Re: CAPTCHAs – Understanding CAPTCHA-Solving Services in an Economic Context

Marti Motoyama, Kirill Levchenko, Chris Kanich, Damon McCoy,
Geoffrey M. Voelker and Stefan Savage
University of California, San Diego
{*mmotoyam, klevchen, ckanich, dlmccoy, voelker, savage*}*@cs.ucsd.edu*

Abstract

Reverse Turing tests, or CAPTCHAs, have become an ubiquitous defense used to protect open Web resources from being exploited at scale. An effective CAPTCHA resists existing mechanistic software solving, yet can be solved with high probability by a human being. In response, a robust solving ecosystem has emerged, reselling both automated solving technology and real-time human labor to bypass these protections. Thus, CAPTCHAs can increasingly be understood and evaluated in purely economic terms; the market price of a solution *vs* the monetizable value of the asset being protected. We examine the market-side of this question in depth, analyzing the behavior and dynamics of CAPTCHA-solving service providers, their price performance, and the underlying labor markets driving this economy.

1 Introduction

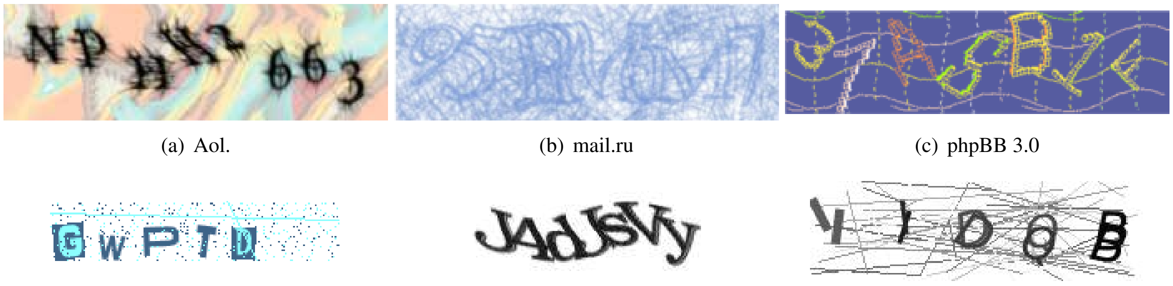
Questions of Internet security frequently reflect underlying economic forces that create both opportunities and incentives for exploitation. For example, much of today's Internet economy revolves around advertising revenue, and consequently, a vast array of services—including e-mail, social networking, blogging—are now available to new users on a basis that is both free and largely anonymous. The implicit compact underlying this model is that the users of these services are *individuals* and thus are effectively "paying" for services indirectly through their unique exposure to ad content. Unsurprisingly, attackers have sought to exploit this same freedom and acquire large numbers of resources under singular control, which can in turn be monetized (e.g., via thousands of free Web mail accounts for sourcing spam e-mail messages).

CAPTCHAs were developed as a means to limit the ability of attackers to scale their activities using automated means. In its most common implementation, a CAPTCHA consists of a visual challenge in the form of alphanumeric characters that are distorted in such a way that available computer vision algorithms have difficulty segmenting and recognizing the text. At the same time, humans, with some effort, have the ability to decipher the text and thus respond to the challenge correctly. Today, CAPTCHAs of various kinds are ubiquitously deployed for guarding account registration, comment posting, and so on.

This innovation has, in turn, attached value to the problem of solving CAPTCHAs and created an industrial market. Such commercial CAPTCHA solving comes in two varieties: automated solving and human labor. The first approach defines a technical arms race between those developing solving algorithms and those who develop ever more obfuscated CAPTCHA challenges in response. However, unlike similar arms races that revolve around spam or malware, we will argue that the underlying cost structure favors the defender, and consequently, the conscientious defender has largely won the war.

The second approach has been transformative, since the use of human labor to solve CAPTCHAs effectively side-steps their design point. Moreover, the combination of cheap Internet access and the commodity nature of today's CAPTCHAs has globalized the solving market; in fact, wholesale cost has dropped rapidly as providers have recruited workers from the lowest cost labor markets. Today, there are many service providers that can solve large numbers of CAPTCHAs via on-demand services with *retail* prices as low as $1 per thousand.

In either case, we argue that the security of CAPTCHAs can now be considered in an economic light. This property pits the underlying cost of CAPTCHA solving, either in amortized development time for software solvers or piece-meal in the global labor market, against the value of the asset it protects. While the very existence of CAPTCHA-solving services tells us that the value of the associated assets (e.g., an e-mail account) is worth more to some attackers than the cost of solving the CAPTCHA, the overall shape of the market is poorly understood. Ab-



(a) Aol. (b) mail.ru (c) phpBB 3.0

(d) Simple Machines Forum (e) Yahoo! (f) youku

Figure 1: Examples of CAPTCHAs from various Internet properties.

sent this understanding, it is difficult to reason about the security value that CAPTCHAs offer us.

This paper investigates this issue in depth and, where possible, on a empirical basis. We document the commercial evolution of automated solving tools (particularly via the successful Xrumer forum spamming package) and how they have been largely eclipsed by the emergence of the human-based CAPTCHA-solving market. To characterize this latter development, our approach is to engage the retail CAPTCHA-solving market on both the supply side and the demand side, as both a client and as "workers for hire." In addition to these empirical measurements, we also interviewed the owner and operator of a successful CAPTCHA-solving service (MR. E), who has provided us both validation and insight into the less visible aspects of the underlying business processes.[1] In the course of our analysis, we attempt to address key questions such as which CAPTCHAs are most heavily targeted, the rough solving capacity of the market leaders, the relationship of service quality to price, the impact of market transparency and arbitrage, the demographics of the underlying workforce and the adaptability of service offerings to changes in CAPTCHA content. We believe our findings, or at least our methodology, provide a context for reasoning about the net value provided by CAPTCHAs under existing threats and offer some directions for future development.

The remainder of this paper is organized as follows: Section 2 reviews CAPTCHA design and provides a qualitative history and overview of the CAPTCHA-solving ecosystem. Next, in Section 3 we empirically characterize two automated solver systems, the popular Xrumer package and a specialized reCaptcha solver. In Sections 4 and 5 we then characterize today's human-powered CAPTCHA-solving services, first describing our data collection approach and then presenting our experiments to measure key qualities such as response time, accuracy, and capacity. Section 6 describes the demographics of the CAPTCHA-solving labor pool. Finally, we discuss the implications of our results in Section 7 along with potential directions for future research.

2 Background

The term "CAPTCHA" was first introduced in 2000 by von Ahn *et al.* [21], describing a test that can differentiate humans from computers. Under common definitions [4], the test must be:

- Easily solved by humans,
- Easily generated and evaluated, but
- *Not* easily solved by computer.

Over the past decade, a number of different techniques for generating CAPTCHAS have been developed, each satisfying the properties described above to varying degrees. The most commonly found CAPTCHAS are visual challenges that require the user to identify alphanumeric characters present in an image obfuscated by some combination of noise and distortion.[2] Figure 1 shows examples of such visual CAPTCHAs. The basic challenge in designing these obfuscations is to make them easy enough that users are not dissuaded from attempting a solution, yet still too difficult to solve using available computer vision algorithms.

The issue of usability has been studied on a functional level—focusing on differences in expected accuracy and response time [3, 19, 22, 26]—but the ultimate effect of CAPTCHA difficulty on legitimate goal-oriented users is not well documented in the literature. That said, Elson *et al.* provide anecdotal evidence that "even relatively simple challenges can drive away a substantial number of po-

[1] By agreement, we do not identify MR. E or the particular service he runs. While we cannot validate all of his statements, when we tested his service empirically our results for measures such as response time, accuracy, capacity and labor makeup were consistent with his reports, supporting his veracity.

[2] There exists a range of non-textual and even non-visual CAPTCHAS that have been created but, excepting Microsoft's *Asirra* [9], we do not consider them here as they play a small role in the current CAPTCHA-solving ecosystem.

tential customers" [9], suggesting CAPTCHA design reflects a real trade-off between protection and usability.

The second challenge, defeating automation, has received far more attention and has kicked off a competition of sorts between those building ever more sophisticated algorithms for breaking CAPTCHAs and those creating new, more obfuscated CAPTCHAs in response [7, 11, 16, 17, 18, 25]. In the next section we examine this issue in more depth and explain why, for economic reasons, automated solving has been relegated to a niche status in the open market.

Finally, an alternative regime for solving CAPTCHAs is to outsource the problem to human workers. Indeed, this labor-based approach has been commoditized and today a broad range of providers operate to buy and sell CAPTCHA-solving service in bulk. We are by no means the first to identify the growth of this activity. In particular, Danchev provides an excellent overview of several CAPTCHA-solving services in his 2008 blog post "Inside India's CAPTCHA solving economy" [5]. We are, however, unaware of significant quantitative analysis of the solving ecosystem and its underlying economics. The closest work to our own is the complementary study of Bursztein *et al.* [3] which also uses active CAPTCHA-solving experiments, but is focused primarily on the issue of CAPTCHA difficulty rather than the underlying business models.

3 Automated Software Solvers

From the standpoint of an adversary, automated solving offers a number of clear advantages, including both near-zero marginal cost and near-infinite capacity. At a high level, automated CAPTCHA solving combines *segmentation* algorithms, designed to extract individual symbols from a distorted image, with basic optical character recognition (OCR) to identify the text present in CAPTCHAs. However, building such algorithms is complex (by definition, since CAPTCHAs are designed to evade existing vision techniques), and automated CAPTCHA solving often fails to replicate human accuracy. These constraints have in turn influenced the evolution of automated CAPTCHA solving as it transitioned from a mere academic contest to an issue of commercial viability.

3.1 Empirical Case Studies

We explore these issues empirically through two representative examples: Xrumer, a mature forum spamming tool with integrated support for solving a range of CAPTCHAs and reCaptchaOCR, a modern specialized solver that targets the popular reCaptcha service.

Xrumer

Xrumer [24] is a well-known forum spamming tool, widely described on "blackhat" SEO forums as being one of the most advanced tools for bypassing many different anti-spam mechanisms, including CAPTCHAs. It has been commercially available since 2006 and currently retails for $540, and we purchased a copy from the author at this price for experimentation. While we would have liked to include several other well known spamming tools (SEnuke, AutoPligg, ScrapeBox, etc), the cost of these packages range from $97 to $297, which would render this study prohibitively expensive.

Xrumer's market success in turn led to a surge of spam postings causing most service providers targeted by Xrumer to update their CAPTCHAs. This development kicked off an "arms race" period in Xrumer's evolution as the author updated solvers to overcome these obstacles. Version 5.0 of Xrumer was released in October of 2008 with significantly improved support for CAPTCHA solving. We empirically verified that 5.0 was capable of solving the default CAPTCHAs for then current versions of a number of major message boards, including: Invision Power Board (IPB) version 2.3.0, phpBB version 3.0.2, Simple Machine Forums (SMF) version 1.1.6, and vBulletin version 3.6. These systems responded in kind, and when we installed versions of these packages released shortly after Xrumer 5.0 (in particular, phpBB and vBulletin) we verified that their CAPTCHAs had been modified to defeat Xrumer's contemporaneous solver. Today, we have found that the only major message forum software whose default CAPTCHA Xrumer can solve is Simple Machines Forum (SMF).

With version 5.0.9 (released August 2009), Xrumer added integration for human-based CAPTCHA-solving services: Anti-Captcha (an alias for Antigate) and CaptchaBot. We take this as an indication that the author of Xrumer found the ongoing investment in CAPTCHA-solving software to be insufficient to support customer requirements.[3] That said, Xrumer can be configured to use a hybrid software/human based approach where Xrumer detects instances of CAPTCHAs vulnerable to its automated solvers and uses human-based solvers otherwise. In the current version of Xrumer (5.0.12), the CAPTCHA-related development seems to focus on supporting automatic navigation and CAPTCHA "extraction" (detecting the CAPTCHA and identifying the image file to send to the human-based CAPTCHA-solving service) of more Web sites, as well as evading other anti-spam techniques.

[3] The developers of Xrumer have recently been advertising enhanced CAPTCHA-solving functionality in their forthcoming "7.0 Elite" version (including support for reCaptcha), but the release date has been steadily postponed and, as of this writing (June 2010), version 5.0.12 is the latest.

When compared with developers targeting "high-value" CAPTCHAs (e.g., reCaptcha, Microsoft, Yahoo, Google, etc.), Xrumer has mostly targeted "weaker" CAPTCHAs and seems to have a policy of only including highly efficient and accurate software-based solvers. In our tests, all but one included solver required a second or less per CAPTCHA (on a netbook class computer with only a 1.6-GHz Intel Atom CPU) and had an accuracy of 100%. The one more difficult case was the solver for the phpBB version 3 forum software with the GD CAPTCHA generator and foreground noise. In this case, Xrumer had an accuracy of only 35% and required 6–7 seconds per CAPTCHA to execute.

reCaptchaOCR

At the other end of the spectrum, we obtained a specialized solver focused singularly on the popular reCaptcha service. Wilkins developed the solver as a proof of concept [23]. The existence of this OCR-based reCaptcha solver was reported in a blog posting on December 15, 2009 [6]. Although developed to defeat an earlier version of reCaptcha CAPTCHAs (Figure 2a), reCaptchaOCR was also able to defeat the CAPTCHA variant in use at the time of release (Figure 2b). Subsequently, reCaptcha changed their CAPTCHA-generation code again to the version as of this writing (Figure 2c). The tool has not been updated to solve this new variant.

We tested reCaptchaOCR on 100 randomly selected CAPTCHAs of the early 2008 variant and 100 randomly selected CAPTCHAs of the late 2009 variant. We scored the answers returned using the same algorithm that reCaptcha uses by default. reCaptcha images consist of two words, a control word for which the correct solution is known, and the other a word for which the solution is unknown (the service is used to opportunistically implement human-based OCR functionality for difficult words). By default reCaptcha will mark a solution as correct if it is within an edit distance of one of the control word. However, while we know the ground truth for both words in our tests, we do not know which was the control word. Thus, we credited the solver with half a correct solution for each *word* it solved correctly in the CAPTCHA, reasoning that there was a 50% chance of each word being the control word.

We observed an accuracy of 30% for the 2008-era test set and 18% for the 2009-era test set using the default setting of 613 iterations,[4] far lower than the average human accuracy for the same challenges (75–90% in our experiments).

Finally, we measured the overhead of reCaptchaOCR. On a laptop using a 2.13-GHz Intel Core 2 Duo each solution required an average of 105 seconds. By reducing the number of iterations to 75 we could reduce the solving time to 12 seconds per CAPTCHA, which is in line with the response time for a human solver. At this number of iterations, reCaptchaOCR still achieved similar accuracies: 29% for the 2008-era CAPTCHAs and 17% for the 2009-era CAPTCHAs.

3.2 Economics

Both of these examples illustrate the inherent challenges in fielding commercial CAPTCHA-solving software.

While the CAPTCHA problem is often portrayed in academia as a technical competition between CAPTCHA designers and computer vision experts, this perspective does not capture the business realities of the CAPTCHA-solving ecosystem. Arms races in computer security (e.g., anti-virus, anti-spam, etc.) traditionally favor the adversary, largely because the attacker's role is to generate new instances while the defender must recognize them—and the recognition problem is almost always much harder. However, CAPTCHAs reverse these roles since Web sites can be agile in their use of new CAPTCHA types, while attackers own the more challenging recognition problem. Thus, the economics of automated solving are driven by several factors: the cost to develop new solvers, the accuracy of these solvers and the responsiveness of the sites whose CAPTCHAs are attacked.

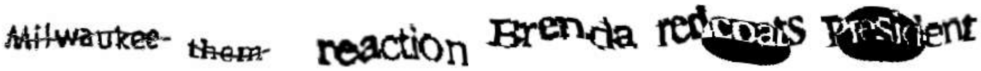
While it is difficult to precisely quantify the development cost for new solvers, it is clear that highly skilled labor is required and such developers must charge commensurate fees to recoup their time investment. Anecdotally, we contacted one such developer who was offering an automated solving library for the current reCaptcha CAPTCHA. He was charging $6,500 on a non-exclusive basis, and we did not pay to test this solver.

At the same time, as we saw with reCaptchaOCR, it can be particularly difficult to produce automated solvers that can deliver human-comparable accuracy (especially for "high-value" CAPTCHAs). While it seems that accuracy should be a minor factor since the cost of attempting a CAPTCHA is all but "free", in reality low success rates limit both the utility of a solver and its useful lifetime. In particular, over short time scales, many forums will blacklist an IP address after 5–7 failed attempts. More importantly, should a solver be put into wide use, changes in the gross CAPTCHA success rate over longer periods (e.g., days) is a strong indicator that a software solver is in use—a signature savvy sites use to revise their CAPTCHAs in turn.[5]

Thus, for a software solver to be profitable, its price must be less than the total value that can be extracted

[4] The solver performs multiple iterations and uses the majority solution to improve its accuracy.

[5] We are aware that some well-managed sites already have alternative CAPTCHAs ready for swift deployment in just such a situation.



(a) Early 2008 (b) December 16th 2009 (c) January 24th 2010

Figure 2: Examples of CAPTCHAs downloaded directly from reCaptcha at different time periods.

in the useful lifetime before the solver is detected and the CAPTCHA changed. Moreover, for this approach to be attractive, it must also cost less than the alternative: using a human CAPTCHA-solving service. To make this tradeoff concrete, consider the scenario in which a CAPTCHA-solving service provider must choose between commissioning a new software solver (e.g., for a variant of a popular CAPTCHA) or simply outsourcing recognition piecemeal to human laborers. If we suppose that it costs $10,000 to implement a solver for a new CAPTCHA type with a 30% accuracy (like reCaptchaOCR), then it would need to be used over 65 million times (20 million successful) before it was a better strategy than simply hiring labor at $0.5/1,000.[6] However, the evidence from reCaptcha's response to reCaptchaOCR suggests that CAPTCHA providers are well able to respond before such amortization is successful. Indeed, in our interview, Mr. E said that he had dabbled with automated solving but that new solvers stopped working too quickly. In his own words, "It is a big waste of time."

For these reasons, software solvers appear to have been relegated to a niche status in the solving ecosystem—focusing on those CAPTCHAs that are static or change slowly in response to pressure. While a technological breakthrough could reverse this state of affairs, for now it appears that human-based solving has come to dominate the commercial market for service.

4 Human Solver Services

Since CAPTCHAs are only intended to obstruct automated solvers, their design point can be entirely sidestepped by outsourcing the task to human labor pools, either opportunistically or on a "for hire" basis. In this section, we review the evolution of this labor market, its basic economics and some of the underlying ethical issues that informed our subsequent measurement study.

4.1 Opportunistic Solving

Opportunistic human solving relies on convincing an individual to solve a CAPTCHA as part of some other unrelated task. For example, an adversary controlling access to a popular Web site might use its visitors to opportunistically solving third-party CAPTCHAs by offering these challenges as its own [1, 8]. A modern variant of this approach has recently been employed by the Koobface botnet, which asks infected users to solve a CAPTCHA (under the guise of a Microsoft system management task) [13]. However, we believe that retention of these unwitting solvers will be difficult due to the high profile nature and annoyance of such a strategy, and we do not believe that opportunistic solving plays a major role in the market today.

4.2 Paid Solving

Our focus is instead on paid labor, which we believe now represents the core of the CAPTCHA-solving ecosystem, and the business model that has emerged around it. Figure 3 illustrates a typical workflow and the business relationships involved.

The premise underlying this approach is that there exists a pool of workers who are willing to interactively solve CAPTCHAs in exchange for less money than the solutions are worth to the client paying for their services.

The earliest description we have found for such a relationship is in a Symantec Blog post from September 2006 that documents an advertisement for a full-time CAPTCHA solver [20]. The author estimates that the resulting bids were equivalent to roughly one cent per CAPTCHA solved, or $10/1,000 (solving prices are commonly expressed in units of 1,000 CAPTCHAs solved). Starting from this date, one can find increasing numbers of such advertisements on "work-for-hire" sites such as getafreelancer.com, freelancejobsearch.com, and mistersoft.com. Shortly thereafter, *retail* CAPTCHA-solving services began to surface to resell such capabilities to a broad range of customers.

Moreover, a fairly standard business model has emerged in which such retailers aggregate the *demand* for CAPTCHA-solving services via a public Web site and open API. The example in Figure 3 shows the DeCaptcher service performing this role in steps ② and ⑥. In addition, these retailers aggregate the *supply* of CAPTCHA-solving labor by actively recruiting individuals to participate in both public and private Web-based "job sites" that provide online payments for CAPTCHAs solved. PixProfit, a worker aggregator for the DeCaptcher service, performs this role in steps ③–⑤ in the example.

[6] Moreover, human labor is highly flexible and can be used for the wide variety of CAPTCHAs demanded by customers, while a software solver inevitably is specialized to one particular CAPTCHA type.

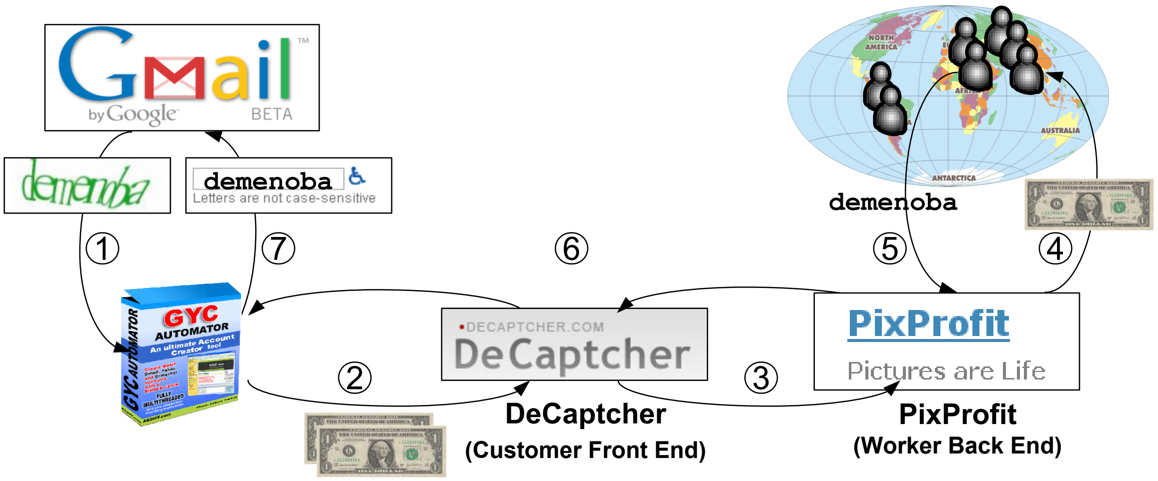


Figure 3: CAPTCHA-solving market workflow: ① GYC Automator attempts to register a Gmail account and is challenged with a Google CAPTCHA. ② GYC uses the DeCaptcher plug-in to solve the CAPTCHA at $2/1,000. ③ DeCaptcher queues the CAPTCHA for a worker on the affiliated PixProfit back end. ④ PixProfit selects a worker and pays at $1/1,000. ⑤ Worker enters a solution to PixProfit, which ⑥ returns it to the plug-in. ⑦ GYC then enters the solution for the CAPTCHA to Gmail to register the account.

4.3 Economics

While the market for CAPTCHA-solving services has expanded, the wages of workers solving CAPTCHAs have been declining. A cursory examination of historical advertisements on getafreelancer.com shows that, in 2007, CAPTCHA solving routinely commanded wages as high as $10/1,000, but by mid-2008 a typical offer had sunk to $1.5/1,000, $1/1,000 by mid-2009, and today $0.75/1,000 is common, with some workers earning as little as $0.5/1,000.

This downward price pressure reflects the commodity nature of CAPTCHA solving. Since solving is an unskilled activity, it can easily be sourced, via the Internet, from the most advantageous labor market—namely the one with the lowest labor cost. We see anecdotal evidence of precisely this pattern as advertisers switched from pursuing laborers in Eastern Europe to those in Bangladesh, China, India and Vietnam (observations further corroborated by our own experimental results later).

Moreover, competition on the retail side exerts pressure for all such employers to reduce their wages in turn. For example, here is an excerpt from a recent announcement at typethat.biz, the "worker side" of one such CAPTCHA-solving service:

```
009-12-14 13:54 Admin post
Hello, as you could see, server was unstable
last days. We can't get more captchas
because of too high prices in comparison
with other services. To solve this problem,
unfortunately we have to change the rate,
on Tuesday it will be reduced.
```

Shortly thereafter, typethat.biz reduced their offered rate from $1/1,000 to $0.75/1,000 to stay competitive.

These changes reflect similar decreases on the retail side: the customer cost to have 1,000 CAPTCHAS solved is now commonly $2/1,000 and can be as low as $1/1,000. To protect prices, a number of retailers have tried to tie their services to third-party products with varying degrees of success. For example, GYC Automator is a popular "black hat" bulk account creator for Gmail, Yahoo and Craigslist; Figure 3 shows GYC's role in the CAPTCHA ecosystem, with the tool scraping a CAPTCHA in step ① and supplying a CAPTCHA solution in step ⑦. GYC has a relationship with the CAPTCHA-solving service Image2Type (not to be confused with ImageToType). Similarly, SENuke is a blog and forum spamming product that has integral support for two "up-market" providers, BypassCaptcha and BeatCaptchas. In both cases, this relationship allows the CAPTCHA-solving services to charge higher rates: roughly $7/1,000 for BypassCaptcha and BeatCaptchas, and over $20/1,000 for Image2Type. It also provides an ongoing revenue source for the software developer. For his service, Mr. E confirms that software partners bring in many customers (indeed, they are the majority revenue source) and that he offers a variety of revenue sharing options to attract such partners.

However, such large price differences encourage arbitrage, and in some cases third-party developers have created plug-ins to allow the use of cheaper services on such packages. Indeed, in the case of GYC Automator, an independent developer built a DeCaptcher plug-in which reduced the solving cost by over an order of magnitude. This development has created an ongoing conflict between the seller of GYC Automator and the distributor of the DeCaptcher plug-in. Other software developers have chosen to forgo large margin revenue sharing in favor of service diversity. For example, modern versions of the Xrumer package can use multiple price-leading services (Antigate and CaptchaBot).

Finally, while it is challenging to measure profitability directly, we have one anecdotal data point. In our discussions with MR. E, whose service is in the middle of the price spectrum, he indicated that routinely 50% of his revenue is profit, roughly 10% is for servers and bandwidth, and the remainder is split between solving labor and incentives for partners.

4.4 Active Measurement Issues

The remainder of our paper focuses on active measurement of such services, both by paying for solutions and by participating in the role of a CAPTCHA-solving laborer. The security community has become increasingly aware of the need to consider the legal and ethical context of its actions, particularly for such active involvement, and we briefly consider each in turn for this project.

In the United States (we restrict our brief discussion to U.S. law since that is where we operate), there are several bodies of law that may impinge on CAPTCHA solving. First, even though the services being protected are themselves "free", it can be argued that CAPTCHAS are an access control mechanism and thus evading them exceeds the authorization granted by the site owner, in potential violation of the Computer Fraud and Abuse Act (and certainly of their terms of service). While this interpretation is debatable, it is a moot point for our study since we never make use of solved CAPTCHAS and thus never access any of the sites in question. A trickier issue is raised by the Digital Millennium Copyright Act's anti-circumvention clause. While there are arguments that CAPTCHA solvers provide a real use outside circumvention of copyright controls (e.g., as aids for the visually impaired) it is not clear—especially in light of increasingly common audio CAPTCHA options—that such a defense is sufficient to protect infringers. Indeed, Ticketmaster recently won a default judgment against RMG Technologies (who sold automated software to bypass the Ticketmaster CAPTCHA) using just such an argument [2]. That said, while one could certainly apply the DMCA against those offering a *service* for CAPTCHA-solving purposes, it seems a stretch to include individual human workers as violators since any such "circumvention" would include innate human visual processes.

Aside from potential legal restrictions, there are also related ethical concerns; one can do harm without such actions being illegal. In considering these questions, we use a consequentialist approach – comparing the consequences of our intervention to an alternate world in which we took no action — and evaluate the outcome for its cost-benefit tradeoff.

On the purchasing side, we impart no direct impact since we do not actually *use* the solutions on their respective sites. We *do* have an indirect impact however since, through purchasing services, we are providing support to both workers and service providers. In weighing this risk, we concluded that the indirect harm of our relatively small investment was outweighed by the benefits that come from better understanding the nature of the threat. On the solving side, the ethical questions are murkier since we understand that solutions to such CAPTCHAS *will* be used to circumvent the sites they are associated with. To sidestep this concern, we chose *not* to solve these CAPTCHAS ourselves. Instead, for each CAPTCHA one of our worker agents was asked to solve, we proxied the image back into the same service via the associated retail interface. Since each CAPTCHA is then solved by the *same* set of solvers who would have solved it *anyway*, we argue that our activities do not impact the gross outcome. This approach does cause slightly more money to be injected into the system, but this amount is small.

Finally, we consulted with our human subjects liaison on this work and we were told that the study did not require approval.

5 Solver Service Quality

In this section we present our analysis of CAPTCHA-solving services based on actively engaging with a range of services as a client. We evaluate the customer interface, solution accuracy, response time, availability, and capacity of the eight retail CAPTCHA-solving services listed in Table 1.

We chose these services through a combination of Web searching and reading Web forums focused on "black-hat" search-engine optimization (SEO). In October of 2009, we selected the eight listed in Table 1 because they were well-advertised and reflected a spectrum of price offerings at the time. Over the course of our study, two of the services (CaptchaGateway and CaptchaBypass) ceased operation—we suspect because of competition from lower-priced vendors.

5.1 Customer Account Creation

For most of these services, account registration is accomplished via a combination of the Web and e-mail: contact information is provided via a Web site and subsequent sign-up interactions are conducted largely via e-mail. However, most services presented some obstacles

Service	$/1K Bulk	Dates (2009–2010)	Requests	Responses
Antigate (AG)	$1.00	Oct 06 – Feb 01 (118 days)	28,210	27,726 (98.28%)
BeatCaptchas (BC)	$6.00	Sep 21 – Feb 01 (133 days)	28,303	25,708 (90.83%)
BypassCaptcha (BY)	$6.50	Sep 23 – Feb 01 (131 days)	28,117	27,729 (98.62%)
CaptchaBot (CB)	$1.00	Oct 06 – Feb 01 (118 days)	28,187	22,677 (80.45%)
CaptchaBypass (CP)	$5.00	Sep 23 – Dec 23 (91 days)	17,739	15,869 (89.46%)
CaptchaGateway (CG)	$6.60	Oct 21 – Nov 03 (13 days)	1,803	1,715 (95.12%)
DeCaptcher (DC)	$2.00	Sep 21 – Feb 01 (133 days)	28,284	24,411 (86.31%)
ImageToText (IT)	$20.00	Oct 06 – Feb 01 (118 days)	14,321	13,246 (92.49%)

Table 1: Summary of the customer workload to the CAPTCHA-solving services.

to account creation, reflecting varying degrees of due diligence.

For example, both CaptchaBot and Antigate required third-party "invitation codes" to join their services, which we acquired from the previously mentioned forums. Interestingly, Antigate guards against Western users by requiring site visitors to enter the name of the Russian prime minister in Cyrillic before granting access—an innovation we refer to as a "culturally-restricted CAPTCHA".[7] Some services require a live phone call for account creation, for which we used an anonymous mobile phone to avoid any potential biases arising from using a University phone number. In our experience, however, the burden of proof demanded is quite low and our precautions were likely unnecessary. For example, setting up an ImageToText account required a validation call, but the only question asked was "Did you open an account on ImageToText?" Upon answering in the affirmative (in a voice clearly conflicting with the gender of the account holder's name), our account was promptly enabled. For one service, DeCaptcher, we created multiple accounts to evaluate whether per-customer rate limiting is in use (we found it was not).

Finally, each service typically requires prepayment by customers, in units defined by their price schedule (1,000 CAPTCHAs is the smallest "package" generally offered). To fund each account, we used prepaid VISA gift cards issued by a national bank unaffiliated with our university.

5.2 Customer Interface

Most services provide an API package for uploading CAPTCHAs and receiving results, often in multiple programming languages; we generally used the PHP-based APIs. BeatCaptchas and BypassCaptcha did not offer pre-built API packages, so we implemented our own API in Ruby to interface with their Web sites. The client APIs generally employ one of two methods when interacting with their corresponding services. In the first, the API client performs a single HTTP POST that uploads the image to the service, waits for the CAPTCHA to be solved, and receives the answer in the HTTP response; BeatCaptchas, BypassCaptcha, CaptchaBypass and CaptchaBot utilize this method.

In the second, the client performs one HTTP POST to upload the image, receives an image ID in the response, and subsequently polls the site for the CAPTCHA solution using the image ID; Antigate, CaptchaGateway, and ImageToText employ this approach. These APIs recommend poll rates between 1–5 seconds; we polled these services once per second. DeCaptcher uses a custom protocol that is not based on HTTP, although they also offer an HTTP interface. One interesting note about ImageToText is that customers must verify that their API code works in a test environment before gaining access to the actual service. The test environment allows users to see the CAPTCHAs they submit and solve them manually.

5.3 Service Pricing

Several of the services, notably Antigate and DeCaptcher, offer bidding systems whereby a customer can offer payment over the market rate in exchange for higher priority access to solvers when load is high. In our experience, DeCaptcher charges customers their full bid price, while Antigate typically charges at a lower rate depending on load (as might happen in a second-price auction). To effectively use Antigate, we set our bid price to $2/1,000 solutions since we experienced a large volume of load shedding error codes at the minimum bid price of $1/1,000 (Section 5.9 reports on our experiences with service load in more detail). We have not seen price fluctuations on the worker side of these services, and thus we believe that this overage represents pure profit to the service provider.

[7] In principle, such an approach could be used to artificially restrict labor markets to specific cultures (i.e., CAPTCHA labor protectionism). However it is an open problem if such a *general* form of culturally-restricted CAPTCHA can be devised that has both a large number of examples and a low false reject rate from its target population.

5.4 Test Corpus

We evaluated the eight CAPTCHA-solving services in Table 1 as a customer over the course of about five months using a representative sample of CAPTCHAs employed by popular Web sites. To collect this CAPTCHA workload, we assembled a list of 25 popular Web sites with unique CAPTCHAs based on the Alexa rank of the site and our informal assessment of its value as a target (see Figure 5 for the complete list). We also used CAPTCHAs from reCaptcha, a popular CAPTCHA provider used by many sites. We then collected about 7,500 instances of each CAPTCHA directly from each site. For the capacity measurement experiments (Section 5.8), we used 12,000 instances of the Yahoo CAPTCHA graciously provided to us by Yahoo.

5.5 Verifying Solutions

To assess the accuracy of each service, we needed to determine the correct solution for each CAPTCHA in our corpus. We used the services themselves to do this for us. For each instance, we used the most frequent solution returned by the solver services, after normalizing capitalization and whitespace. If there was more than one most frequent solution, we treated all answers as incorrect (taking this to mean that the CAPTCHA had no correct solution). Table 1 shows the overall accuracy of each service as given by our method.

To validate this heuristic, we randomly selected 1,025 CAPTCHAs having at least one service-provided solution and manually examined the images. Of these, we were able to solve 1,009, of which 940 had a unique plurality that agreed with our solution, giving an error rate for the heuristic of just over 8%. Of the 16 CAPTCHAs (1.6%) we could not solve, seven were entirely unreadable, six had ambiguous characters (e.g., ‘0’ vs. ‘o’, ‘6’ vs. ‘b’), and three were rendered ambiguous due to overlapping characters. (We note that Bursztein *et al.* [3] removed CAPTCHAs with no majority from their calculation, which resulted in a higher estimated accuracy than we found in our study.)

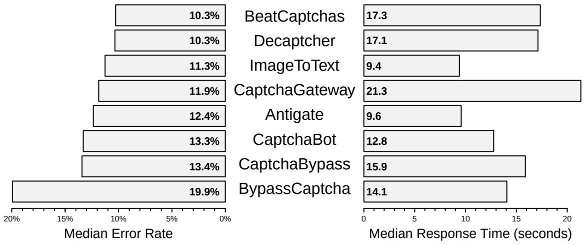


Figure 4: Median error rate and response time (in seconds) for all services. Services are ranked top-to-bottom in order of increasing error rate.

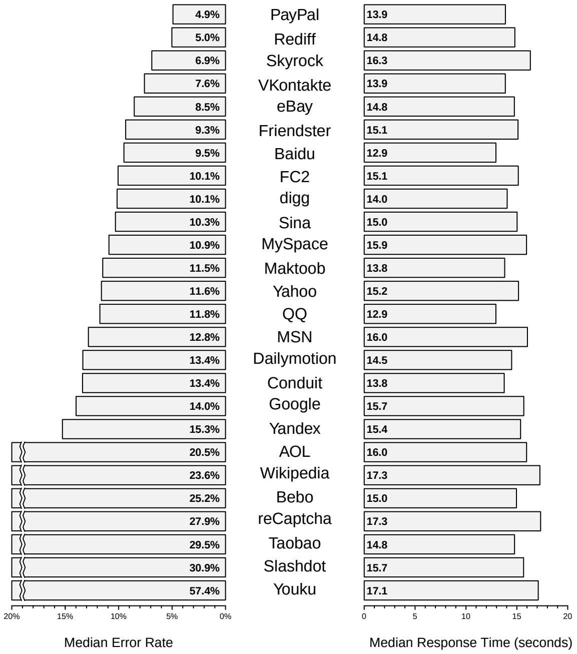


Figure 6: Median error rate and response time (in seconds) for all CAPTCHAs. CAPTCHAs are ranked top-to-bottom in order of increasing error rate.

5.6 Quality of Service

To assess the accuracy, response time, and service availability of the eight CAPTCHA solving services, we continuously submitted CAPTCHAs from our corpus to each service over the course of the study. We submitted a single CAPTCHA every five minutes to all services simultaneously, recording the time when we submitted the CAPTCHA and the time when we received the response. Recall that ImageToText, Antigate and CaptchaGateway require customers to poll the service for the response to a submitted CAPTCHA; we paused one second between each poll call.

Table 1 also summarizes the dates, durations, and number of CAPTCHA requests we submitted to the services; Figure 5 presents the error rate and mean response time at a glance for each combination of solver service and CAPTCHA type. We used each service for up to 118 days, submitting up to 28,303 requests per service during that period. We were not able to submit the same number of CAPTCHAs to all services for a number of reasons. For example, services would go offline temporarily, or we would rewrite parts of our client implementation, thus requiring us to temporarily remove the service from the experiment. Furthermore, CaptchaGateway and CaptchaBypass ceased operation during our study.

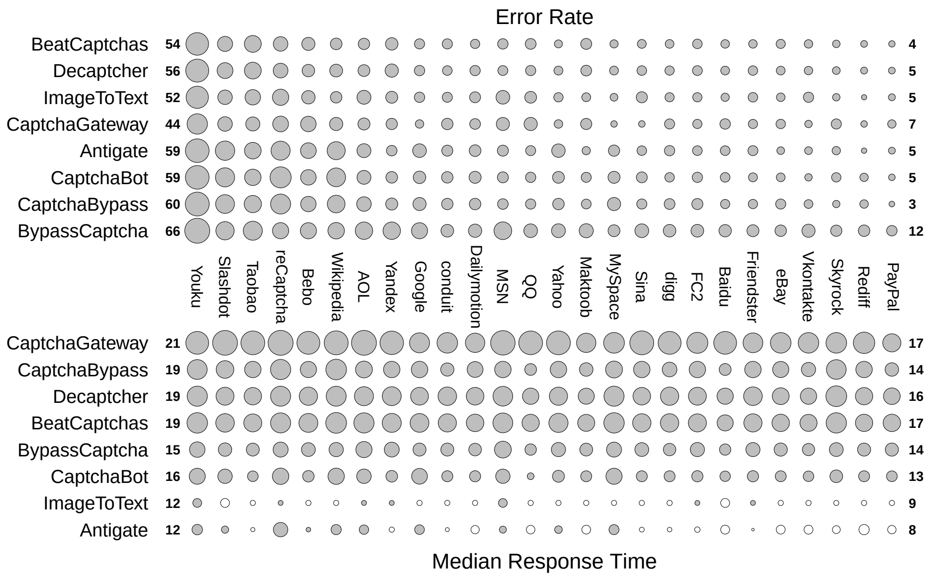


Figure 5: Error rate and median response time for each combination of service and CAPTCHA type. The area of each circle upper table is proportional to the error rate (among solved CAPTCHAs). In the lower table, circle area is proportional to the response time *minus ten seconds* (for increased contrast); negative values are denoted by unshaded circles. Numeric values corresponding to the values in the leftmost and rightmost columns are shown on the side. Thus, the error rate of BypassCaptcha on Youku CAPTCHAs is 66%, and for BeatCaptchas on PayPal 4%. The median response time of CaptchaGateway on Youku is 21 seconds, and 8 seconds for Antigate on PayPal.

Accuracy

A CAPTCHA solution is only useful if it is correct. The left bar plot in Figure 4 shows the median error rate for each service. Overall the services are reasonably accurate: with the exception of BypassCaptcha, 86–89% of responses [8] were correct. This level of accuracy is in line with results reported by Bursztein *et al.* [3] for human solvers and substantially better than the accuracy of reCaptchaOCR (Section 3).

By design, CAPTCHAs vary in difficulty. Do the observed error rates reflect such differences? The top half of Figure 5 shows service accuracy (in terms of its error rate) on each CAPTCHA type. The area of each circle is proportional to a service's mean error rate on a particular CAPTCHA type. Services are arranged along the y-axis in order of increasing accuracy, with the most accurate (lowest error rate) at the top and the least accurate (highest error rate) at the bottom. CAPTCHA types are arranged in decreasing order of their median error rate. The median error rate of each type is also shown in Figure 6.

Accuracy clearly depends on the type of CAPTCHA. The error rate for ImageToText with Youku, for instance, is 5 times its PayPal error rate. Furthermore, the ranking of CAPTCHA accuracies are generally consistent across the services—all services have relatively poor accuracy on Youku and good accuracy on PayPal.

Based on the data, one might conclude that a group of CAPTCHAs on the left headed by Youku, reCaptcha, Slashdot, and Taobao are "harder" than the rest. However an important factor affecting solution accuracy (as well as response time) in our measurements is worker familiarity with a CAPTCHA type. In the case of Youku, for instance, workers may simply be unfamiliar with these CAPTCHAs. On the other hand, workers are likely familiar with reCaptcha CAPTCHAs (see Section 6.6), which may genuinely be "harder" than the rest. As a point of comparison, MR. E reported in our interview that his service experiences a 5–10% error rate. Since his CAPTCHA mix is likely different, and less diverse, than our full set, his claim seems reasonable.

Response Time

In addition to accuracy, customers want services that solve CAPTCHAs quickly. Figure 7 shows the cumulative distribution of response times of each service. The curves of CaptchaBot, CaptchaBypass, ImageToText, and Antigate exhibit the quantization effect of polling—either in the client API or on the server—as a stair-step pattern. The shape of the distributions is characteristically lognormal, with a median response of 14 seconds (across all services) and a third-quartile response time of 20 seconds—well within the session timeout of most Web

[8] The error rate is over received responses and does not include rejected requests. We consider response rate to be a measure of *availability* rather than accuracy.

sites. For convenience, Figure 4 also shows median response times for each service. In contrast to Bursztein *et al.* [3], who used a different labor pool (Amazon Mechanical Turk), we found no significant difference in response times of correct and incorrect responses.

Services differ considerably in the relative response times they provide to their customers. Antigate (for which we paid a slight premium for priority service as described in Section 5.3) and ImageToText provided the fastest service with median response times of 9.6 seconds and 9.4 seconds, respectively, with 90% of CAPTCHAs solved under 25 seconds. CaptchaGateway was the slowest service we measured, with a median of 21.3 seconds and 10% of responses taking over a minute; it was also one of the two services that ceased operation during our study. The remaining services fall in between those extremes. MR. E reported that his service trains workers to achieve response times of 10–12 seconds on average, which is consistent with our measurements of his service.

DeCaptcher and BeatCaptchas have very similar distributions. We have seen evidence (i.e., error messages from BeatCaptchas that are identical to ones documented for the DeCaptcher API) that suggests that BeatCaptchas uses DeCaptcher as a back end. Antigate returns some correct responses unusually quickly (a few seconds), for which we currently do not have an explanation; we have ruled out caching effects.

Services have an advantage if they have better response times than their competition, and the services we measured differ substantially. We suspect that it is a combination of two factors: software and queueing delay in the service infrastructure, and worker efficiency. Antigate, for instance, appears to have an unusually large labor pool (Section 5.8), which may enable them to keep queueing delay low. Similarly, ImageToText appears to have an adaptive, high-quality labor pool (Section 6.4). We observed additional delays of 5 seconds due to load (Section 5.9), but load likely affects all services similarly.

We found that accuracy varied with the type of CAPTCHA. A closely related issue is to what degree response time also varies according to CAPTCHA type. The bottom of Figure 5 shows response times by CAPTCHA type. Services are listed along the y-axis from slowest (top) to fastest service (bottom). The area of each circle is proportional to the median response time of a service on a particular CAPTCHA type *minus ten seconds* (for greater contrast). Shaded circles are times in excess of ten seconds, unshaded circles are times less than ten seconds. For example, the median response time of Antigate on PayPal CAPTCHAs—8 seconds—is shown as an unshaded circle. Note that CAPTCHA types are still sorted by *accuracy*. The right half of Figure 4 aggregates response times by service, showing the median response time of each.

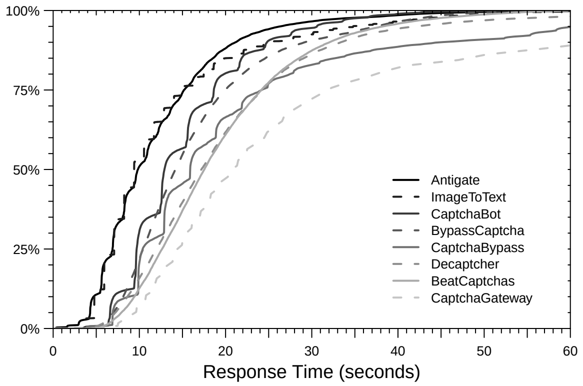


Figure 7: Cumulative distribution of response times for each service.

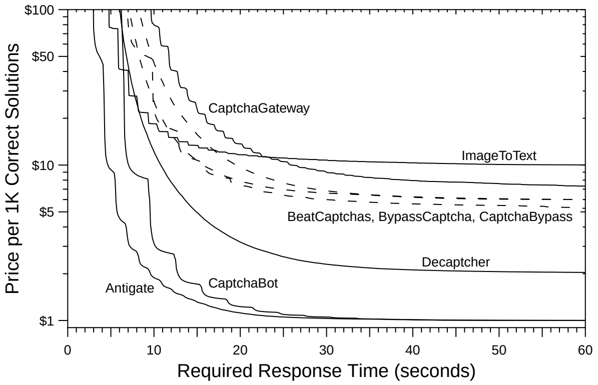


Figure 8: Price for 1,000 correctly-solved CAPTCHAs within a given response time threshold.

We see some variation in response time among CAPTCHA types. Youku and reCaptcha, for instance, consistently induce longer response times across services, whereas Baidu, eBay, and QQ consistently have shorter response times. However, the variation in response times among the services dominates the variation due to CAPTCHA type. The fastest CAPTCHAs that DeCaptcher solves (e.g., Baidu and QQ) are slower on average than the slowest CAPTCHAs that Antigate and ImageToText solve.

5.7 Value

CAPTCHA solvers differ in terms of accuracy, response time, and price. The *value* of a particular solver to a customer depends upon the combination of all of these factors: a customer wants to pay the lowest price for both fast and accurate CAPTCHAs. For example, suppose that a customer wants to create 1,000 accounts on an Internet service, and the Internet service requires that CAPTCHAs be solved within 30 seconds. When using a CAPTCHA solver, the customer will have to pay to have at least 1,000 CAPTCHAs solved, and likely more due to solutions with response times longer than the 30-second

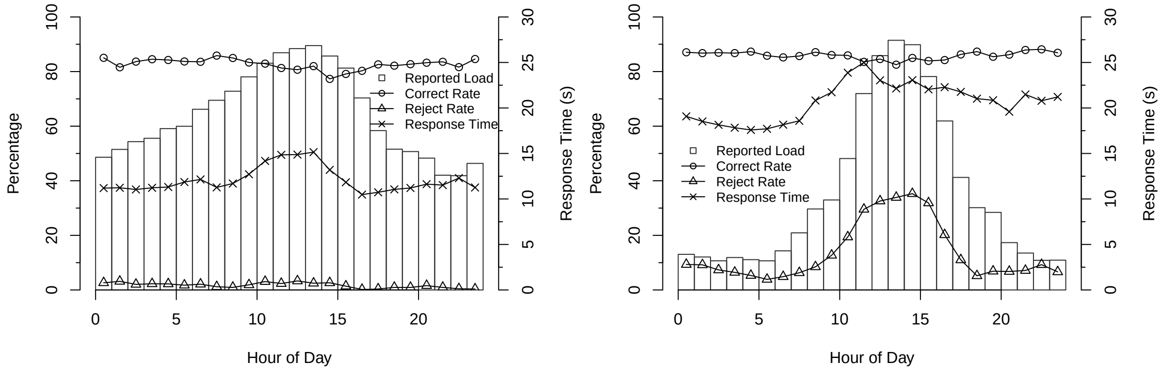


Figure 9: Load reported by (a) Antigate and (b) DeCaptcher as a function of time-of-day in one-hour increments. For comparison, we show the percentage of correct responses and rejected requests per hour, as well as the average response time per hour.

threshold (recall that customers do not have to pay for incorrect solutions). From this perspective, the solver with the best value may not be the one with the cheapest price.

Figure 8 explores the relationship among accuracy, response time, and price for this scenario. The x-axis is the time threshold T within which a CAPTCHA is useful to a customer. The y-axis is the *adjusted price* per bundle of 1,000 CAPTCHAs that are both solved correctly *and* solved within time T. Each curve corresponds to a solver. Each solver charges a price per CAPTCHA solved (Table 1), but not all solved CAPTCHAs will be useful to the customer. The adjusted price therefore includes the overhead of solving CAPTCHAs that take longer than T and are effectively useless. Consider an example where a customer wants to have 1,000 correct CAPTCHAs solved within 30 seconds, a solver charges \$2/1,000 CAPTCHAs, and 70% of the solver's CAPTCHA responses are correct and returned within 30 seconds. In this case, the customer will effectively pay an adjusted price of $\$2 \times (1/0.70) = \$2.86/1,000$ useful CAPTCHAs.

The results in Figure 8 show that the solver with the best value depends on the response time threshold. For high thresholds (more than 25 seconds), both Antigate and CaptchaBot provide the best value and ImageToText is the most expensive as suggested by their bulk prices (Table 1). However, below this threshold the rankings begin to change. Antigate begins to have better value than CaptchaBot due to having consistently better response times. In addition, ImageToText starts to overtake the other services. Even though its bulk price is $5x$ that of DeCaptcher, for instance, its service is a better value for having CAPTCHAs solved within 8 seconds (albeit at a premium adjusted price).

5.8 Capacity

Another point of differentiation is solver capacity, namely how many CAPTCHAs a service can solve in a given unit of time. In addition to low-rate measurements, we also attempted to measure a service's maximum capacity using bursts of CAPTCHA requests. Specifically, we measured the number and rate of solutions returned in response to a given offered load, substantially increasing the load in increments until the service appeared overloaded. We carried out this experiment successfully for five of the services. Of them, Antigate had by far the highest capacity, solving on the order of 27 to 41 CAPTCHAs per second. Even at our highest sustained offered load (1,536 threads submitting CAPTCHAs simultaneously, bid set at \$3/1,000), our rejection rate was very low, suggesting that Antigate's actual capacity may in fact be higher. Due to financial considerations, we did not attempt higher offered loads.

For the remaining services, we exceeded their available capacity. We took a non-negligible reject rate to be an indicator of the service running at full capacity. Both DeCaptcher and CaptchaBot were able to sustain a rate of about 14–15 CAPTCHAs per second, with BeatCaptchas and BypassCaptchas sustaining a solve rate of eight and four CAPTCHAs per second, respectively.

Based on these rates, we can calculate a rough estimate of the number of workers at these services. Assuming 10–13 seconds per CAPTCHA (based on our interview with MR. E, and consistent with our measured latencies of his service in the 10–20 second range), Antigate would have had at least 400–500 workers available to service our request. Since we did not exceed their available capacity, the actual number may be larger. Both DeCaptcher and CaptchaBot, at a solve rate of 15 CAPTCHAs per second mentioned above, would have had 130–200 workers available.

5.9 Load and Availability

Customers can poll the transient load on the services and offer payment over the market rate in exchange for higher priority access when load is high. During our background CAPTCHA data collection for these services, we also recorded the transient load that they reported. From these measurements, we can examine to what extent services report substantial load, and correlate reported load with other observable metrics (response time, reject rate) to evaluate the validity of the load reports. Because DeCaptcher charges the full customer bid independent of actual load, for instance, it might be motivated to report a false high load in an attempt to encourage higher bids from customers.

Figure 9 shows the average reported load as a function of the time of day (in the US Pacific time zone) for both services: for each hour, we compute the average of all load samples taken during that hour for all days of our data set. Antigate reports a higher nominal background load than DeCaptcher, but both services clearly report a pronounced diurnal load effect.

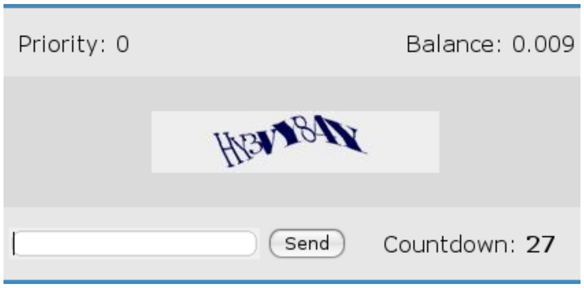
For comparison, we also overlay three other service metrics for each hour across all days: average response time of solved CAPTCHAs, percentage of submitted CAPTCHAs rejected by the service, and the percentage of responses with correct solutions. Response time correlates with reported load, increasing by 5 seconds during high load for each service—suggesting that the high load reports are indeed valid. The percentage of rejected requests for DeCaptcher further validates the load reports. When our bids to DeCaptcher were at the base price of $2/1,000 at times of high load, DeCaptcher aggressively rejected our work requests. To confirm that a higher bid resulted in lower rejection rates, we measured available capacity at 5PM (US Pacific time) at the base price of $2 and then, a few minutes later, at $5, obtaining solve rates of 8 and 18 CAPTCHAs per second, respectively. Although not conclusive, this experience suggests that higher bids may be necessary to achieve a desired level of service at times of high load. Likewise, Antigate exhibits better quality of service when bidding $1 over the base price, though bidding over this amount produced no noticeable improvement (we tested up to $6/1,000).

As further evidence, recall that for Antigate we had to offer premium bids before the service would solve our requests (Section 5.2). As a result, even during high loads Antigate did not reject our requests, presumably prioritizing our requests over others with lower bids.

Finally, as expected, accuracy is independent of load: workers are shielded from load behind work queues, solving CAPTCHAs to their ability unaffected by the offered load on the system.

6 Workforce

Human CAPTCHA solving services are effectively aggregators. On one hand, they aggregate demand by providing a singular point for purchasing solving services. At the same time, they aggregate the labor supply by providing a singular point through which workers can depend on being offered consistent CAPTCHA solving work for hire. Thus, for each of the publicly-facing retail sites described previously, there is typically also a private "job site" accessed by workers to receive CAPTCHA images and provide textual solutions. Identifying these job sites and which retail service they support is an investigative challenge. For this study, we focused our efforts on two services for which we feel confident about the mapping: Kolotibablo and PixProfit. Kolotibablo is a Russian-run job site that supplies solutions for the retail service Antigate (which, along with CaptchaBot, is the current price leader).



Priority: 0

Balance: 0.009

Countdown: 27

Figure 10: Portion of a PixProfit worker interface displaying a Microsoft CAPTCHA.

6.1 Account Creation

For each job site, account creation is similar to the retail side, but due diligence remains minimal. As a form of quality control, some job sites will evaluate new workers using a corpus of "test" CAPTCHAs (whose solutions are known *a priori*) before they allow them to solve externally provided CAPTCHAs. For this reason, we discard the first 30 CAPTCHAs provided by PixProfit, which we learned by experience correspond to test CAPTCHAs.

6.2 Worker Interface

Services provide workers with a Web based interface that, after logging in, displays CAPTCHAs to be solved and provides a text box for entering the solution (Figure 10 shows an example of the interface for PixProfit). Each site also tracks the number of CAPTCHAs solved, the number that were reported as correct (by customers of the retail service), and the amount of money earned. PixProfit also assigns each worker a "priority" based on solution accuracy. Better accuracy results in more CAPTCHAs to solve during times of lower load. If a solver's accuracy decreases too much, services ban the account. In our experiments, our worker agents always used fresh accounts with the highest level of priority.

Language	Example	AG	BC	BY	CB	DC	IT	All
English	one two three	51.1	37.6	4.76	40.6	39.0	62.0	39.2
Chinese (Simp.)	一 二 三	48.4	31.0	0.00	68.9	26.9	35.8	35.2
Chinese (Trad.)	一 二 三	52.9	24.4	0.00	63.8	30.2	33.0	34.1
Spanish	uno dos tres	1.81	13.8	0.00	2.90	7.78	56.8	13.9
Italian	uno due tre	3.65	8.45	0.00	4.65	5.44	57.1	13.2
Tagalog	isá dalawá tatló	0.00	5.79	0.00	0.00	7.84	57.2	11.8
Portuguese	um dois três	3.15	10.1	0.00	1.48	3.98	48.9	11.3
Russian	один два три	24.1	0.00	0.00	11.4	0.55	16.5	8.76
Tamil	ஒன்று இரண்டு மூன்று	2.26	21.1	3.26	0.74	12.1	5.36	7.47
Dutch	een twee drie	4.09	1.36	0.00	0.00	1.22	31.1	6.30
Hindi	एक दो तीन	10.5	5.38	2.47	1.52	6.30	9.49	5.94
German	eins zwei drei	3.62	0.72	0.00	1.46	0.58	29.1	5.91
Malay	satu dua tiga	0.00	1.42	0.00	0.00	0.55	29.4	5.23
Vietnamese	một hai ba	0.46	2.07	0.00	0.00	1.74	18.1	3.72
Korean	일 이 삼	0.00	0.00	0.00	0.00	0.00	20.2	3.37
Greek	ένα δύο τρία	0.45	0.00	0.00	0.00	0.00	15.5	2.65
Arabic	واحد اثنين ثلاثة	0.00	0.00	0.00	0.00	0.00	15.3	2.56
Bengali	এক দুই তিন	0.45	0.00	9.89	0.00	0.00	0.00	1.72
Kannada	ಒಂದು ಎರಡು ಮೂರು	0.91	0.00	0.00	0.00	0.55	6.14	1.26
Klingon		0.00	0.00	0.00	0.00	0.00	1.12	0.19
Farsi	یک دو سه	0.45	0.00	0.00	0.00	0.00	0.00	0.08

Table 2: Percentage of responses from the services with correct answers for the language CAPTCHAs.

6.3 Worker Wages

Kolotibablo pays workers at a variable rate depending on how many CAPTCHAs they have solved. This rate varies from $0.50/1,000 up to over $0.75/1,000 CAPTCHAs. PixProfit is the equivalent supplier for DeCaptcher and offers a somewhat higher rate of $1/1,000. Typically, workers must earn a minimum amount of money before payout ($3.00 at PixProfit and $1.00 at Kolotibablo), and services commonly provide payment via an online e-currency system such as WebMoney.

While we cannot directly measure the gross wages paid by either service, Kolotibablo provides a public list to its workers detailing the monthly earnings for the top 100 solvers each day (presumably as a worker incentive). We monitored these earnings for two months beginning on Dec. 1st, 2009. On this date, the average monthly payout among the top 100 workers was $106.31. However, during December, Kolotibablo revised its bonus payout system, which reduced the payout range by approximately 50% (again reflecting downward price pressure on CAPTCHA-solving labor). As a result, one month later on Jan. 1st, 2010, the average monthly payout to the top 100 earners decreased to $47.32. In general, these earnings are roughly consistent with wages paid to low-income textile workers in Asia [12], suggesting that CAPTCHA-solving is being outsourced to similar labor pools; we investigate this question next.

6.4 Geolocating Workers

We crafted CAPTCHAs whose solutions would reveal information about the geographic demographics of the CAPTCHA solvers. We created CAPTCHAs using words corresponding to digits in the native script of various languages ("uno", "dos", "tres", etc., for the CAPTCHA challenge in Spanish), where the correct solution is the sequence of Roman numerals corresponding to those words ("1", "2", "3", etc.) for any CAPTCHA in any language. Ideally, such CAPTCHAs should be easy to grasp and fast to solve by the language's speakers, yet substantially less likely to be solved by non-speakers or random chance. We expect a measurably high accuracy for services employing workers familiar with those languages.

Table 2 lists the languages we used in this experiment along with an example three-digit CAPTCHA in the language corresponding to the solution "123". For broad global coverage, we selected 21 languages based on a combination of factors including global exposure (English), prevalence of world-wide native speakers (Chinese, Spanish, English, Hindi, Arabic), regions of expected low-cost labor markets with inexpensive Internet access (India, China, Southeast Asia, Latin America), and developed regions unlikely to be sources of affordable CAPTCHA labor (e.g., Western Europe) and lastly one synthetic language as a control (Klingon [15]).

The CAPTCHA we submitted had instructions in the language for how to solve the CAPTCHA (e.g., "Por favor escriba los números abajo" for Spanish), as well as an initial word and Roman numeral as a concrete example ("uno", "1"). In our experiments, we randomly generated 222 unique CAPTCHAs in each language and submitted them to the six services still operating in January 2010. We rotated through languages such that we submitted a CAPTCHA in this format once every 20–25 minutes. The CAPTCHAs did not repeat digits to reduce the correlated effect of a random guess. As a result, the actual probability for guessing a CAPTCHA is 1/504 (9 × 8 × 7, reduced by 1 due to the example), although workers unaware of the construction would still be making guesses out of 1,000 possibilities.

Table 2 also shows the accuracy of the services when presented with these CAPTCHAs. The accuracy corresponds to a response with all three digits correct (since we created them we have their ground truth). For a convenient ordering, we sort the languages by the average accuracy across all services.

The results paint a revealing picture. First, although Roman alphanumerics in typical CAPTCHAs are globally comprehensible—and therefore easily outsourced—English words for numerals represent a noticeable semantic gap for presumably non-English speakers. Very high accuracies on normal CAPTCHAs drop to 38–62% for the challenge presented in English.

Second, workers at a number of the services exhibit strong affinities to particular languages. Five of the services have accuracies for Chinese (Traditional and Simplified) either substantially higher or nearly as high as English. The services evidently include a sizeable workforce fluent in Chinese, likely mainland China with available low-cost labor. In addition, Antigate has appreciable accuracies for Russian and Hindi, presumably drawing on workforces in Russia and India. Similarly for CaptchaBypass and Russian; BeatCaptcha and Tamil, Portuguese, and Spanish; and DeCaptcher and Tamil. Other non-trivial accuracies in Bengali and Tagalog suggest further recruitment in India and southeast Asia. Services with non-trivial accuracies in Portuguese, Spanish, and Italian could be explained by a workforce familiar with one language who can readily deduce similar words in the other Romance languages. Consistent with these observations, MR. E reported in our interview that they draw from labor markets in China, India, Bangladesh, and Vietnam.

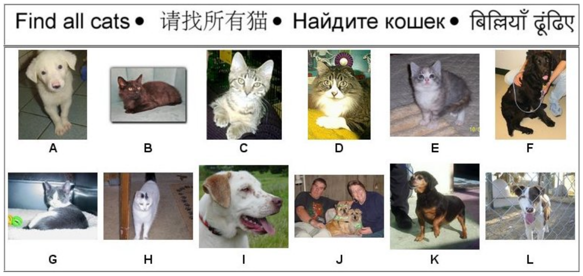

Find all cats • 请找所有猫 • Найдите кошек • बिल्लियाँ ढूंढिए
A
B
C
D
E
F
G
H
I
J
K
L

Figure 11: Custom Asirra CAPTCHA: workers must type the letters corresponding to pictures of cats.

Finally, the results for ImageToText are impressive. Relative to the other services, ImageToText has appreciable accuracy across a remarkable range of languages, including languages where none of the other services had few if any correct solutions (Dutch, Korean, Vietnamese, Greek, Arabic) and even two correct solutions of CAPTCHAs in Klingon. Either ImageToText recruits a truly international workforce, or the workers were able to identify the CAPTCHA construction and learn the correct answers. ImageToText is the most expensive service by a wide margin, but clearly has a dynamic and adaptive labor pool.

Time Zone. As another approach for using CAPTCHAs to reveal demographic information about workers—in this case, their time zone—we translated the following instruction into 14 of the languages as CAPTCHA images: "Enter the current time". We sent these CAPTCHAs to each of the six services at the same rate as the other language CAPTCHAs with numbers. We received 15,775 responses, with the most common response being a retype of the instruction in the native language. Of the remaining responses, we received 1,583 (10.0%) with an answer in a recognizable time format. Of those, 77.9% of them came from UTC+8, further reinforcing the estimation of a large labor pool from China; the two other top time zones were the Indian UTC+5.5 with 5.7% and Eastern Europe UTC+2 with 3.0%.

6.5 Adaptability

As a final assessment, we wanted to examine how both CAPTCHA services and solvers adapt to changes in state-of-the-art CAPTCHA generation. We focused on the recently proposed Asirra CAPTCHA [9], which is based on identifying pictures of cats and dogs among a set of 12 images. Using the corpus of images provided by the Asirra authors, we hand crafted our own version of the

Kolotibablo (Antigate)				PixProfit (DeCaptcher)			
Service	**# CAPTCHAs**	**% Total**	**% Cum.**	**Service**	**# CAPTCHAs**	**% Total**	**% Cum.**
Microsoft	6,552	25.5%	25.5%	Microsoft	12,135	43.1%	43.1%
Vkontakte.ru	5,908	23.0%	48.5%	reCaptcha	10,788	38.3%	81.4%
Mail.ru	3,607	14.0%	62.5%	Google	1,202	4.3%	85.7%
Captcha.ru	2,476	9.6%	72.2%	Yahoo	1,307	3.7%	89.3%
reCaptcha	921	3.6%	75.8%	AOL	415	1.5%	90.8%
Other (18 sites)	3680	14.3%	90.1%	Other (18 sites)	1086	3.9%	94.7%
Unknown	2551	9.9%	100%	Unknown	1505	5.3%	100%
Total	25,695			Total	28,166		

Table 3: The top 5 targeted CAPTCHA types on Kolotibablo and PixProfit, based on CAPTCHAs observed posing as workers.

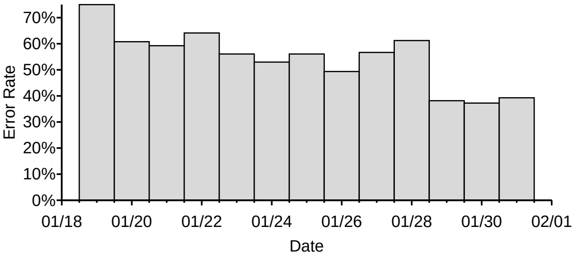


Figure 12: ImageToText error rate for the custom Asirra CAPTCHA over time.

CAPTCHA suitable for use with standard solver image APIs. Figure 11 shows an example. We wrote the instructions “Find all cats” in English, Chinese (Simpl.), Russian and Hindi across the top, as the majority of the workers speak one of these languages. We submitted this image once every three minutes to all services over 12 days. ImageToText displayed a remarkable adaptability to this new CAPTCHA type, successfully solving the CAPTCHA on average 39.9% of the time. Figure 12 shows the declining error rate for ImageToText; as time progresses, the workers become increasingly adept at solving the CAPTCHA. The next closest service was BeatCaptchas, which succeeded 20.4% of the time. The remaining services, excluding DeCaptcher, had success rates below 7%.

Coincidentally, as we were evaluating our own version of the Asirra CAPTCHA, on January 17th, 2010 DeCaptcher began offering an API method that supported it directly—albeit at $4 per 1,000 Asirra solves (double its base price). Microsoft had deployed the Asirra CAPTCHA on December 8th, 2009 on Club Bing. Demand for solving this CAPTCHA was apparently sufficiently strong enough that DeCaptcher took only five weeks to incorporate it into their service. We then performed the same experiment described above using the new DeCaptcher API method and received 1,494 responses. DeCaptcher successfully solved 696 (46.5%) requests with a median response time of 39 seconds, about 2.3 times its median of 17 seconds for regular CAPTCHAs. DeCaptcher appears to have factored in the longer solve times for the Asirra CAPTCHAs into the charged price. From what we can tell, though, DeCaptcher does not pay PixProfit workers double the amount for solving them, consequently increasing its profit margin on these new CAPTCHAs.

6.6 Targeted Sites

Customers of CAPTCHA-solving services target a number of different Web sites. Using our worker accounts on Kolotibablo and PixProfit, the public worker sites of Antigate and DeCaptcher, respectively, we can identify which Web sites are targeted by the customers of these services. Over the course of 82 days we recorded over 25,000 CAPTCHAs from Kolotibablo and 28,000 CAPTCHAs from PixProfit.

To identify the Web sites from which these CAPTCHAs originated, we first grouped the CAPTCHAs by image dimensions. Most groups consisted of a single CAPTCHA type, which we confirmed visually. We then attempted to identify the Web sites from which these CAPTCHAs were taken. In this manner we identified 90% of Kolotibablo CAPTCHAs and 94% of PixProfit CAPTCHAs.

Table 3 shows the top five CAPTCHA types we observed on Kolotibablo and PixProfit, with the remaining identified CAPTCHA types (18 CAPTCHA in both cases) representing 14% and 4% of the CAPTCHA volume on Kolotibablo and PixProfit respectively. Both distributions of CAPTCHA types are highly skewed: on PixProfit, the top two CAPTCHAs types represent 81% of the volume, with the top five accounting for 91%. Kolotibablo is not quite as concentrated, but the top five still account for 76% of its volume.

Clearly the markets for the services are different. Although Microsoft is by far the most common target for both, PixProfit tailors to CAPTCHAs from large global services (Google, Yahoo, AOL, and MySpace) whereas Russian sites otherwise dominate Kolotibablo (VKontakte.ru, Mail.ru, CAPTCHA.ru, Mamba.ru, and Yandex) — a demographic that correlates well with the observed worker fluency in Russian for Antigate (Table 2).

7 Discussion and Conclusion

By design, CAPTCHAs are simple and easy to solve by humans. Their "low-impact" quality makes them attractive to site operators who are wary of any defense that could turn away visitors. However, this same quality has made them easy to outsource to the global unskilled labor market. In this study, we have shed light on the business of solving CAPTCHAs, showing it to be a well-developed, highly-competitive industry with the capacity to solve on the order of a million CAPTCHAs per day. Wholesale and retail prices continue to decline, suggesting that this is a demand-limited market; an assertion further supported by our informal survey of several freelancer forums where workers in search of CAPTCHA-solving work greatly outnumber CAPTCHA-solving service recruitments. One may well ask: *Do* CAPTCHAs *actually work?* The answer depends on what it is that we expect CAPTCHAs to do.

Telling computers and humans apart. The original purpose of CAPTCHAs is to distinguish humans from machines. To this day, no completely general means of solving CAPTCHAs has emerged, nor is the cat-and-mouse game of creating automated solvers viable as a business model. In this regard, then, CAPTCHAs have succeeded.

Preventing automated site access. Today, the retail price for solving one million CAPTCHAs is as low as $1,000. Indeed, for well-motivated adversaries, CAPTCHAs are an acceptable cost of doing business when measured against the value of gaining access to the protected resource. E-mail spammers, for example, solve CAPTCHAs to gain access to Web mail accounts from which to send their advertisements, while blog spammers seek to acquire organic "clicks" and influence result placement on major search engines. Thus, in an absolute sense, CAPTCHAs do not prevent large-scale automated site access.

Limiting automated site access. However, it is short-sighted to evaluate CAPTCHAs as a defense in isolation. Rather, they exert friction on the underlying economic model and should be evaluated in terms of how efficiently they can undermine the attacker's profitability.

Put simply, a CAPTCHA reduces an attacker's expected profit by the cost of solving the CAPTCHA. If the attacker's revenue cannot cover this cost, CAPTCHAs as a defense mechanism have succeeded. Indeed, for many sites (e.g., low PageRank blogs), CAPTCHAs alone may be sufficient to dissuade abuse. For higher-value sites, CAPTCHAs place a utilization constraint on otherwise "free" resources, below which it makes no sense to target them. Taking e-mail spam as an example, let us suppose that each newly registered Web mail account can send some number of spam messages before being shut down. The marginal revenue per message is given by the average revenue per sale divided by the expected number of messages needed to generate a single sale. For pharmaceutical spam, Kanich *et al.* [14] estimate the marginal revenue per message to be roughly $0.00001; at $1 per 1,000 CAPTCHAs, a new Web mail account starts to break even only after about 100 messages sent.[9]

Thus, CAPTCHAs naturally limit site access to those attackers whose business models are efficient enough to be profitable in spite of these costs and act as a drag on profit for all actors. Indeed, MR. E reported that while his service had thousands of customers, 75% of traffic was generated by a small subset of them (5–10).

The role of CAPTCHAs today. Continuing our reasoning, the profitability of any particular scam is a function of three factors: the cost of CAPTCHA-solving, the effectiveness of any secondary defenses (e.g., SMS validation, account shutdowns, additional CAPTCHA screens, etc.) and the efficiency of the attacker's business model. As the cost of CAPTCHA solving decreases, a site operator must employ secondary defenses more aggressively to maintain a given level of fraud.

Unfortunately, secondary defenses are invariably more expensive both in infrastructure and customer impact when compared to CAPTCHAs. However, a key observation is that secondary defenses need only be deployed quickly enough to undermine profitability (e.g., within a certain number of messages sent, accounts registered per IP, etc.). Indeed, the optimal point for this transition is precisely the point at which the attacker "breaks even." Before this point it is preferable to use CAPTCHAs to minimize the cost burden to the site owner and the potential impact on legitimate users. While we do not believe that such economic models have been carefully developed by site owners, we see evidence that precisely this kind of tradeoff is being made. For example, a number of popular sites such as Google are now making aggressive use of secondary mechanisms to screen account sign-ups (e.g., SMS challenges), but *only* after a CAPTCHA is passed and some usage threshold is triggered (e.g., multiple sign-ups from the same IP address).[10]

In summary, we have argued that CAPTCHAs, while traditionally viewed as a *technological* impediment to an attacker, should more properly be regarded as an *economic* one, as witnessed by a robust and mature CAPTCHA-solving industry which bypasses the underly-

[9] These numbers should be taken with a grain of salt, both because the cited study is but a single data point, and because they studied SMTP-based spam, which generally has lower deliverability than Webmail-based spam. Anecdotally, the retail cost of Webmail-based delivery can be over 100 times more than via SMTP from raw bots.

[10] Anecdotally, this strategy appears effective for now and Gmail accounts on the underground market have gone from a typical asking price of $8/1,000, to being hard to come by at any price. We will not be surprised, however, if this mechanism leads to the monetization of smartphone botnets, or mobots [10], in response.

ing technological issue completely. Viewed in this light, CAPTCHAs are a low-impact mechanism that adds friction to the attacker's business model and thus minimizes the cost and legitimate user impact of heavier-weight secondary defenses. CAPTCHAs continue to serve this function, but as with most such defensive mechanisms, they simply work less efficiently over time.

Acknowledgments

We would like to thank the anonymous reviewers and our shepherd, Rachna Dhamija, for their feedback as well as Luis von Ahn for his input and discussion early on in the project. We also thank Jonathan Wilkins for granting us access to reCaptchaOCR, Jon Howell and Jeremy Elson for discussions about the Asirra CAPTCHA, and the volunteers who assisted in manual identification of targeted CAPTCHAs. We are particularly indebted to MR. E for his generosity and time in answering our questions and sharing his insights about the technical and business aspects of operating a CAPTCHA-solving service. Finally we would also like to thank Anastasia Levchenko and Ilya Kolupaev for their assistance. This work was supported in part by National Science Foundation grants NSF-0433668 and NSF-0831138, by the Office of Naval Research MURI grant N000140911081, and by generous research, operational and in-kind support from Yahoo, Microsoft, HP, Google, and the UCSD Center for Networked Systems (CNS). McCoy was supported by a CCC-CRA-NSF Computing Innovation Fellowship.

References

[1] BBC news PC stripper helps spam to spread. http://news.bbc.co.uk/2/hi/technology/7067962.stm.

[2] Ticketmaster, LLC v. RMG Technologies, Inc., et al 507 F.Supp.2d 1096 (C.D. Ca., October 16, 2007).

[3] E. Bursztein, S. Bethard, J. C. Mitchell, D. Jurafsky, and C. Fabry. How good are humans at solving CAPTCHAs? a large scale evaluation. In *IEEE S&P '10*, 2010.

[4] M. Chew and D. Tygar. Image recognition CAPTCHAs. In *Information Security, 7th International Conference, ISC 2004*, pages 268–279. Springer, 2004.

[5] D. Danchev. Inside India's CAPTCHA solving economy. http://blogs.zdnet.com/security/?p=1835, 2008.

[6] D. Danchev. Report: Google's reCAPTCHA flawed. http://blogs.zdnet.com/security/?p=5123, 2009.

[7] R. Datta, J. Li, and J. Z. Wang. Exploiting the Human-Machine Gap in Image Recognition for Designing CAPTCHAs. *IEEE Transactions on Information Forensics and Security*, 4(3):504–518, 2009.

[8] M. Egele, L. Bilge, E. Kirda, and C. Kruegel. CAPTCHA Smuggling: Hijacking Web Browsing Sessions to Create CAPTCHA Farms. In *The 25th Symposium On Applied Computing (SAC)*, pages 1865–1870. ACM, March 2010.

[9] J. Elson, J. R. Douceur, J. Howell, and J. Saul. Asirra: a CAPTCHA that exploits interest-aligned manual image categorization. In *CCS '07*, pages 366–374, New York, NY, USA, 2007. ACM.

[10] C. Fleizach, M. Liljenstam, P. Johansson, G. M. Voelker, and A. Méhes. Can You Infect Me Now? Malware Propagation in Mobile Phone Networks. In *Proceedings of the ACM Workshop on Recurring Malcode (WORM)*, Washington D.C., Nov. 2007.

[11] A. Hindle, M. W. Godfrey, and R. C. Holt. Reverse Engineering CAPTCHAs. In *Proc. of the 15th Working Conference on Reverse Engineering*, pages 59–68, 2008.

[12] L. Jassin-O'Rourke Group. Global Apparel Manufacturing Labor Cost Analysis 2008. http://www.tammonline.com/files/GlobalApparelLaborCostSummary2008.pdf, 2008.

[13] R. F. Jonell Baltazar, Joey Costoya. The heart of KOOBFACE: C&C and social network propagation. http://us.trendmicro.com/us/trendwatch/research-and-analysis/white-papers-and-articles/, October 2009.

[14] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage. Spamalytics: an empirical analysis of spam marketing conversion. In *CCS '08*, pages 3–14, New York, NY, USA, 2008. ACM.

[15] The Klingon language institute. http://www.kli.org, Accessed February 2010.

[16] G. Mori and J. Malik. Recognizing objects in adversarial clutter: Breaking a visual CAPTCHA. In *CVPR*, volume 1, pages 134–141, 2003.

[17] G. Moy, N. Jones, C. Harkless, and R. Potter. Distortion estimation techniques in solving visual CAPTCHAs. pages II: 23–28, 2004.

[18] PWNTcha. Pretend We're Not a Turing computer but a human antagonist. http://caca.zoy.org/wiki/PWNtcha.

[19] G. Sauer, H. Hochheiser, J. Feng, and J. Lazar. Towards a universally usable CAPTCHA. In *SOUPS '08*, 2008.

[20] Symantec. A captcha-solving service. http://www.symantec.com/connect/blogs/captcha-solving-service.

[21] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford. Captcha: Using hard AI problems for security. In *Advances in Cryptology - EUROCRYPT*, 2003.

[22] S.-Y. Wang, H. S. Baird, and J. L. Bentley. CAPTCHA challenge tradeoffs: Familiarity of strings versus degradation of images. In *ICPR '06*, 2006.

[23] J. Wilkins. Strong captcha guidelines v1.2. http://bitland.net/captcha.pdf.

[24] Xrumer. http://www.botmasternet.com/.

[25] J. Yan and A. S. El Ahmad. A low-cost attack on a Microsoft CAPTCHA. In *CCS '08*, pages 543–554, New York, NY, USA, 2008. ACM.

[26] J. Yan and A. S. El Ahmad. Usability of CAPTCHAs or usability issues in CAPTCHA design. In *SOUPS '08*, pages 44–52, New York, NY, USA, 2008. ACM.