unCaptcha
A Low-Resource Defeat of reCaptcha's Audio Challenge

Kevin Bock, Daven Patel, George Hughey, Dave Levin
Acknowledgements
Captchas

- **Completely Automated Public Turing test to tell Computers and Humans Apart**
- Captchas are the **primary defense** for many online services against bots
  - Prevents automated account creation
  - Bot service abuse
  - Artificial flow of information
Testing for humanity with reCaptcha

2009 – 2014

- Based on transcription
- Text-based challenges
- Acquired then defeated by Google

2014 – present

- Based on recorded user interaction
- Used by hundreds of thousands of sites
- “Easy on humans, tough on bots”
reCaptcha: How it works

- **Risk analysis engine** takes in information from the browser
  - Most importantly: **cookies**

- Each time a browser interacts with a Google service, that interaction is recorded with their cookie
  - Allows for the **noCaptcha reCaptcha**
reCaptcha Challenges

Image recognition

- Easy for most humans, hard for computers
- Data strengthens other services (e.g., street view, image search)

But what about visually impaired users?
reCaptcha Challenges

Audio recognition

- **Necessary** for visually impaired users
- As of this paper:
  - All digits: “Two… seven… three…”
  - With gaps between numbers
  - And some distortion

Easy for most humans, but is it hard for computers?
Attacking reCaptcha - Threat Model

- Previous works assumed **well-resourced attackers**
- Solutions/defeats were generally:
  - Offline
  - Requiring training data
  - On powerful computers
- What is “success”? 
  - 1% solving rate can be a “success”
Our Threat Model

- Assume a **low-resource attacker**
  - Need high success rate
  - Minimal training data

- All testing was done on a **free-tier** Amazon Elastic Computing
  *t2.micro* instance
  - 1GB of RAM
  - 1 virtual CPU
  - 8GB hard-drive

Actual footage of *t2.micro*
Talk is cheap

Why re-invent the wheel when Google, Apple, Amazon, etc. have already done it better?
Can we use Google against itself to solve captchas?
Can we leverage Speech-to-Text?

“one...six...eight...four...two...nine...eight...one...three...two”

1684298132
In reality

“won...sax...hate...flower...too...mine...ate...own...free...toad”

???????????
Segmentation

- Simple **amplitude analysis** to find split points
- Divide audio at periods of **silence**
Multiple Speech-to-Text Services

Submit each audio clip to six **free, public** speech-to-text services
Multiple Speech-to-Text Services

Submit each audio clip to six **free, public** speech-to-text services

- Google Cloud
- Bing
- IBM
- Sphinx

"one"
"bun"
"who"
"soon"
Phonetic Mapping

Speech-to-text services are not designed to work for *only* digits.
Ensembling

"one" → 1 → 1
"bun" → "bun" → 1
"who" → "who" → 2
"soon" → "soon" → X

Repeat for each digit
Demo
CREATE A NEW ACCOUNT

```
samuel72
```

```
password
```

```
password
```

email

```
I'm not a robot
```

[CAPTCHA]

[CHECKBOX]

[LOG IN]

By signing up, you agree to our Terms and that you have read our Privacy Policy and Content Policy.
Experimental Evaluation

- Overall accuracy; is it viable for a low-resource attacker?
- Benefits of phonetic mapping and ensembling?
- Limitations?
unCaptcha’s Overall Performance

91.99% Digit accuracy
80.31% Captcha success
Dealing with the poor performance of “6”

Replace all unknowns (“X”) with a guess of “6”
unCaptcha’s Overall Performance

91.99% Digit accuracy
80.31% Captcha success
unCaptcha’s Overall Performance

93.41% Digit accuracy

85.15% Captcha success
Benefit of Phonetic Mapping

Increased digit accuracy by 17% overall
Benefit of Ensembling

Increased captcha success by 78%
uncaptcha’s Speed

22.24 sec
Avg. time to solve

Main bottleneck: Service response time

Trade-off for a low-resource attack

19.22 sec
Avg. audio challenge

reCaptcha accepts answers

before a human could listen to the challenge
unCaptcha’s Speed

5.42 sec
Avg. time to solve

Main bottleneck: Service response time

Trade-off for a low-resource attack

19.22 sec
Avg. audio challenge

reCaptcha accepts answers before a human could listen to the challenge
Limitations

● Services are free, but have API limits

● ...yet reCaptcha is the **sole defense** against creating a new account on many of these services

● unCaptcha could theoretically be made **self-sufficient**
Future Recommendations

- Increase **vocabulary size**
  - Mitigate the boost of the phonetic mapping

```
"one"  "four"
"six"  "dinosaur"
"traffic"
"keyboard"
```
Future Recommendations

- Increase distortion
  - Negatively affect segmentation and accuracy
Future Recommendations

● Increase **complexity** of task
  ○ Semantic over syntactic

“draw a circle”

“type ‘car’ but **not** ‘dog’”

“type every other word in the following phrase…”

“type ‘bus’ **three times**”

“type only the words that **rhyme** in the following list…”

“type only the **English words** in the following list…”

“type only the **animals** in the following list…”
New reCaptcha Updates

- Disclosed to Google in March
  - Already “aware of the issue” as of April
- English **phrases** instead of **digits**
- Uses static instead of silence
- Better browser automation detection
Summary

- Boosts **per-digit accuracy** and **overall captcha success** by
  - **Ensembling** 6 online speech-to-text services
  - **Phonetically Mapping** their output to digits

85.15% Captcha success
5.42 sec Avg. time to solve

- unCaptcha proves that a **low-resource, high-accuracy** defeat of Google's reCaptcha system is possible

Check out project website at [http://uncaptcha.cs.umd.edu](http://uncaptcha.cs.umd.edu)
unCaptcha