was presented.

Although PageFocus is meant to be used for online testing on the Web, we first conducted an unproctored validation test in the lab to maximize control over the test session. Testing in the lab allowed us to use the participants' behavior on the operating system level as a gold standard for detecting cheating that could not have been obtained on the Web. By capturing operating system data, including keyboard inputs, clipboard content, and the title of the active window, we were able to record when participants switched browser tabs or windows and to determine whether they looked up solutions on the Internet. In addition, we asked the participants to indicate all questions for which they had looked up the solution and to name the sources they had consulted for this purpose. We used both the operating system data and the participants' self-reported cheating behavior as external validation criteria to determine the sensitivity and specificity of detecting cheating on the basis of the PageFocus script. To test whether the results of the unproctored lab sample would generalize to an unproctored Web assessment, we conducted a parallel lab and Web validation study. A close agreement between the result pattern in the lab sample and the Web sample would lend support to the notion that PageFocus can be used to detect and prevent cheating not only in the lab but also on unproctored Internet tests.

Method

Design Participants completed a general knowledge test consisting of 16 difficult items. The test was divided into two halves, each consisting of eight items that were presented in a random order. The experiment had a 2×2 mixed factorial design with the between-subjects factor *instructions* (cheating vs. control) and the within-subjects factor *popup warning* (no warning vs. warning). Participants were randomly assigned to one of the two instruction groups. Before taking the knowledge test, participants in the control group were asked to choose the most plausible

answer if they did not know the solution to a question. Participants in the cheating group were instructed to cheat by looking up the correct solution if necessary. On the second half of the test, a popup warning asked participants in both conditions not to look up questions whenever PageFocus registered that a participant had triggered a page-defocusing event by switching to another window or browser tab. This popup warning was worded as a kind instruction so as not to provoke any feelings of reactance. Participants in the lab and on the Web received exactly the same webpages.

The number of page-defocusing events registered by PageFocus and the test score achieved by the participants—calculated as the number of correctly answered items—served as dependent variables. For both the lab and Web samples and separately for each question, we collected participants' self-reports on whether they had cheated. These self-reports were used as a first external criterion to determine the sensitivity and specificity of detecting cheating on the basis of PageFocus. In the lab, we additionally recorded the participants' behavior on the level of the operating system by capturing the following data throughout the experiment: keyboard input, clipboard content, and the title of the window that was currently active (i.e., the focus page). The window title indicated the application or webpage to which the user had switched because it contained the name of the application or the webpage displayed in the browser (e.g., "Google search") or even the search term used (e.g., "search term – Google search"). Because the titles of all browser windows that presented a test item were known, capturing the title of the currently active window once every second allowed us to detect whenever participants switched from the test page to another window or browser tab. For participants in the lab sample, the window-switching behavior caught at the level of the operating system served as the criterion that would be used to determine how accurately PageFocus was able to detect page-focusing events. In addition to the test takers' self-reports, operating-system-level data were also used as a second external criterion in the lab sample to calculate the sensitivity and specificity of PageFocus as a cheating detection device. Cheating was documented for a particular question when the keyboard input, clipboard content, or window title contained a keyword from the knowledge question the participant was currently answering. To account for mistyping, cheating was also documented when a Levenshtein distance (Levenshtein, 1966) from a keyword of less than 2 was observed in a word-by-word comparison that disregarded capitalization. For all 16 items, the questions, the answer options, and the keywords that were used to detect cheating are provided in Table 5 in the [Appendix](#). To confirm the functionality of our technical implementation, screenshots were taken once every second for all participants in the lab sample.

Material The general knowledge test consisted of 16 multiple-choice questions covering various knowledge domains (e.g., history, literature, linguistics, mathematics, geography; see Table 5 in the [Appendix](#)). Four answer options were presented

for each question. The test was split into two halves—Item Sets A and B—containing eight items each. Because we expected participants to cheat only if they were unable to identify the correct answer, we used very difficult questions that could, however, be easily looked up on the Web. For example, one question read “How long is a nautical mile?” All questions contained at least one keyword or phrase that could easily be identified and copied or typed into a Web search to obtain the correct solution (“nautical mile” in the above example). We made sure that searching for the keywords using the Google search engine, which currently has the greatest market share in Germany (95 %; Statista, 2015), always returned the correct answer within its first three search results. For all questions, participants could obtain the correct answer directly from the page preview of the Google search results; it was not even necessary to open the webpages linked to the search result page.

Procedure Both in the lab and on the Web, the experiment was conducted using Unipark EFS Survey 9.1 software (QuestBack, 2013). Participants in the lab were provided with the Chrome browser (version 31) to access the study. Test takers were seated in cubicles that did not allow anybody but themselves to see the computer screen. Prior to taking the general knowledge test, participants were asked to report their age, gender, and first language. Next, participants were invited to answer the 16 difficult multiple-choice general knowledge questions. To maximize their score, participants in the cheating group were asked to look up the questions they were not able to solve in a separate browser tab. Participants in the control group were simply instructed to choose the most plausible answer option. As a manipulation check, we asked participants to indicate what they were expected to do if they did not know the answer to a question. The two available answer options were “choose the most plausible answer” and “look up the solution on the Internet.” The two halves of the test were administered in a random order with each item presented on a separate page. The order of items on a test half and the order of answer options for each item were randomized. On the second half of the test (i.e., after the 8th item had been presented), a popup warning started to appear whenever a page-defocusing event was triggered and asked the participants to refrain from looking up the solutions. The popup warning was an overlay implemented in HTML/CSS that covered the test page; as such, it could not be blocked by popup blockers. To close the popup warning and to continue with the test, participants had to click on an “OK” button. After completing the knowledge test and in the same order in which the questions had previously been presented, participants were then asked, for each question separately, whether they had looked up the solution and to indicate the source they had consulted for each look-up. Participants were then provided with feedback on their test performance, debriefed, and thanked for their participation.

Sample A total of 127 participants completed the knowledge test in the lab. Participants in the lab sample who indicated a first

language other than German ($n = 12$) were excluded from the analysis. The resulting lab sample consisted of 115 psychology students (81 % female) with a mean age of 23 years ($SD = 5$). The random assignment of participants to the two instruction conditions placed 55 participants in the cheating group and 60 participants in the control group.

Participants were recruited for the Web sample by e-mail invitations that were sent to the members of an online panel consisting of the participants of unrelated previous studies conducted by the Department of Experimental Psychology at the University of Düsseldorf. None of the members of the panel had previously taken a test containing any of the materials used in the present investigation. Participants in the Web sample who took part repeatedly from the same IP address ($n = 6$), did not complete the knowledge test ($n = 13$), failed the manipulation check ($n = 1$), or indicated a first language other than German ($n = 1$) were excluded from the analysis. Data from 8 participants had to be discarded because they used a Safari browser for mobile devices that was incompatible with the PageFocus script¹ due to its use of nonstandard JavaScript events. In total, data from 186 participants (46 % female) were available for analysis in the Web sample. The participants' mean age was 33 years ($SD = 12$). In the Web sample, the random assignment placed 91 participants in the cheating group and 95 participants in the control group.

Results

Participants achieved 4.63 points ($SD = 2.27$) for Item Set A and 4.67 points ($SD = 2.23$) for Item Set B, on average. The internal consistencies of the items (Cronbach's alpha) were $\alpha = .62$ in the lab sample and $\alpha = .75$ in the Web sample. For all 16 items, the item difficulty and discriminatory power are shown in Table 6 of the Appendix. Because the average difficulties of the items in Sets A and B did not differ in the lab sample [$M_A = 0.54$ vs. $M_B = 0.59$, $t(14) = -0.70$, $p = .493$, $d = 0.35$] or in the Web sample [$M_A = 0.60$ vs. $M_B = 0.58$, $t(14) = 0.37$, $p = .718$, $d = 0.18$], the answers to both item sets were collapsed for all of the following analyses. Tables 1 and 2 display the numbers of participants triggering a page-defocusing event and reporting a look-up for a question.

In the lab, we used timestamps to match the page-focusing events caught by PageFocus with the participants' window-switching behavior captured on the level of the operating system. We were thus able to determine the accuracy with which PageFocus detected the participants' page-focusing behavior. In 1,264 cases (99.22 %), the matching was successful. In ten cases (0.78 %), the delay between the page-defocusing event and the page-refocusing event was below 1 s, and was therefore not detected by our tracking system, which captured the title of the

¹ For future investigations, we developed an updated version of the PageFocus script that is now also compatible with mobile Safari browsers.

Table 1 Numbers of participants in the lab sample ($n = 115$) and the Web sample ($n = 186$) triggering either no or at least one page-defocusing event during the knowledge test, under either the cheating or control instructions

Instructions		Sample			
		Lab		Web	
Cheating	At least one page-defocusing event	54	(98.18 %)	84	(92.31 %)
	No page-defocusing event	1	(1.82 %)	7	(7.69 %)
	Σ	55	(100.00 %)	91	(100.00 %)
Control	At least one page-defocusing event	2	(3.33 %)	18	(18.95 %)
	No page-defocusing event	58	(96.67 %)	77	(81.05 %)
	Σ	60	(100.00 %)	95	(100.00 %)

active window at 1-s intervals because we expected that no cheating would be possible in such a short period of time. This conjecture was confirmed when we analyzed the keyboard inputs, clipboard content, and window titles to determine the minimum time a test taker needed to cheat on a question. Participants always needed at least 3 s to copy and paste a keyword from the question to a search engine, to wait for and look up the solution, and to return to the original test page. It is therefore plausible to assume that no cheating went undetected by the tracking that we implemented at the level of the operating system.

Separately for the Web and lab samples, Fig. 1 shows the number of defocusing events that participants engaged in as a function of the instructions they received while completing both halves of the general knowledge test. For all analyses, the significance level was set at .05. We calculated an analysis of variance (ANOVA) with the number of page-defocusing events per participant as the dependent variable. As between-subjects independent variables, we used instructions (cheating vs. control) and setting (lab vs. Web). The presence of a popup warning (no warning vs. warning) served as an additional within-subjects independent variable.

As expected, in comparison with the control group, participants in the cheating group defocused considerably more often, $F(1, 297) = 651.49, p < .001$, generalized eta-squared (η_g^2) = .57. Confirming that our manipulation was successful, the presentation of a popup warning on the second half of the test successfully decreased the total

number of defocusing events as compared with the first half of the test, $F(1, 297) = 562.45, p < .001, \eta_g^2 = .43$. This decrease was larger for the cheating than for the control group, as indicated by a significant interaction between popup warning and instruction condition, $F(1, 297) = 505.85, p < .001, \eta_g^2 = .41$. Because the popup warning appeared only when a participant triggered a defocusing event, the earliest a warning could occur was after a test taker had looked up the correct answer to the first item on the second half of the test (i.e., after the ninth item). Thus, the popup warning could affect the participants' behavior from the tenth item onward (see Fig. 2). For this reason, a difference in the numbers of defocusing events between the cheating group and the control group remained when the popup warning was presented on the second half of the test, both in the lab, $t(113) = -9.22, p < .001, d = 1.72$, and on the Web, $t(184) = -6.09, p < .001, d = 0.89$. There was also a significant two-way interaction between setting and popup warning, $F(1, 297) = 11.25, p < .001, \eta_g^2 = .02$, which was qualified by a significant three-way interaction between setting, popup warning, and instructions, $F(1, 297) = 8.50, p = .004, \eta_g^2 = .01$. The reason for this three-way interaction was that on the first half of the test, in the no-warning condition, participants in the lab more readily followed the instructions to cheat than did participants on the Web, as reflected by a significantly enhanced number of defocusing events in the lab sample, $t(144) = 2.81, p = .006, d = 0.48$.

Separately for the lab and Web samples, Fig. 3 displays the test scores that participants achieved on the two halves of the general

Table 2 Numbers of participants in the lab sample ($n = 115$) and the Web sample ($n = 186$) who either denied cheating or reported that they had cheated on at least one question, under either the cheating or control instructions

Instructions		Sample			
		Lab		Web	
Cheating	Self-reported cheating	54	(98.18 %)	86	(94.51 %)
	No self-reported cheating	1	(1.82 %)	5	(5.49 %)
	Σ	55	(100.00 %)	91	(100.00 %)
Control	Self-reported cheating	2	(3.33 %)	3	(3.16 %)
	No self-reported cheating	58	(96.67 %)	92	(96.84 %)
	Σ	60	(100.00 %)	95	(100.00 %)

Fig. 1 Numbers of page-defocusing events per participant in the lab sample and the Web sample for the two instruction groups (cheating vs. control) on both test halves, which consisted of eight items each. On the

second half of the test, a popup warning asked participants to refrain from looking up solutions whenever a participant engaged in a page-defocusing event

knowledge test for both instruction groups, with and without a popup warning. We calculated an ANOVA with the participants' test scores as the dependent variable and instructions (cheating vs. control), setting (lab vs. Web), and popup warning (no warning vs. warning) as independent variables. As expected, participants who were invited to cheat achieved higher test scores than did the participants in the control group, $F(1, 297) = 305.37, p < .001, \eta_g^2 = .38$. Participants also scored higher on the first half of the general knowledge test, when no popup warning was presented, than on the second half, in which the popup warning appeared, $F(1, 297) = 225.31, p < .001, \eta_g^2 = .24$. As indicated by a significant instruction $\times$ popup warning interaction, and consistent with the prediction, participants received the highest scores when they were invited to

cheat and no popup warning was presented, $F(1, 297) = 291.17, p < .001, \eta_g^2 = .29$. We also found an interaction between sample and popup warning, $F(1, 297) = 8.74, p = .003, \eta_g^2 = .01$, that was qualified by a significant three-way interaction between setting, popup warning, and instructions, $F(1, 297) = 6.51, p = .011, \eta_g^2 = .01$. This three-way interaction was again due to the higher compliance of participants in the lab sample; that is, the popup manipulation had a stronger effect in the lab than on the Web. When participants were first invited to cheat, the subsequent presentation of a popup warning asking participants to refrain from further cheating led to a somewhat stronger decrease in scores in the lab than in the Web sample, $t(144) = -3.23, p = .002, d = 0.55$.

To assess whether the page-defocusing events registered by PageFocus can justifiably be used as indicators of participants' cheating behavior, we determined the sensitivity and specificity of the defocusing events with regard to two different external validation criteria. The first external validation criterion was the test takers' self-report with regard to the questions that he or she had cheated on. This external criterion was available for both the lab and Web samples. With regard to the test takers' self-reports, the sensitivity and specificity of the page-defocusing events captured by PageFocus were 99.54 % and 97.29 % in the lab sample, and 96.64 % and 94.56 % in the Web sample, respectively. For the lab sample, we were also able to assess how well participants remembered the questions they had cheated on and how honest their respective self-reports were. With the operating system data as an external criterion, the participants' self-reports were characterized by a sensitivity and specificity of 92.77 % and 99.85 %, respectively. Thus, in almost all instances, participants correctly remembered the questions they had cheated on and indicated them truthfully.

The second external criterion that was used to determine the sensitivity and specificity of PageFocus was available only for the

Fig. 2 Numbers of page-defocusing events as a function of item position, collapsed across the lab and Web samples. On the second half of the test, a popup warning asked participants to refrain from looking up solutions whenever they triggered a page-defocusing event. A considerable number of defocusing events still occurred for the ninth item, because the earliest occasion for the popup warning to appear was after participants had already looked up the answer to that item

Fig. 3 Test scores achieved by participants in the lab and Web samples for the two instruction groups (cheating vs. control) on both halves of the test, consisting of eight items each. On the second half of the test, a popup

warning asked participants to refrain from looking up the solutions whenever a participant engaged in a page-defocusing event

lab sample. For this sample, we analyzed operating system data, including keyboard inputs, clipboard contents, and window titles to determine whether a participant had cheated by searching for a keyword. With regard to this second external criterion, the sensitivity and specificity of the page-defocusing events captured by PageFocus were 100.00 % and 99.71 %, respectively. Specificity was thus virtually perfect, with the sole exception of four cases (0.29 %) in which participants briefly switched to another browser tab without cheating on the question.

Discussion

In Study 1, in parallel lab and Web experiments, we validated PageFocus, a JavaScript capable of detecting whenever test takers abandoned a test page by switching to another window or browser tab. In our experiment, we found the number of page-defocusing events to be highly useful for identifying test takers who looked up the solutions to an unproctored online knowledge test. As expected, test takers engaged in more page-defocusing events when they were invited to cheat than when they were simply instructed to choose the most plausible answer.

On the second half of the test, a popup warning appeared whenever participants triggered a page-defocusing event. By simply asking participants not to cheat, this popup warning successfully reduced the number of page-defocusing events. The results show that participants stopped looking up solutions directly after a popup warning was first presented, thus demonstrating that PageFocus can be used not only to detect cheating but also to prevent cheating when combined with a warning message. In the lab sample, PageFocus indicated page-focusing events almost as reliably (99.22 %) as a change in the title of the active window caught on the level of the operating system that was used as the gold standard for comparison. Thus, PageFocus can be employed to detect very

accurately whether and when test takers switch to another window or browser tab.

When the participants' self-reports were used as the external criterion, cheating was detected with PageFocus at very high sensitivity and specificity rates of 99.54 % and 97.29 %, respectively, in the lab sample. Even in the Web sample, for which a number of additional reasons for the occurrence of page-defocusing events can easily be imagined, PageFocus still achieved very high sensitivity and specificity rates of 96.64 % and 94.56 %, respectively. Extremely high sensitivity (100.00 %) and specificity (99.71 %) rates for PageFocus as a cheating detection device were also established with regard to the second external criterion—namely operating system data, including keyboard inputs, clipboard contents, and window titles. Even though validation criteria at the level of the operating system were available only for the lab sample, the fact that we observed very similar patterns of results in the lab and Web samples strongly suggests that the favorable properties observed for PageFocus under controlled lab settings generalize to Web testing environments.

In the standardized lab setting, the hardware and software that were used by the participants did not vary. This unavoidably limits the generalizability of the results. The converging and largely parallel results in the lab and the Web samples suggest, however, that PageFocus also runs reliably in an environment with much more diverse setups consisting of a large number of different devices and configurations. An important criticism of Study 1, however, is that the participants were instructed to cheat and had not decided on their own to pursue a desirable outcome by engaging in dishonest behavior as is necessary to meet the usual definition of cheating. Therefore, to validate PageFocus in an unproctored Internet test including real cheating behavior, we conducted a second study in which we varied the incentive and the opportunity to cheat across several experimental groups.

Study 2

In applied contexts, cheating is difficult to observe without intruding on participants' privacy. Therefore, in Study 1, we simply asked participants to indicate whether they had engaged in cheating behavior. However, this is not an option in high-stakes situations in which participants will be inclined to provide untruthful reports to obtain a desired reward. To validate PageFocus in a more realistic setting, we implemented experimental conditions in Study 2 for which a specific result pattern would be indicative of real cheating. To this end, participants were asked to complete both (a) general knowledge questions that were easy to look up on the Internet and (b) reasoning tasks that were based on matrices that could not be solved by looking up the solution on the Internet (cf. Karim et al., 2014). We expected a higher number of defocusing events for the general knowledge questions than for the matrices. To manipulate the incentive to cheat, participants were offered no reward, a performance-unrelated reward (the chance to win a lottery), or a performance-related reward (which was available only to the top scorers). The performance-related reward was expected to provide a strong motivation to achieve a high score, and we therefore expected at least some of the participants in this condition to look up the solutions on the Internet. Because looking up the solutions gave such participants an unfair advantage over the more honest participants, this kind of behavior had to be considered dishonest and an attempt to cheat to gain an advantage.

We expected participants in the performance-related reward condition to engage in a larger number of defocusing events than participants in the performance-unrelated and no-reward conditions. In an additional attempt to further increase the motivation to cheat in order to obtain a better score, we announced to half of the participants that they would be provided with personalized feedback at the end of the study. The other half of the participants received no such announcement and were therefore potentially less motivated to perform well.

Because successful cheating should lead to better test performance, we expected participants to achieve higher test scores if they defocused more often. We also expected a positive correlation between the number of defocusing events and the score on the knowledge test in which cheating was easy to do. No such correlation was expected for the matrices test on which cheating was impossible. We also expected the correlation between defocusing behavior and test scores to be higher when a performance-related reward was offered rather than a performance-unrelated reward or no reward.

Method

Design Study 2 had a $3 \times 2 \times 2$ mixed factorial design with the between-subjects factors *reward* (none vs. performance-unrelated vs. performance-related) and *individualized feedback* (feedback announced vs. not announced) and the within-subjects

factor *item type* (knowledge questions vs. reasoning matrices). All participants completed both the test containing the knowledge questions and the test consisting of the reasoning matrices. The order of the tests was randomized. Participants were also randomly assigned to one of the three reward groups and to one of the two feedback groups. We experimentally manipulated the incentive to cheat by varying the type of reward across the three reward groups. In one group, there was no reward. In another group, as a performance-unrelated reward, participants were informed that they had the chance to win one of 40 vouchers worth €20 each in a lottery involving the 400 participants who were invited to participate in one of the three reward conditions (thus, a total of 1,200 participants were contacted and invited to participate in the study). In a third group, to offer a performance-related reward, we informed participants that the top 40 scorers out of the 400 invited participants would win a €20 voucher. Depending on the feedback group, performance feedback at the end of the study was either announced or not announced. We expected that participants would be more incentivized to cheat if they knew that they could thus improve the individual performance feedback they received at the end of the test. As dependent variables, participants' test scores were computed separately for the general knowledge questions and the reasoning matrices, and the number of page-focusing events they produced while completing these tests was captured. For both types of items, the test score was calculated as the number of correctly solved items.

Material We presented ten reasoning matrices from the Viennese Matrices Test 2 (Items 1, 2, 4, 7, 9, 13, 14, 15, 16, and 18; Formann, Waldherr, & Piswanger, 2011). The matrices were impossible to look up on the Internet because they were purely figural. Solution frequencies for the matrices according to the norms reported in the test manual ranged from very easy (.92) to very difficult (.39). Using data from 387 pretest participants, we selected ten knowledge questions with difficulties that matched the difficulties of the reasoning matrices. As in Study 1, the solutions to all knowledge questions could easily be looked up on the Internet. All items on the knowledge test are provided in Table 7 in the [Appendix](#).

Procedure At the beginning of the study, participants were welcomed and informed about the two tests and their reward condition. Depending on the feedback group, personal performance feedback at the end of the study was either announced or not. Next, participants were asked to indicate their gender, age, and first language. Participants were instructed to guess if they could not identify the correct solution to an item. After the participants finished both tests, they were provided with detailed performance feedback. Participants were debriefed and thanked, and participants in the two conditions who had the chance to win a voucher were informed that they would be contacted if they won.

Sample The recruiting was similar to Study 1. Participants were members of the same online panel used in Study 1 but had not been invited to participate in Study 1. None of the participants had previously taken a test containing any of the materials used in the present investigation. For each reward condition, 400 participants were invited, resulting in a total of 1,200 invitations. Participants who participated repeatedly using the same IP address ($n = 10$), did not finish the study ($n = 46$), reported a first language other than German ($n = 11$), or used an incompatible Safari browser for mobile devices ($n = 20$) were excluded from the analysis. In total, 510 online participants completed the study (59 % female). Participants had a mean age of 33 years ($SD = 13$). In all, 172, 166, and 172 participants completed the study in the no-reward, performance-unrelated-reward, and performance-related-reward conditions, respectively, and the numbers of participants completing the study in the two feedback conditions were 253 (feedback) and 257 (no feedback). The drop-out rates differed neither as a function of the reward conditions, $\chi^2(2) = 3.27, p = .195$, nor as a function of the feedback conditions, $\chi^2(1) = 0.75, p = .388$.

Results

Participants achieved an average of 7.25 out of 10 points ($SD = 1.90$) on the knowledge test, and 7.08 out of 10 points ($SD = 2.30$) on the reasoning test. The internal consistencies of the items (Cronbach's alpha) were $\alpha = .59$ for the knowledge test and $\alpha = .73$ for the reasoning test. For all items on the knowledge and reasoning tests, the item difficulty and discriminatory power are provided in Table 8 in the [Appendix](#).

The results showed that 32.55 % of the sample defocused at least once during the study. Participants who defocused at least once usually did so repeatedly—4.4 times, on average. To scrutinize whether there was a link between defocusing behavior and cheating, we conducted separate analyses for all experimental conditions.

To this end, we calculated an ANOVA with the number of page-defocusing events as the dependent variable and the between-subjects independent variables reward (none vs. performance-unrelated vs. performance-related) and individualized feedback (feedback announced vs. not announced). The item type (knowledge questions vs. reasoning matrices) was employed as a within-subjects independent variable. We found that participants produced more page-defocusing events on the knowledge test—for which they could look up the solutions—than on the reasoning test, $F(1, 504) = 17.46, p < .001, \eta_g^2 = .01$ (Fig. 4). As expected, the reward offered to the participants influenced the number of page-defocusing events, $F(2, 504) = 11.51, p < .001, \eta_g^2 = .03$. Post-hoc comparisons using Tukey's HSD test revealed that significantly more page-defocusing events were registered if there was a performance-related reward than with either a performance-unrelated reward ($p < .001$) or no reward ($p < .001$). We found no difference in the numbers of defocusing events between the performance-unrelated-reward and no-

Fig. 4 Average numbers of defocusing events that occurred while participants responded to the ten knowledge questions and the ten matrices tasks in the no-reward, performance-unrelated-reward, and performance-related-reward conditions. Error bars indicate standard errors

reward conditions ($p = .966$). The reward $\times$ item type interaction was significant, $F(2, 504) = 10.30, p < .001, \eta_g^2 = .02$. Post-hoc comparisons revealed significantly more page-defocusing events than in all other conditions when a performance-related reward was at stake on the knowledge test, for which it was possible to look up the solutions (all $ps < .001$).

The announcement of individualized performance feedback had no influence on the number of page-defocusing events, $F(1, 504) = 0.03, p = .866, \eta_g^2 = .00$. The feedback $\times$ reward interaction, $F(2, 504) = 0.39, p = .677, \eta_g^2 = .00$, the feedback $\times$ item type interaction, $F(1, 504) = 0.08, p = .774, \eta_g^2 = .00$, and the feedback $\times$ reward $\times$ item type interaction, $F(2, 504) = 0.38, p = .687, \eta_g^2 = .00$, were not significant, either. Because the feedback manipulation had no impact on defocusing behavior, it was not considered further in the following analyses.

To test whether participants benefited from frequent page-defocusing behavior, we correlated the participants' test scores on the two tests with the numbers of page-defocusing events that participants produced while completing the tests (Table 3). For comparisons between the dependent and independent correlations, we used the tests by Steiger (1980) and Fisher (1925), respectively, as implemented in the R package *cocor* (Diedenhofen & Musch, 2015). All tests were one-tailed in accordance with the hypothesis that the correlation between test scores and page-defocusing events should be higher for the knowledge test than for the reasoning test and with the hypothesis that test scores should be significantly more strongly correlated with the number of page-defocusing events in the performance-related-reward condition than in the other conditions. As expected, for the knowledge test and across all reward conditions, the correlation between test scores and the number of page-defocusing events ($r = .37$) was significantly higher than for the reasoning test ($r = .07; z = 4.95, p < .001$). The same result pattern was also found in separate analyses conducted for the participants in the no-

Table 3 Correlations between the test score and the number of defocusing events for the knowledge test and the reasoning test under the different reward conditions

Item type	Reward	<i>r</i>	<i>p</i>	<i>n</i>
Knowledge questions	None	.29	<.001	172
	Performance-unrelated	.28	<.001	166
	Performance-related	.46	<.001	172
	(All reward conditions)	.37	<.001	510
Logical matrices	None	.07	.368	172
	Performance-unrelated	−.03	.685	166
	Performance-related	.13	.089	172
	(All reward conditions)	.07	.102	510

r = correlation coefficient, *p* = *p* value, *n* = group size

reward condition ($r = .29$ vs. $.07$; $z = 2.07$, $p = .019$), in the performance-unrelated-reward condition ($r = .28$ vs. $-.03$; $z = 2.93$, $p = .002$), and in the performance-related-reward condition ($r = .46$ vs. $.13$; $z = 3.43$, $p < .001$). As expected, the correlation between test scores and the number of page-defocusing events was highest for the knowledge test when the reward was performance-related ($r = .46$ vs. $.29$; $z = 1.92$, $p = .027$).

For both the knowledge test and the reasoning test, we calculated the probability of a correct answer as a function of whether or not a defocusing event had occurred (Fig. 5). For the knowledge questions, the probability that an item was solved correctly was significantly higher when a defocusing event was registered (.95 vs. .70; $z = 5.08$, $p < .001$). On the reasoning test, whether defocusing occurred or not did not predict whether an item was solved (.67 vs. .71; $z = 0.62$, $p = .535$).

Fig. 5 Item success probabilities for the knowledge questions, which could be looked up on the Internet, and the reasoning matrices, which could not be looked up, as a function of whether a defocusing event had occurred. Error bars indicate standard errors

Discussion

In Study 2, we validated PageFocus using two online achievement tests—a knowledge test on which cheating was possible and a matrices test that made cheating impossible. We manipulated the incentive to cheat by offering a reward that was absent, performance-unrelated, or performance-related. We also varied whether performance feedback at the end of the study was announced or not.

As expected, defocusing events occurred more often for knowledge test items that could be easily looked up than for reasoning items that did not offer this possibility. Also as expected, more defocusing events were registered if a performance-related reward was offered in comparison with a performance-unrelated reward or no reward. Announcing individualized performance feedback, however, did not influence defocusing behavior. As was to be expected if cheaters were successful, test takers who defocused more frequently on the knowledge test also achieved higher test scores. No such relationship was observed for the reasoning test, on which cheating was not an option. This interaction with the type of item supports the conclusion that page-defocusing events are indicative of cheating behavior.

General discussion

Unproctored online achievement tests are on the rise in research and personnel selection contexts. Cheating is an important threat to the validity of such tests. PageFocus is a new JavaScript that can be used to detect and prevent cheating on unproctored Internet tests by registering whether test takers abandon a test page by switching to another window or browser tab. In Study 1, we validated PageFocus by using a knowledge test that we administered in a controlled lab environment and in an applied Web context. The number of defocusing events and test scores were highest when participants were instructed to cheat if necessary. On the second half of the test, a popup warning that was triggered whenever a participant left the test page successfully reduced cheating. In both the lab and Web samples, the overall result patterns for page-defocusing events and test scores across experimental conditions provided evidence for the functionality of PageFocus and for the validity of the paradata collected by the script. Using the participants' self-reports and the strong validation data captured on the level of the operating system as external criteria, we found a very high sensitivity and specificity of the detection of cheating with the PageFocus script. In Study 2, we further validated PageFocus by using an unproctored Internet test consisting of knowledge questions on which cheating was easy and a reasoning test involving matrices on which cheating was next to impossible. Across experimental conditions, we varied the reward that was offered. Rewards were absent, performance-unrelated, or performance-related. Both the test scores and the number of defocusing events were highest for the knowledge test, on which cheating was

possible, and when a performance-related reward was at stake. This overall result pattern confirmed the validity of the paradata collected by PageFocus.

When evaluating the data collected by PageFocus, our recommendation is to screen out all pairs of page-defocusing and refocusing events with a total duration of less than 3 s. This recommendation is based on our observation that browser-based cheating seems to be impossible to conduct in such a small amount of time. In Study 1, we found that participants needed at least 3 s to cheat on a question by looking up the solution. Using a cutoff criterion of 3 s also helps to avoid false alarms caused by system popups or unrelated applications running in the background. In the lab sample in Study 1, the rate of false alarms ($=1 - \text{specificity}$) was 0.29 % and 2.71 % using operating system data and the participants' self-reports as criteria, respectively. In the Web sample using self-reports as the criterion, the false alarm rate was 5.44 %. In the lab sample, the imperfect sensitivity (92.77 %) of the participants' self-report with operating system data as the external criterion revealed that the participants did not report all questions they actually cheated on. This means that some of the apparent false alarms produced by PageFocus were probably actually due to the inaccuracy of the participants' self-reports, and would in fact have been hits if a perfect external criterion would have been available. The screenshots that were captured on the lab computers further revealed that some of the false alarms were produced by participants who switched to another browser tab for reasons unrelated to cheating. Participants who were distracted by other activities are arguably also a likely reason for why the false alarm rate was higher in the Web sample than in the lab sample.

We further recommend that when test administrators use the PageFocus script, test takers should always first be instructed not to switch to other applications or browser tabs running in the background. To enforce such instructions, PageFocus can then be combined with a popup warning. In the present investigation, we opted to present an extremely gentle and kindly worded popup warning. In principle, however, the detection of a page-defocusing event could also be associated with more severe consequences for the test taker, for example, nullifying an answer or the entire test. An obvious limitation of PageFocus that should be noted is that the script cannot detect test takers who look up questions on another device (e.g., a smartphone). In addition to detecting cheaters, PageFocus may also be useful in other contexts. For example, Do (2009) pointed out that inconsistencies between the results of unproctored and proctored tests might not occur only as a result of cheating but may also be the result of distracted test takers. PageFocus can be used to detect when and for how long online participants are distracted during a study. Such distraction detection may be useful in various situations. In most online studies, participants are required to carefully read all instructions and to attentively complete any given tasks. If participants abandon a study, they are more likely to forget part of the instructions, and any interruptions may interfere with an experimental manipulation, for example, the induction of an emotional state (Göritz,

2007). PageFocus can help to identify distracted participants and can be used to examine whether participants' responses deviate from the rest of the sample. The information provided by PageFocus may be especially useful when presenting experimental stimuli in online research. If a researcher wants a participant in an online study to concentrate on a stimulus (e.g., a text, image, or video) for a defined period of time, page-defocusing events registered by PageFocus can be used to identify participants who were not fully engaged with the presented material.

On an achievement test, PageFocus cannot tell whether participants have abandoned a test page to cheat or because they were distracted and occupied with an unrelated activity. However, additional contextual information may be used to determine the actual reason for a page-defocusing event. For example, if test takers defocused on a difficult knowledge question, it is more likely that they cheated than if they defocused on a page that asked them to provide demographic information.

A general limitation of JavaScript is that client-side scripting languages are potentially more prone to incompatibilities than server-side scripts (Schwarz & Reips, 2001) because it cannot be ruled out entirely that some users are using a browser with an incomplete or erroneous implementation of the language. Recent studies have shown that JavaScript can be used successfully and with sufficient precision in online studies for collecting reaction-time data (Barnhoorn, Haasnoot, Bocanegra, & Steenbergen, 2015; Chetverikov & Upravitelev, 2015; de Leeuw & Motz, 2016). The PageFocus script is a useful supplement to such studies as it has the potential to improve the recording and interpretation of response times collected online. For example, when a response time outlier occurs, it is often difficult to tell whether the longer response time was the result of a particularly thorough reasoning process or whether participants simply abandoned the test page to pursue another activity, for example, to check their e-mail. In such cases, collecting information on the participants' page-defocusing behavior can be helpful for determining the amount of time a participant actually spent on the test page.

This and other potential applications of PageFocus should be investigated in future studies. Additional studies should also validate PageFocus in real personnel selection contexts in which high stakes may motivate test takers to cheat even more rigorously than in the present study. Comparisons of the ability to detect cheating using PageFocus with alternative measures that promise protection against cheating should also be conducted in future studies.

To summarize, the present studies demonstrated that page-focusing events are useful paradata that can be successfully captured to improve data quality in online testing. Our results show that the PageFocus script is a valid tool that has high sensitivity and specificity and can be used to reliably detect and prevent cheating on unproctored Internet tests. The PageFocus script is freely available as an [electronic supplement](#) to this article and on GitHub (<https://github.com/deboerk/PageFocus/>), and we recommend that test administrators and researchers routinely employ the script when administering Web-based performance tests.

Author note We thank Stefan Trost for his help in recording and collecting the lab data.

Appendix

Table 4 List of browsers that we tested and found to be compatible with PageFocus

Browser	Versions
Firefox	26, 45
Chrome	18, 31, 49
Internet Explorer	7, 8, 9, 10, 11
Safari	5.1, 9.1

Table 5 Item content and keywords for all 16 items from Item Sets A and B used in Study 1

Item	Knowledge questions	Keywords
A1	“Ditanyè” is the national anthem of <i>a) Burkina Faso</i> , b) Botswana, c) Benin, d) Brunei	Ditanyè, Ditanye
A2	„Ditanyè” ist die Nationalhymne von <i>a) Burkina Faso</i> , b) Botswana, c) Benin, d) Brunei	Ditanyè, Ditanye
A2	When did the Crimean War take place? <i>a) 1853–1856</i> , b) 1854–1857, c) 1855–1858, d) 1856–1859	Crimean War
A2	In welchem Zeitraum fand der Krimkrieg statt? <i>a) 1853–1856</i> , b) 1854–1857, c) 1855–1858, d) 1856–1859	Krimkrieg
A3	What is the “cosecant”? <i>a) a trigonometric function</i> , b) a Hungarian dance, c) a Russian liquor, d) a leader of the Cossack	Cosecant
A3	Was ist der „Kosekans”? <i>a) eine trigonometrische Funktion</i> , b) ein ungarischer Tanz, c) ein russischer Schnaps, d) ein Anführer der Kosaken	Kosekans
A4	What is a “Bilwiss”? <i>a) a goblin</i> , b) a know-it-all, c) a stone, d) a traitor	Bilwiss
A4	Was ist ein „Bilwiss”? <i>a) ein Kobold</i> , b) ein Besserwisser, c) ein Stein, d) ein Verräter	Bilwiss
A5	What is a “bitter orange”? <i>a) a citrus fruit</i> , b) a butterfly species, c) a fruit vermin, d) a bird	bitter orange
A5	Was ist eine „Pomeranze”? <i>a) eine Zitruspflanze</i> , b) eine Schmetterlingsart, c) ein Obstschädling, d) ein Vogel	Pomeranze
A6	How long is a nautical mile? <i>a) 1,852 m</i> , b) 1,659 m, c) 1,620 m, d) 1,735 m	Nautical mile
A6	Wie lang ist die internationale Seemeile? <i>a) 1852 m</i> , b) 1659 m, c) 1620 m, d) 1735 m	Seemeile
A7	“Onomatopoeia” designates? <i>a) Echoism</i> , b) a meter, c) Acoustics, d) Free dance	Onomatopoeia, Onomato
A7	„Onomatopoesie” bezeichnet? <i>a) Lautmalerei</i> , b) ein Versmaß, c) Kanglehre, d) Ausdruckstanz	Onomatopoesie, Onomato
A8	Who wrote the youth novel “Krabat”? <i>a) Otfried Preußler</i> , b) Michael Ende, c) James Krüss, d) Erich Kästner	Krabat
A8	Wer schrieb den Jugendroman „Krabat”? <i>a) Otfried Preußler</i> , b) Michael Ende, c) James Krüss, d) Erich Kästner	Krabat
B9	What are the “Rhodopes”? <i>a) Mountains</i> , b) an animal species, c) the native people of Rhodes, d) a religious sect	Rhodopes
B9	Was sind die „Rhodopen”? <i>a) ein Gebirge</i> , b) eine Tiergattung, c) die Ureinwohner von Rhodos, d) eine religiöse Sekte	Rhodopen
B10	What is a “Schwa”? <i>a) a sound</i> , b) a fish, c) a car, d) a dance move	Schwa
B10	Was ist ein „Schwa”? <i>a) ein Laut</i> , b) ein Fisch, c) ein Auto, d) eine Tanzbewegung	Schwa
B11	In which month was John F. Kennedy murdered? <i>a) November</i> , b) August, c) September, d) October	Kennedy, John, JFK
B11	In welchem Monat wurde John F. Kennedy ermordet? <i>a) November</i> , b) August, c) September, d) Oktober	Kennedy, John, JFK
B12	What does “the doctrine of the seven year cycle” refer to? <i>a) the development of humans in Waldorf pedagogics</i> , b) the periodic change in harvest yield, c) a calendar system developed by the ancient Egyptians, d) a medieval tax system	seven year cycle, teachings
B12	Worauf bezieht sich die „Lehre von den Jahrsiebten”? <i>a) auf die Entwicklung des Menschen in der Waldorfpädagogik</i> , b) auf die periodische Veränderung des Ernteertrags, c) auf ein Kalendersystem der alten Ägypter, d) auf ein mittelalterliches Besteuerungssystem	Jahrsiebten, Lehre
B13	What is a “groyne”? <i>a) an artificial dam for bank protection</i> , b) a female relative, c) an underground storage room in mining, d) a guide element in a turning machine	Groyne
B13	Was ist eine „Buhne”? <i>a) ein künstlicher Damm zum Uferschutz</i> , b) eine weibliche Verwandte, c) ein unterirdischer Lagerraum im Bergbau, d) ein Führungselement einer Drehmaschine	Buhne
B14	What does “EMDR” stand for? <i>a) a therapeutic procedure</i> , b) a circuit technology, c) a European agency, d) a measurement method	EMDR
B14	Was ist unter „EMDR” zu verstehen? <i>a) ein therapeutisches Verfahren</i> , b) eine Schaltungstechnik, c) eine europäische Behörde, d) eine Messmethode	EMDR
B15	What is the molecular formula of alcohol? <i>a) C2H6O</i> , b) C2H4O, c) C3H4O, d) C4H2O	Alcohol
B15	Wie lautet die Summenformel von Alkohol? <i>a) C2H6O</i> , b) C2H4O, c) C3H4O, d) C4H2O	Alkohol
B16	What is the name of Peter Parker’s alter ego? <i>a) Spider-Man</i> , b) Iron Man, c) Captain America, d) Silver Surfer	Peter, Parker, Alter Ego
B16	Wie heißt das Alter Ego von Peter Parker? <i>a) Spider-Man</i> , b) Iron Man, c) Captain America, d) Silver Surfer	Peter, Parker, Alter Ego

For each item, the first answer option is the correct solution. In the experiment, the order of answer options and the order of all items in the respective item set were randomized.

Table 6 Item difficulties and discriminatory power for all 16 items from Item Sets A and B used in Study 1, listed separately for the lab and Web samples

Item	Difficulty						Discriminatory Power	
	Lab			Web				
	All	Cheating	Control	All	Cheating	Control	Lab	Web
A1	.56	.69	.43	.54	.69	.39	.29	.27
A2	.53	.73	.35	.52	.75	.29	.36	.40
A3	.41	.64	.20	.48	.66	.32	.35	.41
A4	.50	.62	.40	.56	.73	.40	.28	.38
A5	.41	.56	.27	.66	.77	.55	.33	.27
A6	.43	.62	.25	.51	.68	.34	.36	.27
A7	.78	.84	.73	.77	.87	.68	.22	.28
A8	.70	.78	.62	.79	.87	.72	.09	.24
B9	.63	.76	.50	.61	.79	.44	.04	.36
B10	.75	.82	.68	.67	.84	.52	.07	.32
B11	.59	.75	.45	.49	.70	.28	.21	.41
B12	.39	.62	.18	.41	.66	.18	.28	.49
B13	.51	.71	.33	.57	.73	.42	.36	.40
B14	.44	.58	.32	.42	.67	.18	.14	.48
B15	.51	.71	.33	.59	.73	.45	.20	.29
B16	.93	.93	.93	.86	.89	.83	.13	.16
Mean	.57	.71	.44	.59	.75	.44	.23	.34

Item difficulties are listed for the cheating group (Cheating), the control group (Control), and across both groups (All). The item difficulty of an item was calculated as the proportion of test takers who correctly solved the item

Table 7 The ten items from the knowledge test used in Study 2

Item	Knowledge questions
1	Honey-wine is referred to as? a) <i>Mead</i> , b) Sling, c) Cider, d) Calvados Honigwein bezeichnet man als? a) <i>Met</i> , b) Sling, c) Cidre, d) Calvados
2	What is amnesia? a) <i>memory loss</i> , b) pardon, c) anemia, d) sleepwalking Was ist eine Amnesie? a) <i>Gedächtnisverlust</i> , b) Begnadigung, c) Blutarmut, d) Schlafwandeln
3	What is the Finnish name for Finland? a) <i>Suomi</i> , b) Finn-Dingi, c) Warääg, d) Firdausi Wie lautet der finnische Name für Finnland? a) <i>Suomi</i> , b) Finn-Dingi, c) Warääg, d) Firdausi
4	What is the currency of Thailand called? a) <i>Baht</i> , b) Kyat, c) Won, d) Rupie Wie nennt man die thailändische Währung? a) <i>Baht</i> , b) Kyat, c) Won, d) Rupie
5	Who murdered the hero of the Nibelungen legend Siegfried? a) <i>Hagen</i> , b) Brunhilde, c) Gunnar, d) Gottfried Wer ermordete den Nibelungenhelden Siegfried? a) <i>Hagen</i> , b) Brunhilde, c) Gunnar, d) Gottfried
6	In which country is the Aswan Dam located? a) <i>Egypt</i> , b) Kenya, c) Libya, d) Sudan In welchem Land befindet sich der Assuan-Staudamm? a) <i>Ägypten</i> , b) Kenia, c) Libyen, d) Sudan
7	Who or what is the blue Peter? a) <i>Signal flag</i> , b) Fish, c) Truant, d) Drunkard Wer oder was ist der blaue Peter? a) <i>Signalflagge</i> , b) Fisch, c) Schulschwänzer, d) Trunkenbold
8	Who was the founder of anthroposophy? a) <i>Rudolf Steiner</i> , b) Johann Heinrich Pestalozzi, c) Friedrich Fröbel, d) Maria Montessori Wer begründete die Anthroposophie? a) <i>Rudolf Steiner</i> , b) Johann Heinrich Pestalozzi, c) Friedrich Fröbel, d) Maria Montessori
9	What was Genghis Khan originally called? a) <i>Temüdschin</i> , b) Burte, c) Guattari, d) Guarani Wie lautet der ursprüngliche Name von Dschingis Khan? a) <i>Temüdschin</i> , b) Burte, c) Guattari, d) Guarani
10	Which color is rose madder? a) <i>red</i> , b) white, c) black, d) green Welche Farbe hat Krapplack? a) <i>rot</i> , b) weiß, c) schwarz, d) grün

For each item, the first answer option is the correct solution. In the experiment, the order of answer options and the order of items were randomized

Table 8 Item difficulties and discriminatory power for all ten items on the knowledge test and the reasoning test used in Study 2

		Difficulty				
		Reward				
Item type	Item	All	None	Performance-unrelated	Performance-related	Discriminatory power
Knowledge questions	1	.96	.96	.94	.99	.15
	2	.95	.94	.95	.96	.06
	3	.85	.87	.81	.86	.33
	4	.84	.83	.84	.83	.18
	5	.74	.76	.73	.73	.35
	6	.73	.76	.69	.75	.38
	7	.68	.65	.69	.69	.27
	8	.60	.56	.60	.64	.31
	9	.50	.47	.49	.52	.31
	10	.41	.38	.39	.45	.26
	Mean	.72	.72	.71	.74	.26
Reasoning matrices	1	.98	.98	.97	.98	.15
	2	.93	.92	.92	.94	.24
	3	.76	.74	.78	.76	.38
	4	.80	.83	.79	.78	.36
	5	.74	.71	.71	.79	.42
	6	.60	.59	.61	.62	.54
	7	.72	.73	.67	.74	.45
	8	.52	.49	.54	.53	.47
	9	.64	.60	.61	.70	.48
	10	.40	.40	.43	.37	.38
	Mean	.71	.70	.70	.72	.39

Item difficulties are listed for all conditions (All) and separately for the no-reward condition (None), the performance-unrelated reward condition, and the performance-related reward condition. The item difficulty of each item was calculated as the proportion of test takers who correctly solved the item

References

- Armstrong, R. D., & Shi, M. (2009). A parametric cumulative sum statistic for person fit. *Applied Psychological Measurement*, 33, 391–410. doi:[10.1177/0146621609331961](https://doi.org/10.1177/0146621609331961)
- Arthur, W., Glaze, R. M., Villado, A. J., & Taylor, J. E. (2010). The magnitude and extent of cheating and response distortion effects on unproctored Internet-based tests of cognitive ability and personality. *International Journal of Selection and Assessment*, 18, 1–16. doi:[10.1111/j.1468-2389.2010.00476.x](https://doi.org/10.1111/j.1468-2389.2010.00476.x)
- Aust, F., Diederhofen, B., Ullrich, S., & Musch, J. (2013). Seriousness checks are useful to improve data validity in online research. *Behavioral Research Methods*, 45, 527–535. doi:[10.3758/s13428-012-0265-2](https://doi.org/10.3758/s13428-012-0265-2)
- Barnhoorn, J. S., Haasnoot, E., Bocanegra, B. R., & van Steenbergen, H. (2015). QRTengine: An easy solution for running online reaction time experiments using Qualtrics. *Behavior Research Methods*, 47, 918–929. doi:[10.3758/s13428-014-0530-7](https://doi.org/10.3758/s13428-014-0530-7)
- Callegaro, M. (2013). Paradata in web surveys. In F. Kreuter (Ed.), *Improving surveys with paradata: Analytic uses of process information* (pp. 261–279). Hoboken, NJ: Wiley.
- Carstairs, J., & Myers, B. (2009). Internet testing: A natural experiment reveals test score inflation on a high-stakes, unproctored cognitive test. *Computers in Human Behavior*, 25, 738–742. doi:[10.1016/j.chb.2009.01.011](https://doi.org/10.1016/j.chb.2009.01.011)
- Chetverikov, A., & Upravitelev, P. (2015). Online versus offline: The Web as a medium for response time data collection. *Behavior Research Methods*. Advance online publication. doi:[10.3758/s13428-015-0632-x](https://doi.org/10.3758/s13428-015-0632-x)
- Couper, M. P. (2005). Technology trends in survey data collection. *Social Science Computer Review*, 23, 486–501. doi:[10.1177/0894439305278972](https://doi.org/10.1177/0894439305278972)
- Couper, M. P. (2008). *Designing effective web surveys*. New York, NY: Cambridge University Press.
- de Leeuw, J. R., & Motz, B. A. (2016). Psychophysics in a Web browser? Comparing response times collected with JavaScript and Psychophysics Toolbox in a visual search task. *Behavior Research Methods*, 48, 1–12. doi:[10.3758/s13428-015-0567-2](https://doi.org/10.3758/s13428-015-0567-2)

- Diedenhofen, B., & Musch, J. (2015). cocor: A comprehensive solution for the statistical comparison of correlations. *PLoS One*, 10, e0121945. doi:10.1371/journal.pone.0121945
- Do, B.-R. (2009). Research on unproctored internet testing. *Industrial and Organizational Psychology*, 2, 49–51. doi:10.1111/j.1754-9434.2008.01107.x
- Fisher, R. A. (1925). *Statistical methods for research workers*. Edinburgh, Scotland: Oliver and Boyd. Retrieved November 4, 2015, from <http://psychclassics.yorku.ca>
- Formann, A. K., Waldherr, K., & Piswanger, K. (2011). *Wiener Matrizen-Test 2 (WMT-2): Ein Rasch-skaliertes sprachfreies Kurztest zur Erfassung der Intelligenz* [Viennese Matrices Test 2: A Rasch-scaled language-free short test for the assessment of intelligence]. Göttingen, Germany: Hogrefe.
- Foster, D. (2009). Secure, online, high-stakes testing: Science fiction or business reality? *Industrial and Organizational Psychology*, 2, 31–34. doi:10.1111/j.1754-9434.2008.01103.x
- Göriz, A. S. (2007). The induction of mood via the WWW. *Motivation and Emotion*, 31, 35–47. doi:10.1007/s11031-006-9047-4
- Guo, J., & Drasgow, F. (2010). Identifying cheating on unproctored Internet tests: The Z-test and the likelihood ratio test. *International Journal of Selection and Assessment*, 18, 351–364. doi:10.1111/j.1468-2389.2010.00518.x
- Hamon, O. R., & Lambrinos, J. (2008). Are online exams an invitation to cheat? *Journal of Economic Education*, 39, 116–125. doi:10.3200/JECE.39.2.116-125
- Heerwegh, D. (2003). Explaining response latencies and changing answers using client-side paradata from a web survey. *Social Science Computer Review*, 21, 360–373. doi:10.1177/0894439303253985
- Jensen, C., & Thomsen, J. P. F. (2014). Self-reported cheating in web surveys on political knowledge. *Quality and Quantity*, 48, 3343–3354. doi:10.1007/s11135-013-9960-z
- Kaczmarek, L. (2008). *Human survey-interaction: Usability and nonresponse in online surveys* (Doctoral dissertation, University of Mannheim). Retrieved April 2, 2015, from <https://ub-madoc.bib.uni-mannheim.de/2150>
- Karabatsos, G. (2003). Comparing the aberrant response detection performance of thirty-six person-fit statistics. *Applied Measurement in Education*, 16, 277–298. doi:10.1207/S15324818AME1604_2
- Karim, M. N., Kaminsky, S. E., & Behrend, T. S. (2014). Cheating, reactions, and performance in remotely proctored testing: An exploratory experimental study. *Journal of Business and Psychology*, 29, 555–572. doi:10.1007/s10869-014-9343-z
- Kerstin, M., & Wottawa, H. (2014). „Gegen schlechte Gewohnheit“ [“Against simple habits”]. *Personalmagazin*, 16(10), 38–39.
- Kieslich, P. J., & Hilbig, B. E. (2014). Cognitive conflict in social dilemmas: An analysis of response dynamics. *Judgment and Decision Making*, 9, 510–522. Retrieved April 2, 2015, from <http://journal.sjdm.org>
- Kreuter, F. (2013). Improving surveys with paradata: Introduction. In F. Kreuter (Ed.), *Improving surveys with paradata: Analytic uses of process information* (pp. 1–9). Hoboken, NJ: Wiley.
- Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10, 707–710.
- Lievens, F., & Burke, E. (2011). Dealing with the threats inherent in unproctored Internet testing of cognitive ability: Results from a large-scale operational test program. *Journal of Occupational and Organizational Psychology*, 84, 817–824. doi:10.1348/096317910X522672
- Malhotra, N. (2008). Completion time and response order effects in web surveys. *Public Opinion Quarterly*, 72, 914–934. doi:10.1093/poq/nfn050
- Nye, C. D., Do, B.-R., Drasgow, F., & Fine, S. (2008). Two-step testing in employee selection: Is score inflation a problem? *International Journal of Selection and Assessment*, 16, 112–120. doi:10.1111/j.1468-2389.2008.00416.x
- Olson, K., & Parkhurst, B. (2013). Collecting paradata for measurement error evaluations. In F. Kreuter (Ed.), *Improving surveys with paradata: Analytic uses of process information* (pp. 43–72). Hoboken, NJ: Wiley.
- QuestBack. (2013). Unipark EFS Survey 9.1. Retrieved April 2, 2015, from <http://www.unipark.de>
- Rand, D. G. (2012). The promise of Mechanical Turk: How online labor markets can help theorists run behavioral experiments. *Journal of Theoretical Biology*, 299, 172–179. doi:10.1016/j.jtbi.2011.03.004
- Reips, U.-D. (2000). The Web experiment method: Advantages, disadvantages, and solutions. In M. H. Birnbaum (Ed.), *Psychology experiments on the Internet* (pp. 89–117). San Diego, CA: Academic Press.
- Reips, U.-D. (2008). How internet-mediated research changes science. In A. Barak (Ed.), *Psychological aspects of cyberspace: Theory, research, applications* (pp. 268–294). New York, NY: Cambridge University Press.
- Schmidt, W. C. (1997). World-Wide Web survey research: Benefits, potential problems, and solutions. *Behavior Research Methods, Instruments, & Computers*, 29, 274–279. doi:10.3758/BF03204826
- Schwarz, S., & Reips, U.-D. (2001). CGI versus JavaScript: A Web experiment on the reversed hindsight bias. In U.-D. Reips & M. Bosnjak (Eds.), *Dimensions of Internet science* (pp. 75–90). Lengerich, Germany: Pabst.
- Statista. (2015). Market share of web search engines in Germany. Retrieved March 24, 2015, from <http://de.statista.com/statistik/daten/studie/167841/umfrage/marktanteile-ausgewahlter-suchmaschinen-in-deutschland/>
- Steiger, J. H. (1980). Tests for comparing elements of a correlation matrix. *Psychological Bulletin*, 87, 245–251. doi:10.1037/0033-2909.87.2.245
- Stern, M. J. (2008). The use of client-side paradata in analyzing the effects of visual layout on changing responses in web surveys. *Field Methods*, 20, 377–398. doi:10.1177/1525822X08320421
- Stieger, S., & Reips, U.-D. (2010). What are participants doing while filling in an online questionnaire: A paradata collection tool and an empirical study. *Computers in Human Behavior*, 26, 1488–1495. doi:10.1016/j.chb.2010.05.013
- Tendeiro, J. N., Meijer, R. R., Schakel, L., & Maij-de Meij, A. M. (2013). Using cumulative sum statistics to detect inconsistencies in unproctored Internet testing. *Educational and Psychological Measurement*, 73, 143–161. doi:10.1177/0013164412444787
- The International Test Commission (2006). International guidelines on computer-based and internet-delivered testing. *International Journal of Testing*, 6, 143–171. doi:10.1207/s15327574ijt0602_4
- Tippins, N. T., Beaty, J., Drasgow, F., Gibson, W. M., Pearlman, K., Segall, D. O., & Shepherd, W. (2006). Unproctored internet testing in employment settings. *Personnel Psychology*, 59, 189–225. doi:10.1111/j.1744-6570.2006.00909.x
- Yan, T., & Olson, K. (2013). Analyzing paradata to investigate measurement error. In F. Kreuter (Ed.), *Improving surveys with paradata: Analytic uses of process information* (pp. 73–95). Hoboken, NJ: Wiley.