Understanding reCAPTCHAv2 via a Large-Scale Live User Study

Andrew Searles
University of California Irvine
searlesa@uci.edu

Renascence Tarafder Prapty
University of California Irvine
rprapty@uci.edu

Gene Tsudik
University of California Irvine
gts@ics.uci.edu

***Abstract*—Since 2003, CAPTCHAS have been widely used as a barrier against bots, while simultaneously annoying great multitudes of users worldwide. As the use of CAPTCHAS grew, techniques to defeat or bypass them kept improving. In response, CAPTCHAS themselves evolved in terms of sophistication and diversity, becoming increasingly difficult to solve for both bots and humans. Given this long-standing and still-ongoing arms race, it is important to investigate usability, solving performance, and user perceptions of modern CAPTCHAS. In this work, we do so via a large scale (over 3,600 distinct users) 13-month real-world user study and post-study survey. The study, conducted at a large public university, is based on a live account creation and password recovery service with currently prevalent CAPTCHA type: reCAPTCHAv2.**

Results show that, with more attempts, users improve in solving checkbox CAPTCHAS. For website developers and user study designers, results indicate that the website context, i.e., whether the service is password recovery or account creation, directly influences (with statistically significant differences) CAPTCHA solving times. We consider the impact of participants' major and education level, showing that certain majors exhibit better performance, while, in general, education level has a direct impact on solving time. Unsurprisingly, we discover that participants find image CAPTCHAS to be annoying, while checkbox CAPTCHAS are perceived as easy. We also show that, rated via System Usability Scale (SUS), image CAPTCHAS are viewed as "OK", while checkbox CAPTCHAS are viewed as "good".

Finally, we also explore the cost and security of reCAPTCHAv2 and conclude that it comes at an immense cost and offers practically no security. Overall, we believe that this study's results prompt a natural conclusion: *reCAPTCHAv2 and similar reCAPTCHA technology should be deprecated.*

I. Introduction

Many types of Internet-based activities and services require verification of human presence, e.g., ticket sales, reservations, and account creation. Left unchecked, bots will gobble up most resources available through such activities: they are much faster and way more agile than any human or a group thereof. This problem is not new: the first seminal step to combat it took place in 2003 when von Ahn et al. [1] proposed CAPTCHA as an automated test that is supposed to be easy for humans to pass, yet difficult or impossible for computer programs (aka bots) at the time. The key conjecture underlying the CAPTCHA concept is that, if a computer program successfully solved CAPTCHAS, then the same program could be repurposed to solve some computationally hard AI problem.

This seemed to be a win-win situation: either CAPTCHAS attest to genuine human presence, or they spur a significant advance in AI technology. Furthermore, CAPTCHAS were touted as a tool for the common good, since human-based solutions helped with difficult (for computers) and useful tasks, such as recognizing blurred text that confounded OCR algorithms, or labeling photos with names of objects appearing in them in order to aid image classification.

Another major advance occurred in 2007 when von Ahn et al. introduced reCAPTCHA [2]. reCAPTCHA was designed to reuse challenge results as a form of human-based data labeling for advancing machine learning. Google acquired reCAPTCHA in late 2009 [3] and, by June 2010, it was reported that reCAPTCHA had over 100 million distinct daily users [4]. Assuming that this number stayed constant since 2010 (though it most likely grew significantly), over half a trillion reCAPTCHAs have been solved in the meantime. This collectively amounts to an immense human cost.

However, almost from the start, an "arms race" began between bot and CAPTCHA developers. Most early CAPTCHA types were based on recognition of distorted text. Unfortunately, as a consequence of rapid advances in machine learning and computer vision, bots evolved to quickly and accurately recognize and classify distorted text [5]–[7], reaching over 99% accuracy by 2014 [8], [9]. To this end, in 2012 Google transitioned from using distorted text to image classification, utilizing images from the Google Street View project [10]. This transition ended in 2014 with the introduction of reCAPTCHAv2 [11], which employs a two-step process: (1) behavioral analysis combined with a simple checkbox (checkbox CAPTCHA), and (2) image classification tasks (image CAPTCHA) as a fallback for users who fail the checkbox challenge [12]. By 2016, both (1) and (2) were defeated with a high degree of accuracy by bots [13].

Regardless of its diminished efficacy, reCAPTCHA remains the prevalent CAPTCHA type on the Internet [14], deployed on over 13 million websites in 2023. It is therefore important to periodically evaluate and quantify its impact in terms of usability, solving performance, and user perceptions.

Several prior CAPTCHA user studies explored solving per-

Symposium on Usable Security and Privacy (USEC) 2025
24 February 2025, San Diego, CA, USA
ISBN 979-8-9919276-5-9
https://dx.doi.org/10.14722/usec.2025.23021
www.ndss-symposium.org, https://www.usablesecurity.net/USEC/

formance, e.g., [15]–[28]. Also, [21], [23], [26] looked into usability of CAPTCHAS via the well-known SUS scale. [15], [17], [18], [22], [24], [25], [28] studied user preferences related to CAPTCHA types. However, only two recent (2019/2023) user studies [15], [28] involved reCAPTCHAv2. Between them, [28] had relatively few participants (40), used unclear methodology, and did not consider usability. On the other hand, [15] presents interesting comparison points discussed in Section V. Most other user studies [18], [21]–[23], [25], [26] were conducted on newly proposed (and therefore, mocked-up) CAPTCHA types.

Furthermore, many previous CAPTCHA studies [15], [16], [21]–[23], [26], [27] were conducted on Amazon Mechanical Turk (MTurk) [29], which exhibits data quality issues [30]. Also, all these studies involved some bias, since participants were informed about study goals, i.e., they were selected based on their willingness to solve CAPTCHAS, for a certain monetary reward.

The above discussion motivates the work presented in this paper, the centerpiece of which is a large-scale ($> 3,600$ participants) 13-month IRB-approved user study of reCAPTCHAv2. The key novelty of the study is its use of a live account creation and password recovery service, with participants who were unaware of their involvement in a research study, most of whom were first-time users of the service. This approach closely resembles the organic scenario of users encountering CAPTCHAS in real-world online services, thus significantly enhancing reliability and generalizability of the findings. Study results yield some interesting observations that might be of interest to CAPTCHA designers as well as websites using (or considering the use of) CAPTCHAS.

Main contributions of this work are:

- A comprehensive quantitative analysis of solving time and its impact. In particular, this work reports on the first study to obtain multiple solving attempts per person. It shows that form-specific checkbox CAPTCHA solving time improves with more attempts, with the first attempt being 35% slower than the 10th, as shown in Tables VII and VIII. We also observe statistically significant differences in checkbox CAPTCHA solving time based on the type of service, with password recovery being faster, as shown in Tables IV, V and VI. With respect to educational level[1], there is a direct trend from freshmen (slowest) to seniors (fastest) at solving reCAPTCHAv2 as shown in Table IX. In terms of participants' major (field of study), there are minor trends with statistical significance: solving times of technical majors are faster than that of non-technical majors, as shown in Table X.
- An in-depth quantitative and qualitative analysis of reCAPTCHAv2 usability for both checkbox and image CAPTCHAS. Results demonstrate that 40% of participants found image CAPTCHAS to be annoying (or very annoying), while $< 10\%$ found checkbox CAPTCHAS annoying. SUS data shows that image CAPTCHAS have a mid-score of 58, while checkbox CAPTCHAS have a score of 78, with 90 being the highest score observed. Based on the open-ended feedback represented in a *word cloud*, participants' most frequent term for checkbox CAPTCHAS was **"easy"** and, for image CAPTCHAS– **"annoying"**.
- A detailed discussion of the cost and security of reCAPTCHAv2 (Section VI). Our security analysis shows a blatant vulnerability in the behavioral analysis of reCAPTCHAv2 [31], the ease of implementing large-scale automation [32], usage of privacy-invasive tracking cookies [32], and the weak security premise of fallback method (image challenge) [33]. Our cost analysis investigates total human time spent solving reCAPTCHAv2, human labor cost, network traffic volume, electricity usage, potential profits, and the corresponding environmental impact. There have been at least 512 billion reCAPTCHAv2 sessions, taking 819 million hours, which translates into at least $6.1 billion USD in free wages. The network traffic resulting from reCAPTCHAv2 consumed 134 Petabytes of bandwidth, translating into about 7.5 million kWhs of energy, and corresponding to 7.5 million pounds of CO2 pollution.

Organization: Section II provides some background on current CAPTCHA types, reCAPTCHAv2 solving scenarios, and System Usability Scale (SUS). Then, Section III describes the methodology, design, ethics, and implementation of the user study. Next, Section IV presents the results and their analysis. Then, Section V contextualizes our results against previous user studies. Next, Section VI presents the cost and security analysis. Section VII concludes the paper.

II. BACKGROUND

A. CAPTCHAS

A recent survey by Guerar et al. [34] provides a comprehensive overview of the current CAPTCHA landscape. It proposes a ten-group classification to encompass all current and emerging schemes: Text-based, Image-based, Audio-based, Video-based, Game-based, Slider-based, Math-based, Behavior-based, Sensor-based, and Liveliness-detection. It also discusses usability, attack resilience, privacy, and open challenges for each class. Since this paper focuses on behavior and image-based CAPTCHAS (which are used in reCAPTCHAv2), we summarize them below. For the rest, we refer to [34].

Image-based CAPTCHAS typically require users to perform an image classification task, such as selecting images that match the accompanying written description. The most popular instances are hCAPTCHA [35] and reCAPTCHA [36], starting from version 2 onward. An example of reCAPTCHAv2 image CAPTCHA is shown in Figure 1. The difficulty of image-based CAPTCHAS is associated with that of computer vision-based image classification. When these CAPTCHA types were introduced, corresponding problems were not easily solvable by machines. However, as computer vision research advanced, attacks on image-based CAPTCHAS became more successful. Concrete attacks include [32], [37]–[41], some of which

[1] In the American undergraduate system, "freshmen" are 1st-year students, "sophomore" – 2nd, "junior" – 3rd, and "senior" – 4th.

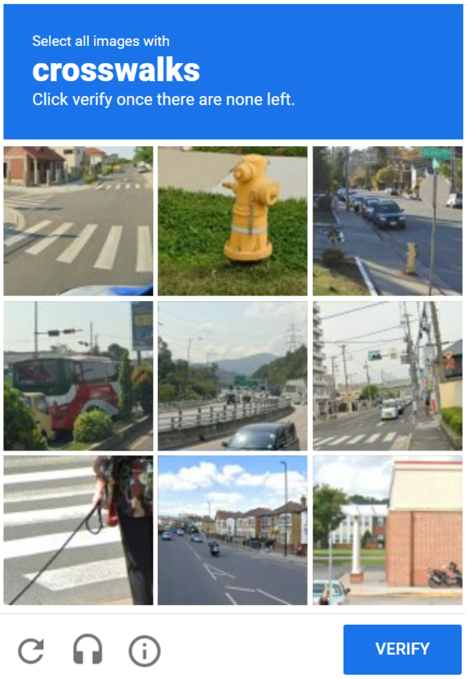

Select all images with
crosswalks
Click verify once there are none left.

Fig. 1: Image Labeling Task CAPTCHA [36]

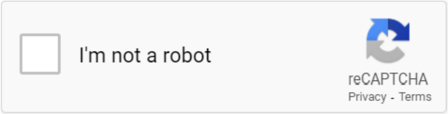

□ I'm not a robot

Fig. 2: reCAPTCHAv2 behavior (checkbox) CAPTCHA [12]

report success rates of 85% for reCAPTCHA and 96% for hCAPTCHA.

Behavior-based (or invisible) CAPTCHAS are newer: they either require users to click a checkbox (e.g., "I am not a robot") or are completely invisible/transparent to the user. Instead of a visual challenge, they rely on client-side scripts and other opaque techniques to collect, in the background, historical behavioral information about the user. This information is sent to the CAPTCHA provider, which uses various heuristic-based techniques to identify bot-like behavior. For instance, Google's popular No-CAPTCHA reCAPTCHA: *"actively considers a user's entire engagement with the CAPTCHA – before, during, and after – to determine whether that user is a human"* [12]. Sivakorn et al. [13] evaluated the reCAPTCHA risk analysis system and determined that Google tracks cookies, browsing history, and the browser environment, such as canvas rendering, user-agent, screen resolution, and mouse movements. [13] also showed that legitimate cookies can be automatically farmed to attack reCAPTCHAv2 with 100% success, at a large scale.

reCAPTCHAv2 Solving Scenarios reCAPTCHAv2 uses a combination of behavior-based and image-based CAPTCHAS. Initially, a user is presented with a behavior-based CAPTCHA: clicking a checkbox. Google also considers various aspects of user interaction, as described in the previous section. If the user fails the checkbox CAPTCHA due to Google considering the user's interactions suspicious, an image-based CAPTCHA is served to the user. Otherwise, the user's interaction with reCAPTCHAv2 ends after clicking the checkbox. In summary, there are two reCAPTCHAv2 solving scenarios:
Checkbox only: After clicking the checkbox, no image CAPTCHA is served.
Checkbox+image: After clicking the checkbox, an image CAPTCHA is served.

B. System Usability Scale (SUS)

System Usability Scale (SUS), shown in Figure 3, is a classical and popular survey method designed to assess usability of various systems or products. Proposed by Brooke et al. [42] in 1996, it consists of ten statements: five positive and five negative. Each statement is on a 5-point Likert scale ranging from *Strongly Disagree (1)* to *Strongly Agree (5)*.

1) **I think that I would like to use this system frequently.**
2) **I found the system unnecessarily complex.**
3) **I thought the system was easy to use.**
4) **I think that I would need the support of a technical person to be able to use this system.**
5) **I found the various functions in this system were well integrated.**
6) **I thought there was too much inconsistency in this system.**
7) **I would imagine that most people would learn to use this system very quickly.**
8) **I found the system very cumbersome to use.**
9) **I felt very confident using the system.**
10) **I needed to learn a lot of things before I could get going with this system.**

Fig. 3: System Usability Scale (SUS)

SUS is widely used to measure usability of a wide range of products and systems, from everyday products (such as phones, fitness bands, and appliances [43], [44]) to websites, software, mobile apps and even CAPTCHAS [43], [45]–[49]. SUS is very popular because of its simplicity and conciseness. Participants tend to easily understand and quickly complete the SUS questionnaire. The process of calculating scores is also very straightforward: (1) For odd-numbered statements, subtract 1 from each response value, (2) For even-numbered statements, subtract each response value from 5, (3) Sum up all response values and multiply the result by 2.5.

This yields a SUS score between 0 and 100 for each participant.

To associate a given usability level with individual scores, [50] provides adjective scaling, shown in Table I. This scale consists of seven usability levels, starting from the worst imaginable usability and going up to the best imaginable usability.

TABLE I: Adjective Ratings of SUS Scores

Adjective	Mean SUS Score
Worst Imaginable	12.5
Awful	20.3
Poor	35.7
OK	50.9
Good	71.4
Excellent	85.5
Best Imaginable	90.9

III. The User Study

Recall that the goals of the user study are to measure solving times and user perceptions of reCAPTCHAv2, the currently prevalent CAPTCHA type.

A. The Setting

This study was conducted continuously over roughly 13 months. It took place on the campus of the University of California Irvine, though the scope was limited to one specific school.[2] The specific school hosting our study is *School of Information & Computer Sciences (SICS)*. It includes several departments, all related to Computer Science. SICS offers a number of fairly typical undergraduate (BS) and graduate (MS and PhD) programs.

For many years, SICS requires every person, who for the first time enrolls in any SICS course, to create a SICS-specific user account via the school's web interface. A typical scenario is that a student who enrolls in at least one SICS course in their entire university career creates a SICS account **only once**. Consequently, a student who wants to create a SICS account has not previously engaged in SICS account creation, meaning that they have no knowledge of the workflow involved, and no expectations of either seeing or not seeing CAPTCHAS as part of the process.

This motivates the key feature of our user study: introduction (insertion) of reCAPTCHAv2 into the SICS account management workflow. This involves two separate services: (1) account creation for new users, and (2) password recovery for users with existing accounts. This was accomplished with the much-appreciated help and cooperation of the SICS IT Department.

As mentioned earlier, the study was conducted over approximately 13 months to include as many distinct users as possible. Since the yearly academic calendar has multiple terms, we aimed to catch the beginning of each term, as this is the time when the bulk of new account creation and password recovery activities typically take place.

B. Justification

We now discuss the rationale for the user study setting. Clearly, an ideal and comprehensive CAPTCHA user study would be as inclusive as possible, comprising a true cross-section of the world population. Whereas, our study's targeted participants are (mostly) university students, including undergraduates who range from incoming (freshmen) to graduating (seniors), as well as graduate students enrolled in a variety of programs (MS, MA, MBA, MFA, JD, MD, PhD). The latter are split among so-called *professional* degree programs, e.g., MBA, JD, MD, and some MS/MA, while others are in regular degree programs, e.g., PhD, MFA, and some MS/MA. Such participants are surely not representative of the world, or even national, user population. Nonetheless, we conjecture that data from this admittedly narrow population segment is useful, as it reflects an "optimistic" perception of CAPTCHAS. This is because young and tech-savvy users represent the most agile population segment and the one most accustomed to dealing with CAPTCHAS, due to their heavy Internet use. Thus, by examining various (not generally positive) impact factors of CAPTCHAS, we prefer to err on the side of the population that is intuitively the least allergic to CAPTCHA use.

Some reasons for our study setting are fairly obvious. In particular, it would have been very challenging, if not impossible, to convince any other organization to introduce CAPTCHAS into its service workflow, or to allow us to collect data about their current CAPTCHA use. Alternatively, one could imagine approaching Google and requesting access to the centralized reCAPTCHAv2 service. This would have been ideal since it would give us access to a huge number of diverse reCAPTCHAv2 users worldwide. Indeed, we attempted to do this. However, Google's legal team denied our request to gain access to large-scale data from reCAPTCHAv2. There is very likely a natural counter-incentive for Google (or any other CAPTCHA provider) to cooperate with outside researchers in a user study, since doing so might reveal certain negative aspects of the service. Another possibility would have been to create our own brand-new service and use CAPTCHA to guard access to it, thus hoping to attract prospective users of broad demographics. While theoretically plausible, doing so would be prohibitively time and effort-consuming for academic researchers.

Finally, even with our somewhat narrow target demographic of university students, the user study could have been more latitudinal, i.e., it could span multiple universities in various parts of the world. This would have yielded more valuable results across political, cultural, and linguistic boundaries. However, this would have been a massive effort requiring careful coordination with, and participation of, both researchers and IT departments in each university.

C. The Website

The SICS website used in the study is hosted within the university network. In order to create a SICS account, a user must first login to the campus VPN with their university account. This allows us to claim, with high confidence, that all collected data stemmed from real human users, who are, for the most part, students (see Section III-E below).

The back-end is a basic PHP server that serves HTML and JavaScript. It is maintained by SICS IT department. The account creation service includes a form requesting basic student information, e.g., name and student ID. The password recovery service includes a form requesting existing account information. In both cases, reCAPTCHAv2 was initially hidden and rendered after clicking the submit button. Basic website workflows for account creation and password recovery are described in Appendix A.

All timing events were measured using JavaScript native Date library, which has millisecond precision. JavaScript was used to block form submission, such that an initial timing event is recorded and a reCAPTCHAv2 is rendered simultaneously.

[2] The term *school* denotes an organizational entity that includes two or more academic departments. The university contains several such schools, e.g., School of Engineering, School of Law, and School of Humanities.

Initially, a behavior-based checkbox CAPTCHA is presented. In order to solve it, a user clicks the checkbox, sending data to Google reCAPTCHAv2 site. It either validates the user as a human or presents an image-based CAPTCHA. Upon successful reCAPTCHAv2 validation (through checkbox or checkbox+image(s)), a second timing event is captured and the form is submitted.

Solving time is thus comprised of the time interval starting from CAPTCHA rendering until the client browser receives a successful validation response from Google reCAPTCHAv2 service (this may even involve the user solving multiple image CAPTCHAS). Upon successful form submission, the IT database stores these two timestamps, along with the form information.

D. Directory Crawler

Recall that the study involved unwitting participants, i.e., unaware of both the existence and purpose of the study. To subsequently obtain demographic information about each participant, we created a JavaScript crawler that automatically searches the university directory using email addresses. This directory is publicly available from both inside and outside the university network. Information gathered by the crawler includes major and college education level (freshman, sophomore, junior, senior, or graduate) of each participant.

E. Logistics & Data Cleaning

In total, the SICS IT department supplied 9169 instances of account creation and password recovery with reCAPTCHAv2 solving time data. The original form data was larger since it included errors, such as incomplete forms and incorrect values. Each record (form) has the following fields: database ID, date and time, student ID, email address, service, and timing. Starting with 9169 instances, we filtered results using the directory crawler, labeling entries with student IDs that were not found and correcting student IDs with minor typos. A total of 226 entries were labeled as none for student ID and 295 student ID typos were corrected.

Successful form submissions have certain constraints, e.g., field formatting. If a person enters erroneous data that does not fit the constraints, they still have to solve a reCAPTCHAv2 CAPTCHA before the form is submitted. Cases of multiple submissions occurred because of unsuccessful attempts to enter form data. For some entries, there were small typos, though mixed with temporal evidence they were correctable.

28 records were removed as outliers since each had a solving time of > 60 seconds. We ended up with 9141 valid records of which 8915 correspond to 3625 unique participants. 226 entries, labeled as none for student ID, are not included among the unique participants, attempts, educational level, and major analysis. Of the 8915, 284 form submissions correspond to 52 unique non-students (i.e., faculty or staff) and are not included in the educational level and major analysis. For the purposes of the educational level and major analysis, 3573 unique students completed 8631 reCAPTCHAv2 CAPTCHAS.

F. Post-Study Survey

After the completion of the study, we randomly selected and contacted, by email, 800 participants to solicit feedback on their reCAPTCHAv2 CAPTCHA solving experience via a survey (a Google form). In the end, a total of 108 completed the survey. The incentive was an $5 Amazon gift card. The survey collected answers to SUS questions regarding both checkbox and image CAPTCHAS. It also collected information about (more detailed) demographics, frequency and nature of internet usage, as well as preferences and opinions about checkbox and image CAPTCHAS.

G. Ethical Considerations

The user study was duly approved by the university's Institutional Review Board (IRB). Collection of student email addresses for recruitment and demographic analysis purposes was also explicitly approved. Since prospective participants were not pre-informed of their participation in the study, two additional documents were filed and approved by the IRB: (1) *"Use of deception/incomplete disclosure"* and (2) *"Waiver or Alteration of the Consent"*. Study participants who completed the post-study survey were compensated US$5 for about 5 minutes of their time. This was also IRB-approved.

No personally identifiable information (PII) was used in the demographics analysis.

After the completion of the study, **all** participants were informed, by email, of their participation and the purpose of the study. They were also informed that some basic public demographic information about them was collected via campus directory lookup. Following our notifications, three participants reached out to the research team expressing concerns regarding data privacy. The study team promptly addressed their inquiries and successfully alleviated their concerns.

IV. RESULTS & ANALYSIS

This section presents the results of the user study based on the live service experiment. We consider both quantitative (solving time) and qualitative (SUS, rating, feedback) data to provide a comprehensive analysis of reCAPTCHAv2 usability.

A. Demographics

The student population of the university is large and diverse. We use university demographics because students from multiple departments who take any SICS course create accounts. Thus, demographics about SICS students would not be enough. Moreover, the university does not maintain or provide SICS-specific demographics.

According to recent statistics, the total number of students is $\approx 36,000$ of whom 54% are female, 44.6% are male, plus 1.4% are non-binary or unstated. In terms of ethnicity, the rough breakdown is: 34% Asian, 24% Hispanic, 17% international, 15.44% White, 2.23% Black, and 7.33% other ethnic groups. The split between undergraduate and graduate students is 78.10% to 21.9%.

As far as the educational level, freshmen constitute 14% of the student body, sophomores – 15%, juniors – 21%, and seniors – 28%. The rest (≈ 22%) are graduate students. Interestingly, the age range of the student population is very wide, ranging from under 18 to over 64. Nonetheless, the majority (82%) fall into the $18--24$ age range.

We also consider demographics of the 108 participants who engaged in the post-study survey. The gender split is 62 (57.4%) male, 44 (40.7%) female, and 2 (1.9%) non-binary. The age of participants ranges from 18 to 30 with the majority (87.04%) under 25. Participants were also asked about their highest level of education. All participants have at least a high school degree. 58 participants (53.7%) are undergraduates and 50 (46.3%) are graduate students. All participants use the Internet daily and the main purpose of Internet usage for the majority (57.4%) is education. Finally, the country of residence for most (82.4%) participants is the United States, which is directly in line with the 17% international students from the overall university demographics.

Unfortunately, similarly detailed demographics for participants who solved reCAPTCHAv2 CAPTCHAS as part of the main live experiment are unknown. However, the demographics of the 108 who participated in the post-study survey, closely resemble those of the overall campus total population in terms of gender, age, and educational level. Therefore, it is reasonable to assume that the demographics of all participants are the same, or very similar.

B. reCAPTCHAv2 Dashboard Data

Google provides reCAPTCHAv2 analytic data for website operators via a dashboard [51]. With it, website operators can generate a key-pair necessary for implementing reCAPTCHAv2 on a web page. The difficulty setting can also be chosen on the dashboard. We used the "easy" setting in all experiments. The admin console allows for data to be downloaded in CSV format with the following fields per day:

```
no CAPTCHAs, Passed CAPTCHAs, Failed
CAPTCHAs, Total Sessions, Failed
Sessions, Average Score, and Average
Response Time.
```

TABLE II: Google's reCAPTCHA dashboard data

no CAPTCHAs (checkbox)	7629
Passed CAPTCHAs (Image)	1890
Failed CAPTCHAs (Image)	143
Total Sessions	9538
Failed Sessions	19
Image accuracy	92.96%
Behavior accuracy	79.98%

Here, the term "no CAPTCHAs" indicates that only a behavior-based CAPTCHA, i.e. a checkbox, was presented to the user. Average score and response time are highly sparse and only appear on days with over 400 total sessions. Table II shows a sum for the entire study period. The image accuracy of 93% is computed as:

$$\frac{(\#passed \text{ CAPTCHAS})}{(\#passed \text{ CAPTCHAS} + \#failed \text{ CAPTCHAS})}$$

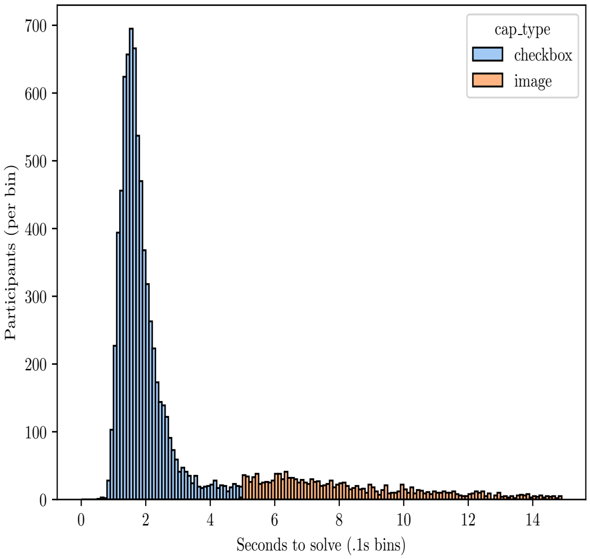


Fig. 4: Timing results in bins of .1 seconds

The behavioral accuracy of 80% is computed as:

$$\frac{(\#of \text{CAPTCHAS})}{(total\#sessions)}$$

Notably, there are 9538 CAPTCHA sessions reported by the admin console data, while we were supplied with 9169 sessions, meaning that 369 form submissions has incomplete data or resulted in an error. This is likely due to incomplete sessions, e.g., refreshing before validation, or other form submission errors.

TABLE III: Agglomerated solving time for reCAPTCHAv2

type	Count	Mean	Median	Std	Var	Max	Min
checkbox	7334	1.85	1.67	0.71	0.50	4.99	0.51
image	1807	10.3	8.20	6.54	42.8	59.8	4.99
total	9141	3.53	1.83	4.50	20.3	59.8	0.51

C. Solving Time

Solving time for reCAPTCHAv2 CAPTCHAS is measured from the initial display to the successful verification. Data for solving time is split based on behavioral accuracy of 80% from Table II. Since all reCAPTCHAv2 CAPTCHAS require clicking a checkbox and some also require solving an image CAPTCHA, we assume that the 80% fastest solving times correspond to checkbox interactions. This split is also noted in the recent work by Searles et al. [15]. All timings for image-based CAPTCHAS are, therefore, a combination of checkbox and image CAPTCHAS solving times.

Table III shows the solving time of 7334 checkbox and 1807 image CAPTCHAS based on this split. The mean solving time for checkbox CAPTCHA is 1.85 seconds, while the mean solving time for image CAPTCHA is 10.3 seconds. The latter corresponds to a notable 557% increase.

Looking at Figure 4, there is a sharp drop-off in solving time starting around 2 seconds, and ending at 5 seconds: it hits a low and then goes back up slightly. The split point for image and checkbox CAPTCHAS is about 5 seconds, which matches the drop-off point, thus strengthening the accuracy of the split.

Solving time can also be partitioned into the following dimensions, based on collected data: Service, Attempts, Educational Level, and Major.

TABLE IV: Checkbox solving time in seconds for each service

Service	Count	Mean	Median	Std	Var	Max	Min
Password Reset	2654	1.67	1.51	0.65	0.42	4.99	0.51
Account Creation	4680	1.96	1.76	0.71	0.51	4.97	0.86

TABLE V: Image solving time in seconds for each service

Service	Count	Mean	Median	Std	Var	Max	Min
Password Reset	332	10.4	8.01	6.59	43.5	43.5	5.01
Account Creation	1475	10.3	8.23	6.53	42.7	59.8	4.99

TABLE VI: Total solving time in seconds for each service

Service	Count	Mean	Median	Std	Var	Max	Min
Password Reset	2986	2.63	1.58	3.56	12.7	43.5	0.51
Account Creation	6155	3.97	2.00	4.84	23.4	59.8	0.86

1) Statistical Testing: For the sake of statistical validity, we apply a series of standard statistical tests to solving time data. We run all of the following statistical tests on both image CAPTCHA and checkbox CAPTCHA solving time data separately. Statistical methods were applied using Python's scipy [52] library. With a null hypothesis that solving times adhere to a normal distribution, we performed the *Shapiro-Wilk normality test*. For both image and checkbox CAPTCHAS, results show that we can reject the null hypothesis ($p < 0.001$). With a null hypothesis that the skewness is the same as that of a corresponding normal distribution, we ran the timing data with skewness. For both image and checkbox CAPTCHAS, the results of skewtest reject the null hypothesis in favor of the alternative: the distribution of solving times is skewed ($p < 0.001$) to the right. With a null hypothesis that the kurtosis is the same as that of a normal distribution, we used the *tailedness test*. For both image and checkbox CAPTCHAS, results show the samples were drawn from a population that has a heavy-tailed distribution ($p < 0.001$). We used the *Brown Forsythe test* to compare the equality of variance between image CAPTCHA and checkbox CAPTCHA solving times, which shows that they do not exhibit equal variance. We used the *Kruskal-Wallis test* with the Holm-Bonferroni method to adjust for family-wise error to test the equality of mean between modes, services, attempts, majors, and educational levels. Significant results are included in Figures 5, 6, and 7.

2) Services: As mentioned earlier, the website had two services that invoked reCAPTCHAv2 CAPTCHAS: password recovery and account creation. Tables IV, V, and VI show results from these two CAPTCHA interactions. There were 6155 account creations and 2986 password recovery form submissions.

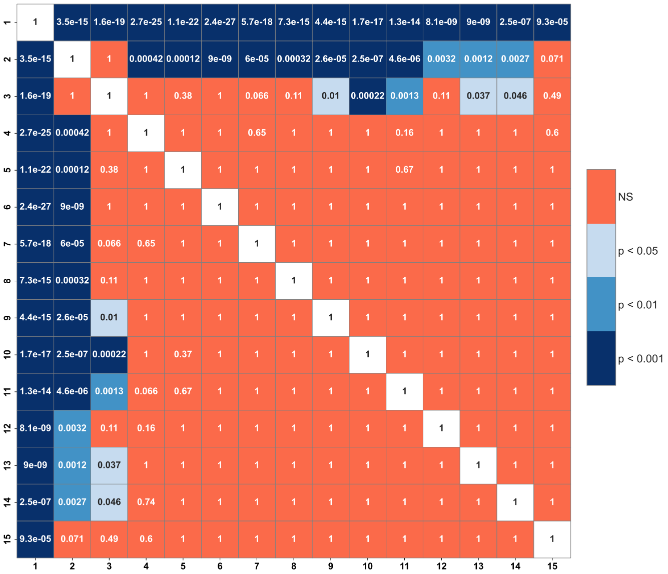


Fig. 5: Kruskal-Wallis results for checkbox attempts

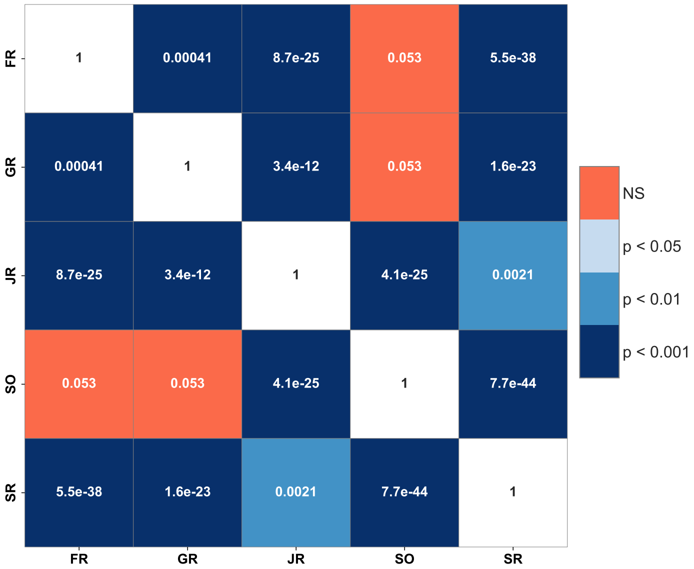


Fig. 6: Kruskal-Wallis results for total attempts and educational level

Notably, for checkbox CAPTCHA solving time, the Kruskal-Wallis test shows statistically significant differences between account creation and password recovery with a $p = 1.1e^{-115}$. Students who interacted with the account creation service solved checkbox CAPTCHAS 17% slower than those who interacted with the password recovery service.

Furthermore, 50% more time was spent solving reCAPTCHAv2 CAPTCHAS during account creation than during

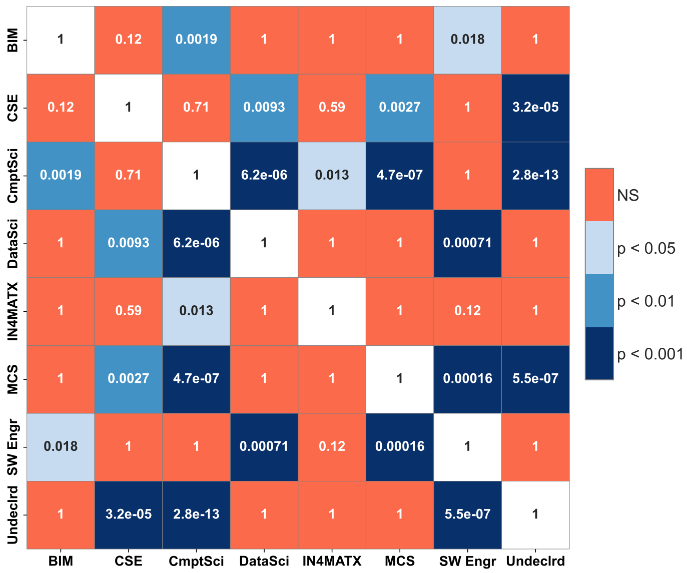


Fig. 7: Kruskal-Wallis results for total attempts and major

password recovery. This service-wise difference in total solving time is also statistically significant with $p = 6.7e^{-162}$. However, since 90% of students who used the password recovery service had also interacted with the account creation service featuring reCAPTCHAv2 CAPTCHAS (the remaining 10% created accounts before reCAPTCHAv2 was integrated into the account creation process), these results may have been influenced by prior attempts.

For image CAPTCHA solving time, the Kruskal-Wallis Test yielded no statistically significant results between account creation and password recovery services.

TABLE VII: Solving time for number of checkbox attempts

Attempt	Count	Mean	Median	Std	Var	Max	Min
1	2888	2.02	1.80	0.73	0.54	4.97	0.94
2	1293	1.84	1.67	0.65	0.42	4.97	0.62
3	751	1.80	1.63	0.66	0.44	4.95	0.80
4	513	1.73	1.55	0.63	0.40	4.89	0.78
5	371	1.73	1.57	0.70	0.49	4.92	0.89
6	272	1.61	1.47	0.58	0.34	4.57	0.84
7	212	1.67	1.52	0.65	0.43	4.90	0.64
8	167	1.66	1.52	0.65	0.43	4.65	0.84
9	127	1.60	1.48	0.57	0.33	4.09	0.88
10	112	1.56	1.44	0.63	0.39	4.97	0.85
11	94	1.63	1.41	0.76	0.57	4.90	0.88
12	67	1.61	1.46	0.68	0.46	4.47	0.51
13	52	1.58	1.37	0.70	0.49	4.49	0.96
14	37	1.53	1.45	0.63	0.40	4.62	0.92
15	28	1.51	1.41	0.56	0.31	3.88	0.88

3) Attempts: Interestingly, some participants submitted forms multiple times. For checkbox CAPTCHAS, the average number of attempts was 3.52, and it was 1.73 for image CAPTCHAS. Tables VII and VIII show timing results over multiple attempts. The highest number of attempts was 37 for checkbox CAPTCHAS and 20 for image CAPTCHAS. Checkbox CAPTCHA solving time from the Kruskal-Wallis test in Figure 5 indicates a statistically significant difference between the first and subsequent attempts ($p < .001$). Also, the second attempt has a statistically significant difference ($p < .001$) with all other attempts except the third. In general, this data shows that checkbox CAPTCHA solving time decreases with more attempts.

TABLE VIII: Solving time for number of image attempts

Attempt	Count	Mean	Median	Std	Var	Max	Min
1	1264	10.5	8.36	6.60	43.5	58.9	4.99
2	260	10.9	8.16	7.47	55.8	55.5	5.00
3	93	9.30	8.16	4.09	16.7	29.2	5.00
4	45	10.0	7.77	8.41	70.7	59.8	5.21
5	25	8.76	7.48	4.56	20.8	26.4	5.12
6	15	7.26	6.06	2.33	5.44	12.3	5.18

We observe an interesting behavioral phenomenon whereby participants react faster when they know what to expect. However, average image CAPTCHA solving time results show a slight increase on the second attempt, while subsequent attempts decrease. This may be attributed to reCAPTCHAv2 presenting a more difficult CAPTCHA on the second attempt. Nonetheless, the Kruskal-Wallis test does not show any statistically significant difference for multiple image CAPTCHA attempts. This is likely due to the drop-off in the number of participants who solved multiple image CAPTCHAS.

TABLE IX: Total solving time for different educational levels

Level	Count	Mean	Median	Std	Var	Max	Min
Freshmen	773	5.15	2.33	5.69	32.4	56.2	0.95
Sophomore	1681	4.33	2.05	5.47	29.9	59.8	0.91
Junior	2246	3.09	1.77	3.84	14.7	45.4	0.51
Senior	2745	2.85	1.71	3.62	13.1	43.9	0.64
Graduate	1186	3.82	1.97	4.83	23.3	50.0	0.91

4) Educational Level: Educational level was obtained via the website crawler, as described in Section III-D. Table IX presents data for different educational levels. In terms of statistical significance, Figure 6 shows statistically significant differences in total solving time for all educational levels. In terms of total time, freshmen are the slowest – 80% slower than seniors. There is a direct trend from freshmen to seniors showing a reduction in solving time. Similarly, there is a trend of the total ratio of image to checkbox CAPTCHAS.

5) Majors: Majors of the study participants (i.e., disciplines they study) were obtained through the website crawler, as described in Section III-D. Table X presents solving times for participants with various majors. Although there are 62 majors in total, Table X only shows 22 majors. This is because each of the remaining 40 majors had < 20 reCAPTCHAv2 sessions. As the Kruskall-Wallis test in Figure 7 shows, only 8 majors had statistically significant differences in terms of checkbox CAPTCHA solving time. Among these, Computer Science had the lowest, and Informatics – the highest, total average solving time.

TABLE X: Total solving time for various majors

Major	Count	Mean	Median	Std	Var	Max	Min
CmptSci	3185	3.19	1.75	4.05	16.4	44.5	0.62
CSE	950	3.51	1.81	4.23	17.9	42.0	0.64
Undclrd	850	4.47	2.03	6.02	36.2	59.8	0.95
SW Engr	796	3.38	1.75	4.06	16.5	45.4	0.51
MCS	404	3.65	2.08	4.32	18.7	38.1	0.91
DataSci	362	3.98	2.02	4.55	20.7	41.2	1.03
IN4MATX	287	4.14	1.89	6.29	39.5	50.9	1.01
BIM	226	3.79	1.97	3.91	15.3	25.5	0.89
GameDes	186	4.38	1.86	7.03	49.4	56.2	0.77
Math	147	3.50	1.89	4.11	16.9	28.9	1.00
MofData	131	3.63	1.94	3.92	15.3	25.6	1.03
EngrCpE	106	3.93	1.98	4.31	18.6	20.1	0.98
PSW ENG	97	3.32	1.93	3.33	11.1	21.3	0.91
Bus Adm	89	2.85	1.83	2.55	6.52	13.1	0.88
CSGames	85	2.43	1.61	2.60	6.77	18.4	0.88
BusEcon	78	3.57	2.06	3.76	14.1	21.3	0.95
Bio Sci	75	3.90	1.87	4.80	23.0	23.0	0.99
Stats	65	3.70	1.68	4.02	16.2	23.9	1.11
Cog Sci	44	2.96	2.02	2.81	7.90	16.8	0.97
Net Sys	39	4.55	2.07	8.81	77.6	50.0	1.33
Engr ME	34	4.08	2.16	4.18	17.5	16.1	1.39
Psych	32	2.14	1.65	1.49	2.21	7.65	1.05

TABLE XI: SUS scores for reCAPTCHAv2

Participants' Solving Scenario	**reCAPTCHA Type**	**SUS Score**
Checkbox only	Checkbox	78.51
Checkbox+image	Checkbox	76.21
Checkbox+image	Image	58.90

D. *Survey Results*

We now discuss the study results pertaining to usability, preferences, and opinions about reCAPTCHAv2. An interactive version of the google form we used is available at [53]. 800 randomly selected study participants were contacted by email, with the goal of obtaining at least 100 respondents. In the end, a total of 108 completed the survey. The breakdown of the participants based on the solving scenarios (from Section II) are: (1) Checkbox only: 42, (2) Checkbox+image: 66.

1) System Usability Scale (SUS) Score Analysis: Table XI reports the SUS score for both scenarios. Results from individual SUS statements are not analyzed, since they do not provide meaningful information [42], [50].

SUS scores for checkbox CAPTCHA are: 78.51 for checkbox only scenario, and 76.21 for checkbox+image scenario. Referring to Table I, the usability level for checkbox CAPTCHA in both scenarios is "Good". We thus conclude that for the checkbox CAPTCHA, the SUS score and the usability level do not vary depending on the solving scenario, i.e., whether or not an image CAPTCHA is served afterwards. On the other hand, the SUS score of image CAPTCHA is 58.90 and the usability level is "OK". This difference is likely influenced by the difficulty of the task, since clicking a checkbox is surely much simpler than classifying an image. Recall that, solving image CAPTCHAS takes 557% longer than solving checkbox CAPTCHAS.

2) Preference Analysis: Participants were asked to provide a rating using a 5-point Likert scale ranging from Very Annoying (1) to Very Pleasant (5). Figure 8 and Figure 9 show the preferences in both scenarios. The rating of checkbox CAPTCHA is: 3.62/5 for checkbox only, and 3.68/5 for checkbox+image, scenario. Similar to the SUS score, the rating of checkbox CAPTCHA is independent of the solving scenario. The rating of image CAPTCHA is appreciably lower, at 2.84/5.

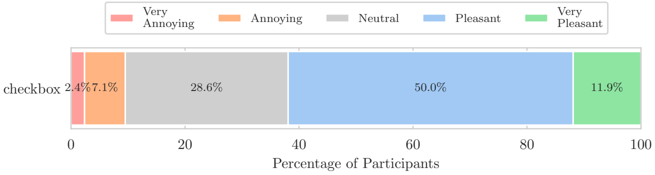


Fig. 8: Preference score for checkbox only scenario

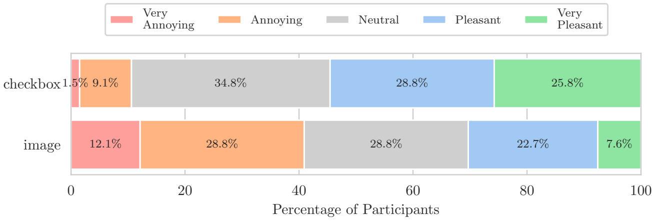


Fig. 9: Preference score for checkbox+image scenario

Comparing preference scores from Figure 8 and Figure 9 with SUS scores from Table XI we observe a trend for both checkbox and image: checkbox is more usable and rated positively, while image is less usable and rated negatively. This leads to an unsurprising conclusion that participants' preference for a given reCAPTCHAv2 type is correlated with its usability level.

3) Qualitative Feedback: In the final part of the survey, participants were asked to provide open-ended feedback about checkbox and image CAPTCHAS using at least one word. Using collected feedback, we generate word clouds for both.

The most prominent words for checkbox CAPTCHA in Figure 10 are "easy" and "simple". Other significant positive words are "good" and "quick". Nevertheless, checkbox CAPTCHA is still labeled "hard" and "annoying" by some participants.

Figure 11 shows that the most prominent word describing

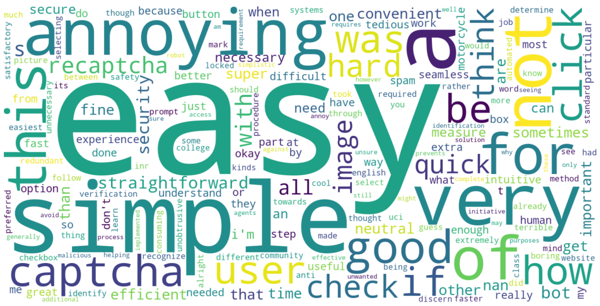


Fig. 10: Word cloud from feedback on checkbox

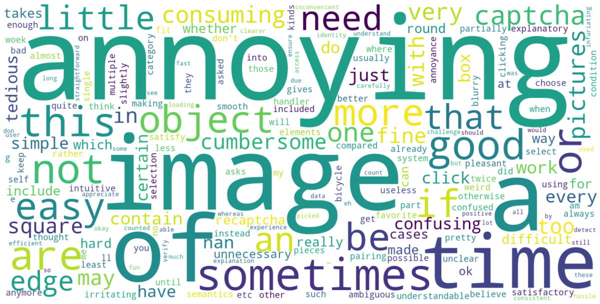


Fig. 11: Word cloud from feedback on image

image CAPTCHA is "annoying", while a small fraction of participants label it as "good" and "easy". We acknowledge that the custom scale used in the survey might possibly introduce bias toward the word "annoying". However, the scale also includes the word "pleasant" and that word is not present in the word cloud as a positive opinion. Instead, participants used other positive-sounding words, such as "easy" or "simple".

Negative words from the checkbox cloud and positive ones from the image cloud indicate that neither reCAPTCHAv2 type is universally liked or disliked.

V. Comparison with Related Work

Relevant and notable prior results include [15]–[28]. These results report average solving times for various CAPTCHAS, ranging from 3.1 to 47 seconds. Compared with our observed mean solving time of 1.8 seconds for reCAPTCHAv2 checkbox CAPTCHA, previous results are 1.7 to 26 times slower. For reCAPTCHAv2 image CAPTCHA, our mean solving time is 10 seconds, which is 3.3 times slower than the fastest, and 5 times faster than the slowest, previously reported results. The faster solving time may be related to the trend noted in [15], [16]: age influencing solving time. Younger participants seem to solve faster than older ones. Since our population is mostly university students (aged $18-25$), our results re-confirm this trend.

TABLE XII: Comparison with results from prior user studies evaluating reCAPTCHAv2: checkbox (C), image (I), total (T). Mean in seconds

Study	Unique users	reCAPTCHAv2s solved	Mean	Accuracy
Ours	3625	9141	10.4 (I), 1.85 (C), 3.53 (T)	93% (I), 80% (C)
Searles et al. [15]	1400	2800	14-26 (I), 3.1-4.9 (C)	71-81% (I), 71-85% (C)
Tanthavech et al. [28]	40	40	3.1 (T)	None

To the best of our knowledge, only two prior efforts studied reCAPTCHAv2: [15] and [28]. The former provides a very limited data set of ($n = 40$) reCAPTCHAv2, containing only total times, while [15] provides the following points of comparison:

1) Amazon Mturk *vs* "real world" participants
2) Participant awareness *vs* unawareness of study purpose
3) Mock *vs* real account creation
4) Preferences/Rating

Table XII shows a direct comparison of the results.

Webb et al. [30] reported several points of concern about the quality of data collected from MTurk [29]. Our data and results are derived from a real-world scenario of actual users creating real accounts for a real service. However, since both this work and Searles et al. [15] implement reCAPTCHAv2 in a similar way, some interesting conclusions can be drawn regarding the efficacy of Mturk data. Mturk users in [15] solved easy checkbox CAPTCHAS $1.7-2.7$ times slower than our participants. They also solved easy image CAPTCHAS $1.6-2.6$ times slower than our participants. Another consideration is network speed, since MTurkers were participating in the [15] study over the Internet. In contrast, our study was conducted with most participants being in close network proximity. Therefore, it would explain why Mturk results are slower since they can originate anywhere in the world (according to demographics reported in [15]). This may also skew our results to be faster than the actual total reCAPTCHAv2 solving time.

Searles et al. [15] showed that participants' awareness of the true purpose of the study can affect solving times. In [15], participants were split into two groups: (1) Contextualized: these were told that they were participating in an account creation study and solved CAPTCHAS after submitting an account creation form, and (2) Direct: these were directly asked to solve CAPTCHAS as part of a user study. The solving time of the contextualized group was up to 57.5% slower than the solving time of the direct group. This can be correlated with the solving time of the first attempt in our study. On the first attempt, our participants were not aware of the fact that they would have to solve a reCAPTCHAv2 CAPTCHA and consequently had the slowest mean solving time for checkbox CAPTCHA.

Participants in [15] rated reCAPTCHAv2 on a Likert scale, from "least enjoyable" to "most enjoyable". Results showed that checkbox CAPTCHA was the most, and image CAPTCHA–the least, enjoyable. The term "enjoyable" is synonymous with pleasant (the opposite of "annoying"), which presents a point of comparison. Our results in Figures 8 and 9 are very similar in terms of positive and negative responses, thereby confirming the results of [15]

VI. Discussion

A. Cost Analysis

We now attempt to quantify various costs incurred by the global use of reCAPTCHA. In particular, we want to estimate the total time spent solving reCAPTCHA CAPTCHAS, the overall amount of human labor, network traffic (bandwidth), power consumption, and the environmental impact. Note that, in the informal analysis below, we consider all estimates to be rather generous lower bounds. One major caveat is that the authors of this paper are not trained in economics. Thus, from a trained economist's viewpoint, our analysis would (should?) be viewed as amateurish.

The historic average solving time for distorted-text CAPTCHAS (the same type used by reCAPTCHAv1) was 9.8 seconds. Using a conservative estimate of 100 million reCAPTCHA CAPTCHAS solved per day [4], about 980 million seconds per day were spent solving reCAPTCHAv1 CAPTCHAS. For reCAPTCHAv1, which was used for 5 years (2009-2014), this amounts to 183 billion reCAPTCHAv1 sessions, taking 1.79 trillion seconds, which translates into 497 million hours of human time. Given the U.S. federal minimum wage of $7.5, this roughly yields $3.7 billion in free wages.

The average solving time for all reCAPTCHAv2 CAPTCHAS is 3.53 seconds. Following a conservative estimate of 100 million reCAPTCHA CAPTCHAS being solved per day [4], 353 million seconds per day are spent solving reCAPTCHAv2 CAPTCHAS. reCAPTCHAv2 was in use for 9 years, amounting to 329 billion reCAPTCHAv2 sessions, taking 1.16 trillion seconds, or 322 million hours of human time. Again, using $7.5 per hour as the U.S. minimum wage, which roughly yields $2.4 billion in free wages.

Assuming un-cached scenarios from our technical analysis (see Appendix B), network bandwidth overhead is 408 KB per CAPTCHA session. This translates into 134 trillion KB or 134 petabytes ($194x1024$ terabytes) of bandwidth. A recent (2017) survey [54] estimated that the cost of energy for network data transmission was 0.06 kWh/GB (kilowatt hours per gigabyte). Based on this rate, we can estimate that 7.5 million kWh of energy was used on just the network transmission of reCAPTCHA data. This does not include client or server-related energy costs. Based on the rates provided by the US Environmental Protection Agency (EPA) [55] and US Energy Information Administration (EIA) [56], 1 kWh roughly equals $1 - 2.4$ pounds of $CO2$ pollution. Therefore, reCAPTCHA bandwidth consumption alone produced in the range of $7.5 - 18$ million pounds of CO2 pollution over 9 years.

Combining reCAPTCHAv1 and reCAPTCHAv2, there have been at least 512 billion reCAPTCHA sessions taking 2.95 trillion seconds, or 819 million hours, which is at least $6.1 billion USD in free wages. Out of the 329 billion reCAPTCHAv2 sessions, following our estimate of 20% of total sessions being image CAPTCHAS, at least 65.8 billion would have been image CAPTCHAS, and 263.2 – checkbox CAPTCHAS. Thus, 250 billion CAPTCHAS would have resulted in labeled data. According to Google, the value of $1,000$ items of labeled data is in the $$35 - 129$ USD range [57], which would be worth at least $$8.75 - 32.3$ billion per sale.

Finally, we consider the economics of tracking cookies, another by-product of reCAPTCHA. Tracking cookies play an ever-increasing role in the rapidly growing online advertisement market. According to Forbes [58], digital ad spending reached over $491 billion globally in 2021, and over half of the market (51%) heavily relied on third-party cookies for advertisement strategies [59]. The expenditure on third-party audience data (collected using tracking cookies) in the United States jumped from $15.9 billion in 2017 to $22 billion in 2021 [60]. More concretely, the current average value of a cookie throughout its lifetime is EUR 2.52 or $2.7 [61]. Given that there have been at least 329 billion reCAPTCHAv2 sessions that created tracking cookies, the estimated value of all those cookies is about $888 billion dollars.

B. Security Analysis

Sections VI-C and VI-D below discuss some successful attacks against reCAPTCHA. Table XIII shows a direct comparison of time and accuracy for humans and bots.

TABLE XIII: Humans vs. bot solving time (seconds) and accuracy (percentage) for reCAPTCHAv2

	Human				Bot	
Type	Time	Acc	Time	Acc	Time	Acc
checkbox	1.85	80%	3.1-4.9 [15]	85% [15]	1.4 [32]	100% [32]
image	10.4	93%	16-26 [15]	81% [15]	17.5 [37]	85% [37]

C. reCAPTCHAv2

reCAPTCHAv2 presents three types of captcha challenges: behavior-based (checkbox), image, and audio. Unfortunately, each type has been shown to be vulnerable to attacks.

1) Checkbox CAPTCHA*:* Soon after the introduction of reCAPTCHAv2, an effective exploit, called click-jacking [31], appeared. Click-jacking allows an adversary to force regular (benign) users in generating g-recaptcha-response-s (cookies), which can then be automatically used to pass challenges, ultimately making a bot's job much easier.

Sivakorn et al. [32] performed an in-depth analysis of the risk analysis system of reCAPTCHA and implemented an attack to manipulate it. Based on that analysis and its implementation:

1) Google primarily uses tracking cookies in the risk analysis system.
2) At least 63,000 valid cookies can be automatically created per day per IP address.
3) 9 days after a cookie creation, checkbox attempts using the cookie will succeed.
4) 52,000-59,000 checkbox CAPTCHAS can be solved with 100% accuracy per day per IP address.
5) The average solution time is 1.4 seconds with 100% accuracy, shown in Table XIII.

Given its blatant vulnerability [31], ease of implementing large-scale automation [32], and usage of privacy-invasive tracking cookies, reCAPTCHAv2 checkbox is hardly a real security tool. Considering Google's recent history (e.g., being sued for 22.5 million for **secretly** adding tracking cookies to Apple devices [62]), it is not far-fetched to conclude that the true purpose of reCAPTCHAv2 is to be a tracking cookie farm for advertising profit, masquerading as a security service.

2) Image CAPTCHA*:* Image-labeling CAPTCHAS appeared around 2004, after the introduction of Image Recognition CAPTCHAS by Chew et al. [63]. Six years later, in 2010, Fritsch et al. [33] published an attack that beat the prevalent image CAPTCHAS of the time with 100% accuracy. At this point, image recognition was no longer difficult to solve automatically with a computer. However, with the introduction of reCAPTCHAv2 in 2014, the fall-back security method became an image CAPTCHA, which was shown to be insecure four years earlier. The idea was that, if one's cookies are not valuable enough, reCAPTCHAv2 would present an image labeling task. Though this does not make sense as a security service, it does make sense considering that labeled image data is highly valuable and is in fact sold by Google [57]. One natural conclusion is that reCAPTCHAv2 represents an image-labeling free-labor and tracking cookie farm, pretending to be a security service.

Sivakorn et al. [32] and Hossen et al. [37] successfully developed automated solutions for reCAPTCHAv2 image CAPTCHAS. In 2016, Sivakorn et al. [32] showed that a plethora of automated services, including Google's own Google Reverse Image Search (GRIS), can solve reCAPTCHAv2 image CAPTCHAS without human involvement. [32] also implemented an easy solver for reCAPTCHAv2 image CAPTCHAS, with 70.8% accuracy and 19.2sec average solving time. In 2020, Hossen et al. [37] again showed that many automated services, including Google's own Google Cloud Vision, can be used to automatically solve reCAPTCHAv2 image CAPTCHAS with reasonable speed and accuracy. [37] also implemented an automatic solver, that is fast (17.5sec) and offers high accuracy (85%), as shown in Table XIII.

3) Audio CAPTCHA*:* As part of reCAPTCHAv2, Google introduced accessibility options allowing audio CAPTCHAS, instead of image-based ones. Unsurprisingly, these audio CAPTCHAS introduce an accessibility side-channel, especially effective due to major advances in speech-to-text technology.

In 2017, Bock et al. [64] introduced an automated system called *unCaptcha* which can solve audio challenges with 85.15% accuracy and 5.42 seconds average solving time. Similar to other attacks, [64] uses Google's own voice recognition technology as a means to break audio CAPTCHAS.

D. reCAPTCHAv3

reCAPTCHAv3 was introduced in 2018 [65]. It proposed returning a score for website developers to use to decide whether to follow up with a CAPTCHA or perform some other action. CAPTCHAS served by reCAPTCHAv3 are the same as those of reCAPTCHAv2. Also, there is no discernible difference between reCAPTCHAv2 and reCAPTCHAv3 in terms of the appearance or perception of image and audio CAPTCHAS. Hence, attacks targeting reCAPTCHAv2 image and audio CAPTCHAS are also applicable for those of reCAPTCHAv3. However, assuming that the risk analysis system was updated from reCAPTCHAv2 to reCAPTCHAv3, breaking behavior-based CAPTCHAS of reCAPTCHAv3 might require new techniques. However, In 2019, Akrou et al. [66] presented an attack based on Reinforcement Learning (RL), breaking reCAPTCHAv3's behavior-based CAPTCHAS. It obtained a high scores $(.9+)$, with 97% accuracy and only required $2,000$ data points as a training set.

VII. Summary

Over 13 years have passed since reCAPTCHA's initial appearance and its current prevalence is undeniable. It is thus both timely and important to investigate its usability. This paper presents a real-world user study with over $3,600$ unbiased (unwitting) participants solving over $9,000$ reCAPTCHAv2 challenges. We explore four new dimensions of reCAPTCHAv2 solving time: # of attempts, service type, as well as educational level and major. Results show that:

- Participants improve in terms of solving time with more attempts, for checkbox CAPTCHAS.
- The service/website setting is an important consideration for researchers and web developers since it has a statistically significant effect on solving time.
- Educational level directly impacts solving time.
- There were minor trends with statistical significance based on participants' majors, e.g., solving times of participants with technical (STEM) majors were faster than that of others.

In terms of usability, the post-study survey results show that the checkbox CAPTCHA earns an average SUS score of 78. This is considered as acceptable and preferred by many participants over the image CAPTCHA, which has an average SUS score of 58. Notably, participants found the image CAPTCHA to be annoying.

In terms of cost, we estimate that – during the entire period of its deployment – 819 million hours of human time has been spent on reCAPTCHA, which corresponds to at least \$6.1 billion in wages. Traffic resulting from reCAPTCHA consumed about 134 Petabytes of bandwidth, which translates into about 7.5 million kWhs of energy, corresponding to 7.5 million pounds of CO2. In addition, Google potentially profited \$888 billion from cookies and $\$8.75 - 32.3$ billion from sales of the labeled data set.

In terms of security reCAPTCHAv2 presents:

- Click-jacking (a blatant vulnerability) [31]
- Trivial implementation of large-scale automation attacks [32]
- Weakness of security premise of fallback (image CAPTCHA) [32], [33], [37]
- Usage of privacy-invasive tracking cookies (for security) [32]

Ultimately, given these points, it can be concluded that reCAPTCHAv2 presents no real security.

Given that: (1) reCAPTCHAv2 is negatively perceived by most users, (2) its immense cost, and (3) its susceptibility to bots, our results prompt a natural conclusion:

> *reCAPTCHAv2 and similar reCAPTCHA technology should be deprecated.*

Acknowledgements: We thank USEC'25 reviewers for their constructive feedback. This work was supported in part by funding from the NSF Award SATC-1956393.

References

[1] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford, "CAPTCHA: Using Hard AI Problems for Security," in *Advances in Cryptology — EUROCRYPT 2003*, E. Biham, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 294–311.

[2] L. von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum, "recaptcha: Human-based character recognition via web security measures," *Science*, vol. 321, no. 5895, pp. 1465–1468, 2008. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.1160379

[3] "Teaching computers to read: Google acquires recaptcha," Sep 2009. [Online]. Available: https://web.archive.org/web/20120511113750/http://googleblog.blogspot.com/2009/09/teaching-computers-to-read-google.html

[4] "recaptcha faq from 2010 archived," 2010. [Online]. Available: https://web.archive.org/web/20100629174402/http://www.google.com:80/recaptcha/faq

[5] J. Yan and A. S. El Ahmad, "A Low-cost Attack on a Microsoft CAPTCHA," in *Proceedings of the 15th ACM conference on Computer and communications security*, 2008, pp. 543–554.

[6] H. Gao, W. Wang, and Y. Fan, "Divide and conquer: an efficient attack on Yahoo! CAPTCHA," in *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*. IEEE, 2012, pp. 9–16.

[7] C. J. Hernandez-Castro and A. Ribagorda, "Pitfalls in CAPTCHA design and implementation: The Math CAPTCHA, a case study," *Computers & Security*, vol. 29, no. 1, pp. 141–157, 2010.

[8] I. J. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud, and V. Shet, "Multi-digit number recognition from street view imagery using deep convolutional neural networks," *arXiv preprint arXiv:1312.6082*, 2014.

[9] V. Shet, "Street View and reCAPTCHA technology just got smarter," https://security.googleblog.com/2014/04/street-view-and-recaptcha-technology.html, 2014.

[10] S. Perez, "Google now using recaptcha to decode street view addresses," Mar 2012. [Online]. Available: https://techcrunch.com/2012/03/29/google-now-using-recaptcha-to-decode-street-view-addresses/

[11] "Are you a robot? Introducing "No CAPTCHA reCAPTCHA"," https://security.googleblog.com/2014/12/are-you-robot-introducing-no-captcha.html, 2014.

[12] "reCAPTCHA v2," https://developers.google.com/recaptcha/docs/display, 2023.

[13] S. Sivakorn, "I'm not a human: Breaking the google recaptcha," 2016.

[14] "CAPTCHA Usage Distribution on the Entire Internet," https://trends.builtwith.com/widgets/captcha/traffic/Entire-Internet, 2023.

[15] A. Searles, Y. Nakatsuka, E. Ozturk, A. Paverd, G. Tsudik, and A. Enkoji, "An empirical study & evaluation of modern CAPTCHAs," in *32nd USENIX Security Symposium (USENIX Security 23)*. Anaheim, CA: USENIX Association, Aug. 2023, pp. 3081–3097. [Online]. Available: https://www.usenix.org/conference/usenixsecurity23/presentation/searles

[16] E. Bursztein, S. Bethard, C. Fabry, J. C. Mitchell, and D. Jurafsky, "How Good Are Humans at Solving CAPTCHAs? A Large Scale Evaluation," in *2010 IEEE Symposium on Security and Privacy*, 2010, pp. 399–413.

[17] J. P. Bigham and A. Cavender, "Evaluating Existing Audio CAPTCHAs and an Interface Optimized for Non-Visual Use," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '09. New York, NY, USA: ACM, 2009, p. 1829–1838.

[18] H. Gao, D. Yao, H. Liu, X. Liu, and L. Wang, "A Novel Image Based CAPTCHA Using Jigsaw Puzzle," in *2010 13th IEEE International Conference on Computational Science and Engineering*, 2010, pp. 351–356.

[19] S. A. Ross, J. A. Halderman, and A. Finkelstein, "Sketcha: A Captcha Based on Line Drawings of 3D Models," in *Proceedings of the 19th International Conference on World Wide Web*. New York, NY, USA: ACM, 2010, p. 821–830.

[20] E. Uzun, S. Chung, I. Essa, and W. Lee, "rtCaptcha: A Real-Time Captcha Based Liveness Detection System," in *Network and Distributed System Security Symposium (NDSS)*, San Diego, California, United States, 02 2018.

[21] M. Mohamed, S. Gao, N. Saxena, and C. Zhang, "Dynamic Cognitive Game CAPTCHA Usability and Detection of Streaming-Based Farming," in *2014 NDSS Workshop on Usable Security*, 2014, pp. 1–10.

[22] M. Jain, R. Tripathi, I. Bhansali, and P. Kumar, "Automatic Generation and Evaluation of Usable and Secure Audio ReCAPTCHA," in *The 21st International ACM SIGACCESS Conference on Computers and Accessibility*, ser. ASSETS '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 355–366. [Online]. Available: https://doi.org/10.1145/3308561.3353777

[23] S. Gao, M. Mohamed, N. Saxena, and C. Zhang, "Emerging-Image Motion CAPTCHAs: Vulnerabilities of Existing Designs, and Countermeasures," *IEEE Transactions on Dependable and Secure Computing*, vol. 16, no. 6, pp. 1040–1053, 2019.

[24] K. Krol, S. Parkin, and M. A. Sasse, "Better the Devil You Know: A User Study of Two CAPTCHAs and a Possible Replacement Technology," in *2016 NDSS Workshop on Usable Security*, 2016, pp. 1–10.

[25] C. A. Fidas, A. G. Voyiatzis, and N. M. Avouris, "On the Necessity of User-Friendly CAPTCHA," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '11. New York, NY, USA: ACM, 2011, p. 2623–2626.

[26] Y. Feng, Q. Cao, H. Qi, and S. Ruoti, "Sencaptcha: A mobile-first captcha using orientation sensors," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 4, no. 2, jun 2020. [Online]. Available: https://doi.org/10.1145/3397312

[27] C.-J. Ho, C.-C. Wu, K.-T. Chen, and C.-L. Lei, "DevilTyper: A Game for CAPTCHA Usability Evaluation," *Comput. Entertain.*, vol. 9, no. 1, apr 2011.

[28] N. Tanthavech and A. Nimkoompai, "Captcha: Impact of website security on user experience," *ICIIT '19: Proceedings of the 2019 4th International Conference on Intelligent Information Technology*, pp. 37–41, 02 2019.

[29] "Amazon Mechanical Turk," https://www.mturk.com/, 2023.

[30] M. A. Webb and J. P. Tangney, "Too good to be true: Bots and bad data from mechanical turk," *Perspectives on Psychological Science*, 2022.

[31] E. Homakov, "The no captcha problem," 2014. [Online]. Available: https://homakov.blogspot.com/2014/12/the-no-captcha-problem.html

[32] S. Sivakorn, I. Polakis, and A. D. Keromytis, "I am Robot: (Deep) Learning to Break Semantic Image CAPTCHAs," in *2016 IEEE European Symposium on Security and Privacy (EuroS P)*, 2016, pp. 388–403.

[33] C. Fritsch, M. Netter, A. Reisser, and G. Pernul, "Attacking image recognition captchas," in *Trust, Privacy and Security in Digital Business*, S. Katsikas, J. Lopez, and M. Soriano, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 13–25.

[34] M. Guerar, L. Verderame, M. Migliardi, F. Palmieri, and A. Merlo, "Gotta CAPTCHA 'Em All: A Survey of Twenty years of the Human-or-Computer Dilemma," *CoRR*, vol. abs/2103.01748, 2021. [Online]. Available: https://arxiv.org/abs/2103.01748

[35] "hCaptcha," https://www.hcaptcha.com/, 2023.

[36] "reCAPTCHA," https://www.google.com/recaptcha/about/, 2023.

[37] M. I. Hossen, Y. Tu, M. F. Rabby, M. N. Islam, H. Cao, and X. Hei, "An Object Detection based Solver for Google's Image reCAPTCHA v2," in *23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020)*. San Sebastian: USENIX Association, Oct. 2020, pp. 269–284.

[38] M. I. Hossen and X. Hei, "A Low-Cost Attack against the hCaptcha System," *CoRR*, vol. abs/2104.04683, 2021. [Online]. Available: https://arxiv.org/abs/2104.04683

[39] F. H. Alqahtani and F. A. Alsulaiman, "Is image-based CAPTCHA secure against attacks based on machine learning? An experimental study," *Computers & Security*, vol. 88, p. 101635, 2020.

[40] H. Weng, B. Zhao, S. Ji, J. Chen, T. Wang, Q. He, and R. Beyah, "Towards understanding the security of modern image captchas and underground captcha-solving services," *Big Data Mining and Analytics*, vol. 2, no. 2, pp. 118–144, 2019.

[41] D. Lorenzi, J. Vaidya, E. Uzun, S. Sural, and V. Atluri, "Attacking Image Based CAPTCHAs Using Image Recognition Techniques," in *Information Systems Security*, V. Venkatakrishnan and D. Goswami, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 327–342.

[42] J. Brooke *et al.*, "Sus-a quick and dirty usability scale," *Usability evaluation in industry*, vol. 189, no. 194, pp. 4–7, 1996.

[43] P. T. Kortum and A. Bangor, "Usability ratings for everyday products measured with the system usability scale," *International Journal of Human-Computer Interaction*, vol. 29, no. 2, pp. 67–76, 2013.

[44] J. Liang, D. Xian, X. Liu, J. Fu, X. Zhang, B. Tang, J. Lei *et al.*, "Usability study of mainstream wearable fitness devices: feature analysis and system usability scale evaluation," *JMIR mHealth and uHealth*, vol. 6, no. 11, p. e11066, 2018.

[45] A. Kaya, R. Ozturk, and C. Altin Gumussoy, “Usability measurement of mobile applications with system usability scale (sus),” in *Industrial Engineering in the Big Data Era: Selected Papers from the Global Joint Conference on Industrial Engineering and Its Application Areas, GJCIE 2018, June 21–22, 2018, Nevsehir, Turkey*. Springer, 2019, pp. 389–400.

[46] P. Vlachogianni and N. Tselios, “Perceived usability evaluation of educational technology using the system usability scale (sus): A systematic review,” *Journal of Research on Technology in Education*, vol. 54, no. 3, pp. 392–409, 2022.

[47] D. Pal and V. Vanijja, “Perceived usability evaluation of microsoft teams as an online learning platform during covid-19 using system usability scale and technology acceptance model in india,” *Children and youth services review*, vol. 119, p. 105535, 2020.

[48] B. Klug, “An overview of the system usability scale in library website and system usability testing,” *Weave: Journal of Library User Experience*, vol. 1, no. 6, 2017.

[49] Y. Feng, Q. Cao, H. Qi, and S. Ruoti, “Sencaptcha: A mobile-first captcha using orientation sensors,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 4, no. 2, pp. 1–26, 2020.

[50] A. Bangor, P. Kortum, and J. Miller, “Determining what individual sus scores mean: Adding an adjective rating scale,” *Journal of usability studies*, vol. 4, no. 3, pp. 114–123, 2009.

[51] “Google recaptcha admin dashboard,” 2023. [Online]. Available: https://www.google.com/u/2/recaptcha/admin/site

[52] “Scipy is an open-source software for mathematics, science, and engineering.” 2023. [Online]. Available: https://scipy.org/

[53] “The post study survey via google forms (interactive version) do not submit personal info,” 2023. [Online]. Available: https://docs.google.com/forms/d/e/1FAIpQLSejdihyw2z3YpjxTYMXeOTrn6ZC8Az6ockPm4b9lbhsvQ77gg/viewform?usp=sf_link

[54] J. Aslan, K. Mayers, J. G. Koomey, and C. France, “Electricity intensity of internet data transmission: Untangling the estimates,” *Journal of Industrial Ecology*, vol. 22, no. 4, pp. 785–798, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/jiec.12630

[55] “Usa environmental protection agency greenhouse gas calculator,” 2023. [Online]. Available: https://www.epa.gov/energy/greenhouse-gas-equivalencies-calculator

[56] “Energy information administration faq,” 2023. [Online]. Available: https://www.eia.gov/tools/faqs/faq.php?id=74&t=11

[57] “Ai platform data labeling service pricing,” 2023. [Online]. Available: https://cloud.google.com/ai-platform/data-labeling/pricing

[58] “The Truth In User Privacy And Targeted Ads,” https://www.forbes.com/sites/forbestechcouncil/2022/02/24/the-truth-in-user-privacy-and-targeted-ads/.

[59] “Degree of reliance on third-party cookies in digital advertising in the United States as of July 2021,” https://www.statista.com/statistics/1222230/reliance-cookie-advertising-usa/.

[60] “Spending on third-party audience data supporting marketing related efforts in the United States from 2017 to 2021, by type,” https://www.statista.com/statistics/1202754/third-party-audience-data-spending-usa/.

[61] K. M. Miller and B. Skiera, “Economic consequences of online tracking restrictions: Evidence from cookies,” *International Journal of Research in Marketing*, 2023.

[62] “Google will pay 22.5 million to settle ftc charges it misrepresented privacy assurances to users of apple’s safari internet browser,” Feb 2019. [Online]. Available: https://www.ftc.gov/news-events/news/press-releases/2012/08/google-will-pay-225-million-settle-ftc-charges-it-misrepresented-privacy-assurances-users-apples

[63] M. Chew and J. D. Tygar, “Image recognition captchas,” in *Information Security*, K. Zhang and Y. Zheng, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 268–279.

[64] K. Bock, D. Patel, G. Hughey, and D. Levin, “unCaptcha: A Low-Resource Defeat of reCaptcha’s Audio Challenge,” in *11th USENIX Workshop on Offensive Technologies (WOOT 17)*. Vancouver, BC: USENIX Association, Aug. 2017.

[65] “reCAPTCHA v3,” https://developers.google.com/search/blog/2018/10/introducing-recaptcha-v3-new-way-to, 2018.

[66] I. Akrout, A. Feriani, and M. Akrout, “Hacking google recaptcha v3 using reinforcement learning,” *arXiv preprint arXiv:1903.01003*, 2019.

[67] “Google chrome,” 2023. [Online]. Available: https://www.google.com/chrome/

[68] “Solarwinds pingdom,” 2023. [Online]. Available: https://www.pingdom.com/

[69] “Webpagetest,” 2023. [Online]. Available: https://www.webpagetest.org/

[70] 2023. [Online]. Available: https://www.jitbit.com/macro-recorder/mouse-recorder/

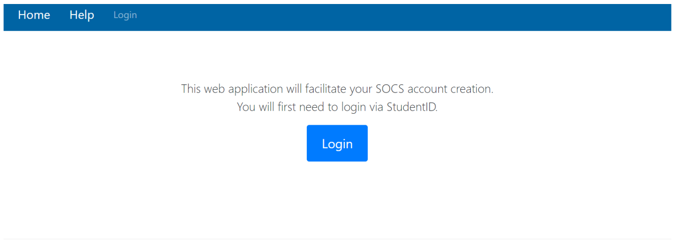
[71] 2023. [Online]. Available: https://playwright.dev/

APPENDIX A: SICS Website Workflow

This appendix contains basic workflows for the account creation and password recovery processes that participants followed in the user study.

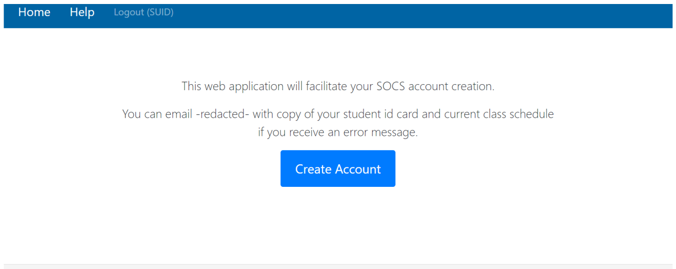
A. Account Creation

Figures 12, 13, 14, 15 constitute the workflow of the account creation process.



This web application will facilitate your SOCS account creation.
You will first need to login via StudentID.

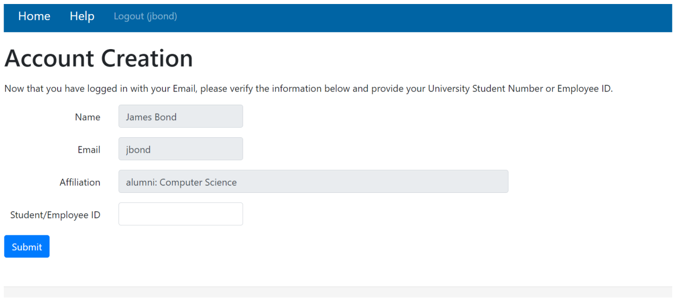
Fig. 12: Initial login page



This web application will facilitate your SOCS account creation.

You can email -redacted- with copy of your student id card and current class schedule
if you receive an error message.

Fig. 13: Initial account creation page



Account Creation

Now that you have logged in with your Email, please verify the information below and provide your University Student Number or Employee ID.

Name James Bond

Email jbond

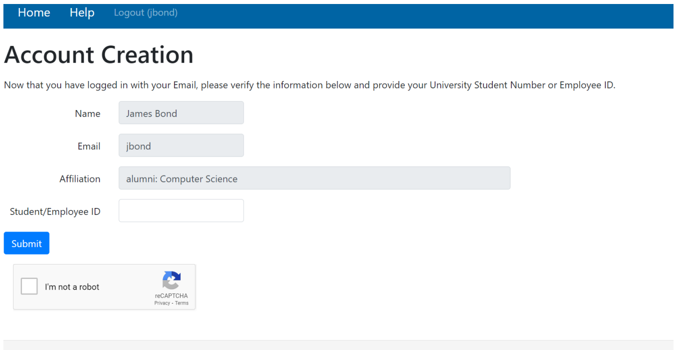
Affiliation alumni: Computer Science

Student/Employee ID

Fig. 14: Account creation form

B. Password Recovery

Figures 16 and 17 present the workflow of the password recovery process.



Home Help Logout (jbond)

Account Creation

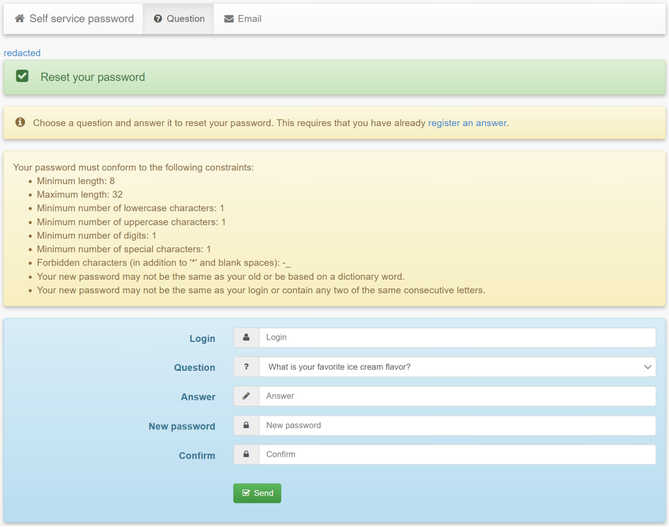
Now that you have logged in with your Email, please verify the information below and provide your University Student Number or Employee ID.

Name	James Bond
Email	jbond
Affiliation	alumni: Computer Science
Student/Employee ID	

Submit

□ I'm not a robot

Fig. 15: Account creation form after clicking submit



redacted

☑ Reset your password

Choose a question and answer it to reset your password. This requires that you have already register an answer.

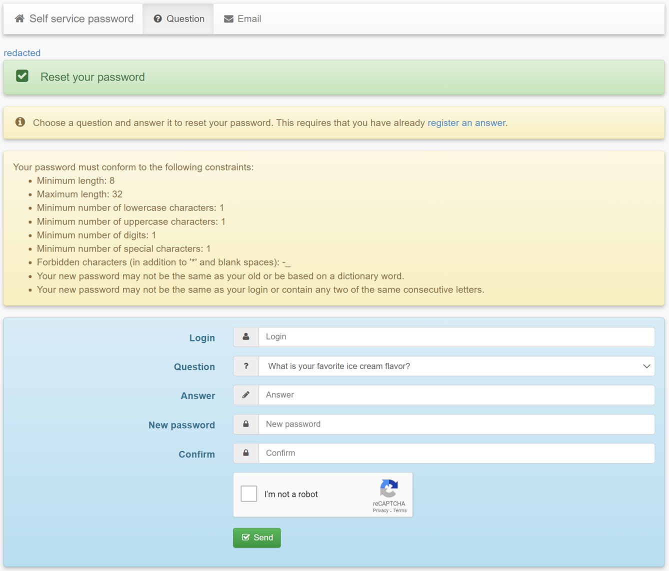
Your password must conform to the following constraints:

- Minimum length: 8
- Maximum length: 32
- Minimum number of lowercase characters: 1
- Minimum number of uppercase characters: 1
- Minimum number of digits: 1
- Minimum number of special characters: 1
- Forbidden characters (in addition to '*' and blank spaces): -_
- Your new password may not be the same as your old or be based on a dictionary word.
- Your new password may not be the same as your login or contain any two of the same consecutive letters.

Login	Login
Question	What is your favorite ice cream flavor?
Answer	Answer
New password	New password
Confirm	Confirm

Send

Fig. 16: Password recovery form



redacted

☑ Reset your password

Choose a question and answer it to reset your password. This requires that you have already register an answer.

Your password must conform to the following constraints:

- Minimum length: 8
- Maximum length: 32
- Minimum number of lowercase characters: 1
- Minimum number of uppercase characters: 1
- Minimum number of digits: 1
- Minimum number of special characters: 1
- Forbidden characters (in addition to '*' and blank spaces): -_
- Your new password may not be the same as your old or be based on a dictionary word.
- Your new password may not be the same as your login or contain any two of the same consecutive letters.

Login	Login
Question	What is your favorite ice cream flavor?
Answer	Answer
New password	New password
Confirm	Confirm

□ I'm not a robot

Send

Fig. 17: Password recovery form after clicking submit

APPENDIX B: NETWORK ANALYSIS OF reCAPTCHAv2

```
<html>
<head> <title> Simple Web Page </title> </head>
<body> <h4> A minimal web page </h4> <br/> </body>
</html>
```

Fig. 18: Source code of simple.html

```
<html>
<head> <title> reCAPTCHA Difficult </title>
<script src=”https://www.google.com/recaptcha/
 api.js” async defer></script>
</head>
<body>
<h4>A minimal web page</h4>
<div class=”g-recaptcha”
    data-sitekey=”obtained-site-key”></div>  <br/>
</body>
</html>
```

Fig. 19: Source code of recaptcha.html

This Appendix contains a high-level technical analysis of reCAPTCHAv2. It has been considered in [32], which described the display method and workflow of reCAPTCHAv2 with an emphasis on security aspects. Whereas, our goal is to: (1) determine various overhead factors incurred whenever a web page uses reCAPTCHA, and (2) investigate reCAPTCHAv2's automation detection capability. To this end, we performed black box program and network traffic analyses for common usage scenarios. We used two simple web pages for this purpose:

- Baseline page without any CAPTCHAS. This page is called *simple.html* and its source code is shown in Figure 18.
- A page similar to the baseline page, except with an additional reCAPTCHAv2. This page is called recaptcha.html and its source code is shown in Figure 19. As evident from the figure, integrating reCAPTCHAv2 into a web page is straightforward.

These pages were visited using Google Chrome browser [67] and each usage scenario was repeated at least ten times. Browsing was performed in both guest and normal (profile logged-in) modes. Relevant information in the format of a *.har* file was collected for each scenario using Chrome DevTools.

The rest of this section describes the findings. Notations are summarized in Table XIV.

TABLE XIV: Notation Summary

Notation	Description
g1	https://www.google.com/recaptcha
g2	https://www.gstatic.com/recaptcha/releases/vkGiR-M4noX1963Xi_DB0JeI
g3	https://www.gstatic.com/recaptcha/api2/
g4	https://www.google.com/recaptcha/api2/
g5	https://fonts.gstatic.com/s/roboto/v18/
dv	different values

C. Page load Latency

Table XV shows additional API calls made while loading *recaptcha.html* webpage.

TABLE XV: reCAPTCHAv2 API Calls during page load

Request URL	Content-Length (B)
g1/api.js	554
g2/recaptcha__en.js	166822
g4/anchor?ar=[dv]	27864 (average)
g2/styles__ltr.css	24605
g2/recaptcha__en.js	166822
g3/logo_48.png	2228
g4/webworker.js?hl=[dv]	112
g2/recaptcha__en.js	166822
g4/bframe?hl=[dv] &v=[dv]&k=[dv]	1141-1145
g2/styles__ltr.css	24605
g2/recaptcha__en.js	166822
Network Overhead	254.01 KB-316.64KB

There are also 2-to-6 calls to g5 for downloading various web fonts. Content length for each call is 15340, 15344, and 15552 bytes. Even though multiple calls are made to download recaptcha__en.js and styles__ltr.css, only the first call downloads the file, if necessary. These observations are taken into account when computing network overhead in Table XV.

Moreover, api.js, recaptcha__en.js, styles__ltr.css, logo_48.png, and web fonts are often served from the cache. Table XV provides an upper bound on network overhead for page load. Average network overhead is computed by extracting actual network transmission during page load from collected *.har* files. Table XVI shows the results.

TABLE XVI: recaptcha.html load network overhead

Scenario	Page Name	Page Size(KB)
First load	simple.html	0.631KB
First load	recaptcha.html	408.5KB
Network Overhead		407.869KB
Subsequent loads	simple.html	0.241 KB
Subsequent loads	recaptcha.html	29.56 KB
Network overhead		29.319 KB

We investigated load latency using Chrome DevTools, pingdom.com [68], and webpagetest.com [69]. Table XVII presents the results. Latency computed using Chrome DevTools is the highest since Chrome DevTools determines the load time of simple.html and recaptcha.html in the same network where the concerned web pages are hosted. Observation shows that load latency increases as the distance between the user and the hosted webpage decreases (in terms of hops).

D. Checkbox Click Overhead

Table XVIII shows additional API calls made after checkbox is clicked. In this scenario, image CAPTCHA is not served to the user.

TABLE XVII: recaptcha.html load latency

Measurement Tool	Page Name	load Time
Chrome DevTools	simple.html	51.16ms
Chrome DevTools	recaptcha.html	425.81ms
Time Overhead		374.65ms, 732.31%
pingdom.com	simple.html	375ms
pingdom.com	recaptcha.html	796ms
Time Overhead		471ms, 125.6%
webpagetest.org	simple.html	814.22ms
Subsequent Loads	recaptcha.html	2074.78ms
Latency		1260.56ms, 154.82%

TABLE XVIII: reCAPTCHAv2 API Calls after checkbox click

Request URL	Content-Length (B)
g4/reload?k=[dv]	23844.67 (average)
g4/userverify?k=[dv]	580.56 (average)
g3/refresh_2x.png	600
g3/audio_2x.png	530
g3/info_2x.png	665
g5/[font].woff2	15552
Network Overhead	24.43 KB-41.77KB

In some cases, only the first two calls are made. Even when other calls are made, files are normally served from the cache, so there is no network traffic. Files are downloaded only in the first-ever attempt to solve reCAPTCHAv2 in a given client browser. Table XVIII depicts the upper and lower bounds for the network overhead.

E. reCAPTCHAv2 Image load Overhead

Table XIX shows additional API calls made when checkbox is clicked and an image CAPTCHA is loaded. It also provides the upper and lower bounds of network overhead due to these calls.

TABLE XIX: reCAPTCHAv2 API Calls for image load

Request URL	Content-Length (B)
g4/reload?k=[dv]	24439.16667 (average)
g3/refresh_2x.png	600
g3/audio_2x.png	530
g3/info_2x.png	665
g4/payload?p=[dv]	39589.45455 (average)
Network Overhead	64.03 KB-96.72KB

In some cases, two calls are made to g5 to download web fonts; content length is 15340 and 15552 bytes, respectively. Also, refresh_2x.png, audio_2x.png, info_2x.png, and web fonts are often served from the cache instead of being downloaded.

F. Image Solution Verification Overhead

Table XX shows additional API calls made when an image CAPTCHA solution is verified. In case of a correct solution, only the third call from Table XX requires network transmission and thus incurs network overhead. In case of a wrong solution, the last call from Table XX is made, which requires network transmission and adds to network overhead. In both cases, other calls are usually served from the cache. In some instances, when a wrong solution occurs, only the third and fifth calls from Table XX are made.

TABLE XX: reCAPTCHAv2 API Calls for correct image solution

Case	Request URL	Content-Length (B)
Both	g3/refresh_2x.png	600
Both	g3/audio_2x.png	530
Both	g4/userverify?k=[dv]	595.88
Both	g3/info_2x.png	665
Wrong Solution	g4/payload?p=[dv]	40922.167 (average)
Correct Solution Network Overhead		0.6KB
Wrong Solution Network Overhead		41.58KB

TABLE XXI: reCAPTCHAv2 API Calls for reCAPTCHAv2 expiration

Request URL	Content-Length (B)
g4/anchor?ar=[dv]	27864 (average)
g2/styles__ltr.css	24605
g2/recaptcha__en.js	166822
g3/logo_48.png	2228
g4/webworker.js?hl=[dv]	112
g2/recaptcha__en.js	166822
g4/bframe?hl=[dv] &v=[dv]&k=[dv]	1141-1145
g2/styles__ltr.css	24605
g2/recaptcha__en.js	166822
Network Overhead	29KB

G. reCAPTCHAv2 Expiration Overhead

Table XXI shows additional API calls made after a reCAPTCHAv2 solution expires. Only the first and seventh calls (g4/anchor and g4/bframe) require network transmission and are considered for network overhead. Other calls are served from the cache.

Summary: Results of evaluating network overhead for various reCAPTCHAv2 usage scenarios are summarized in Table XXII. As evident from these results, using reCAPTCHAv2 incurs considerable network and timing overhead.

TABLE XXII: Summary of reCAPTCHAv2 Network Overhead

Scenario	Network Overhead(KB)
First time Page Load	408.5
Subsequent Page Loads	29.319
Checkbox Click	24.43-41.77
Image Load	64.03-96.72
Image Correct Solution Verification	0.6
Image Wrong Solution Verification & New Image load	41.58
Solution Expiration	29

H. Automation Detection

Finally, we briefly looked into the automation detection capability of reCAPTCHAv2. Specifically, checkbox click is performed through Jitbit mouse macro recorder [70] and playwright automated headless Chrome browser [71]. Interestingly, the use of the mouse macro is not considered as suspicious bot activity by reCAPTCHAv2. When checkbox is clicked and the page is reloaded in quick succession, an image CAPTCHA is served on around 14 tries, regardless of whether the tasks were performed manually or via the mouse macro. However, performing the same tasks via Playwright Chrome browser is considered suspicious – an Image CAPTCHA is served upon the first request.