University of Padova
Engineering Departement

Computer Engineering Master Degree

Invisible CAPPCHA

Candidate
Di Nardo Di Maio Raffaele

Supervisor
Prof. Migliardi Mauro

Co-Supervisors
Guerar Meriem

DD-MM-YYYY

Accademic Year 2020-2021

To my parents, that always help
me to be happy doing what I love
and support me reaching my goals.

“*Most people assume that once security software is installed, they're protected. This isn't the case. It's critical that companies be proactive in thinking about security on a long-term basis.*”
Kevin Mitnick

“*You have to learn the rules of the game. And then you have to play better than anyone else.*”
Albert Einstein

“*Si come il ferro s'arrugginisce sanza esercizio, e l'acqua si putrefà o nel freddo s'addiaccia, così lo 'ngegno sanza esercizio si guasta.*”
Leonardo da Vinci

Contents

1 Introduction **1**

2 State of the Art **3**
2.1 Traditional CAPTCHAs 4
2.1.1 Audio-based CAPTCHAs 5
2.1.2 Game-based CAPTCHAs 5
2.1.3 Image-based CAPTCHAs 6
2.1.4 Math CAPTCHAs 9
2.1.5 Slider CAPTCHAs 10
2.1.6 Text-based CAPTCHAs 11
2.1.7 Video-based CAPTCHAs 11
2.2 Modern CAPTCHAs 13
2.2.1 Biometrics-based CAPTCHAs 13
2.2.2 Behavioural-based CAPTCHAs 14
2.2.3 Sensor-based CAPTCHAs 15
2.3 CAPTCHA security 16

3 Invisible CAPPCHA **21**
3.1 Motion detection 21
3.1.1 Secure Element 21
3.2 Communication between Client and Server 21
3.2.1 Elliptic Curve Digital Signature Algorithm (ECDSA) . 21
3.3 Security 21
3.3.1 Threat model 21
3.3.2 Strength against known attacks 21

4 Side-channel attacks **23**
4.1 Local side-channel attacks 24
4.1.1 Passive 24
4.1.2 Active 25
4.2 Vicinity side-channel attacks 25

4.2.1 Passive 26
4.2.2 Active 26
4.3 Remote side-channel attacks 26
4.3.1 Passive 26
4.3.2 Active 27
4.4 Popular side-channel attacks 27
4.5 Authentication 28

Bibliography **28**

Chapter 1

Introduction

CAPTCHA (Completely Automated Public Turing Test to Tell Computers and Humans Apart) is a program used to distinguish human users from bots. A bot is a malicious application that automates a task, gathering useful information about user credentials or pretending to be a human interaction with Web application. Hence the term *"bot"* is an abbreviation of the words "software robot".
The CAPTCHAs are traditionally used in Web applications for[1]:

- **Online Polls**
  CAPTCHAs prevent the creation and the submission of a large number of votes, favouring a party.

- **Protecting Web Registration**
  CAPTCHAs prevent the creation of free mail account to bot instead of human users. The goal of the use of CAPTCHAs is to remove the possibility that the hacker could take advantages from the large amount of registrations.

- **Preventing comment spam**
  CAPTCHAs prevent the insertion of a large amount of posts made by bot on pages of social platforms or blogs.

- **Search engine bots**
  CAPTCHAs are used to guarantee that a website should be unindexed to prevent the reading of the page through search engine bots. The CAPTCHAs are added because the html tag, used to unindex the web page, doesn't guarantee unindexing.

- **E-Ticketing**
  CAPTCHAs prevent that a big events would sell out minutes after

tickets become available. In fact ticket scalpers that make large number of ticket purchases for big events.

- **Email spam**
  CAPTCHAs are used to verify that a human has sent the email.

- **Preventing Dictionary Attacks**
  CAPTCHAs prevent bot to guess the password of a specific user. The hacker could guess the password, taking it from a dictionary of passwords. The use of the CAPTCHA challenge prevents the iteration of the login phase made by the bot using all the words of the dictionary. After a certain number of failures POST requests, the CAPTCHA challenge is shown to the user.

- **Verifying digitized books**
  ReCAPTCHA can verify the contents of a scanned piece of paper analysing responses in CAPTCHA fields. A computer cannot identify all the words from a digital scan.
  The application submits two words to the user in the CAPTCHA challenge: the first one that the machine has already recognized and the other for which it can correctly associate a word. If the user types the two words and the first one was correctly detected, it assumes that also the second one is correct.
  In this case the second word is added to a set of words that are going to be added to other users' challenges. If the application receives enough responses with the same typed word related to the unknown word, the program extablishes that typed word is the CAPTCHA is related only to the first word and the challenge related to the second word is exploited by the application to scan digitally the paper.

Another useful application of CAPTCHA is the support to the authentication process. This application is going to be analysed in details in the next chapters, looking at the authentication from smartphone.
In Chapter 2 there is a description of the state of art of CAPTCHA, looking at types of CAPTCHA and the related tests from which this challenge is born.
In Chapter 3 Invisible CAPPCHA is described in details.
DESCRIPTION OF THE CONTENT OF THE CHAPTERS

Chapter 2

State of the Art

CAPTCHA takes inspiration and is related to three main elements[2]:

1. **Turing test**
   it's used to determine how much a machine can think like a human. The test is made by three figures: a human examiner, an human and a machine. The examiner asks some questions to both other two figures and, after a fixed amount of time, evaluates if the two answers are different or not.
   If they are similar w.r.t. the point of view of the examiner, the machine is an AI (Artificial Intelligence) similar to an human. The test is very important if the answers have many possibilities.

2. **Human-Computer Interaction (HCI)**
   according to cognitive psychology studies, a human process data in a specific way and this test evaluates the interaction between humans and machines. The HCI model is divided into five levels:

   - task level
   - semantic level
   - syntactic level
   - interactive level
   - a level of physical devices

   Then the obtained information is processed by:

   - reasoning
   - problem solving
   - skill acquisition

- error

3. **Human Interactive Proof (HIP)**
   it's used to make differentiation between machine and human users and computer user programs. The test require a type of interaction, that is simple to be done by human instead of bot. The main goals of this type of test are:

   - To differentiate the humans from the computers
   - To differentiate a category of the humans
   - To differentiate a specific human from the category of humans

   HIP has the test program that is subjected to the human and the computer. As a result, only a specific group of humans can positively solve the test and then the test results can be validated by the computer.

In order to guarantee a good level of security, a CAPTCHA has to satisfy the following requirements:

- The solution to the CAPTCHA isn't conditional and shouldn't depend on the user's language and/or age.
- The solution of the CAPTCHA must be easy for the humans and hard for the bots. Hence, humans in no longer than 30 seconds with very high success rate
- The creation of the CAPTCHA must not disturb the user privacy (not linked to the user).

2.1 Traditional CAPTCHAs

The traditional CAPTCHAs are based on the knowledge and correct insertion of solution by the user. These CAPTCHA schemes are designed to exploit character recognition, image understanding and speech recognition to guarantee that the challenges will successfully block bots.
The main types of these CAPTCHAs are described in the following sections but the details about specific implementations can be found in the article of Walid Khalifa Abdullah Hasan[3]. With respect to user experience, the most enjoyable traditional CAPTCHAs are usually the game-based and image-based ones but the most frustrating CAPTCHA is the text-based one[4]. A summary of usability and security issues is shown in Table 2.2.

2.1.1 Audio-based CAPTCHAs

This type of CAPTCHAs asks the user to type the words listened by an audio file (see Figure 2.1). It's developed for vision-impaired users. It usually has problems in usability related to the language dictionary, from which words are taken, and the similarity of the sound between several words. It has been proofed that this CAPTCHA is a hard task even for blind users, in fact only 46% of the challenges were solved by participants to an experiment[5].
One of the most popular CAPTCHAs is *Audio reCAPTCHA*, developed at Carnegie Mellon University and then bought by Google. In this scheme, the user needs to recognize and write a set of 8 spoken characters from a noisy audio file with background voices. If the user makes a mistake, the test declares that he's a bot.
Audio-based CAPTCHAs are vulnerable to many Automatic Speech Recognition (ASR) programs[6] but also Deep Learning techniques (e.g. DeepCRACk[7]). A good overview about results, obtained by several classification methods, is described in the work of Jennifer Tam et al.[8].

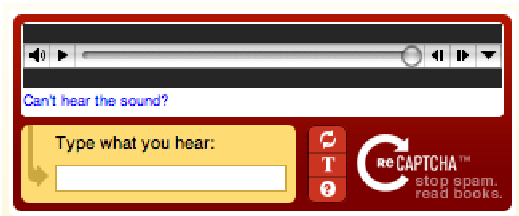


Figure 2.1: Example of audio-based CAPTCHA.

2.1.2 Game-based CAPTCHAs

This type of CAPTCHAs performs the verification of the user nature through a set of several kind of games (see Figure 2.2). The strength of this CAPTCHAs is relative to the comprehension phase of the rules that only humans can perform.
This type of CAPTCHAs is called *Dynamic Cognitive Game (DCG)* is usually developed using Flash and HTML5 with JavaScript. These technologies download the game code to the client and execute it locally.
The only difficult for the bot to attack the challenge is the encryption/obfuscation of the code. This strategy prevent the store of the code onto different internet domains. However for example, there exists a bot attack, called *Stream Relay Attack*, that obtains good results bypassing these challenges [9] (see Section 2.3).

Figure 2.2: Examples of game-based CAPTCHAs.

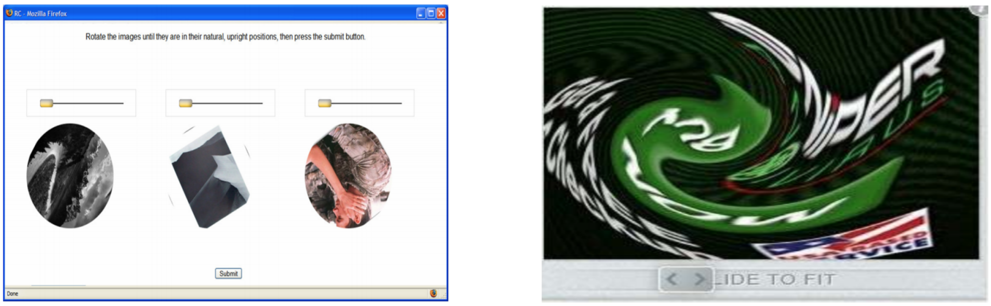
2.1.3 Image-based CAPTCHAs

Image-based CAPTCHAs require to understand a written text describing a task that needs an image evaluation to pass the test. This type of CAPTCHAs can be categorized into the following classes, looking to the task that the user needs to perform:

- **Click-based CAPTCHAs**
  this type of CAPTCHAs shows an image and a text that explains where the user needs to click (see Figure 2.3).

Figure 2.3: Example of click-based CAPTCHA.

- **Sliding image-based CAPTCHAs**
  this type of CAPTCHAs asks the user to use the slider to solve an image-based challenge such as adjusting the orientation of an image, selecting the correct form of an image, or moving a fragment of an image to the correct location (see Figure 2.4).

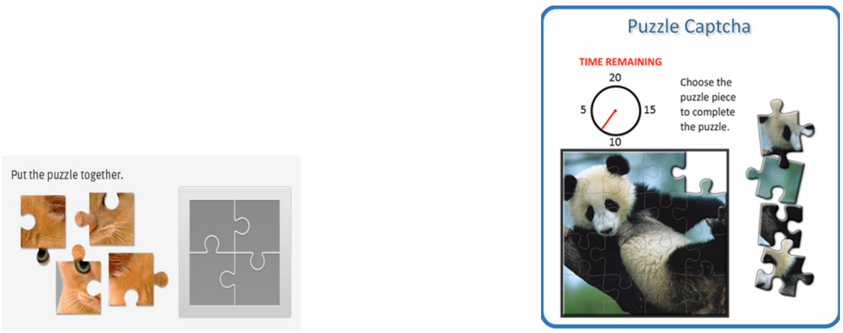


(a) Orientation based.

(b) Form based.

Figure 2.4: Examples of sliding image-based CAPTCHAs.

- **Drag & Drop-based CAPTCHAs**
  this type of CAPTCHAs usually asks the user to complete a visual puzzle, created by dividing a given image in a set of pieces[10] (see Figure 2.5a).
  The task isn't easy for users because this type of CAPTCHAs takes more time to solve the puzzle but the security level is very high[10]. To improve the usability of the CAPTCHA, there exists a variant of the puzzle-based CAPTCHA in which needs to insert only some pieces of the puzzle instead of completing the whole puzzle (see Figure 2.5b).



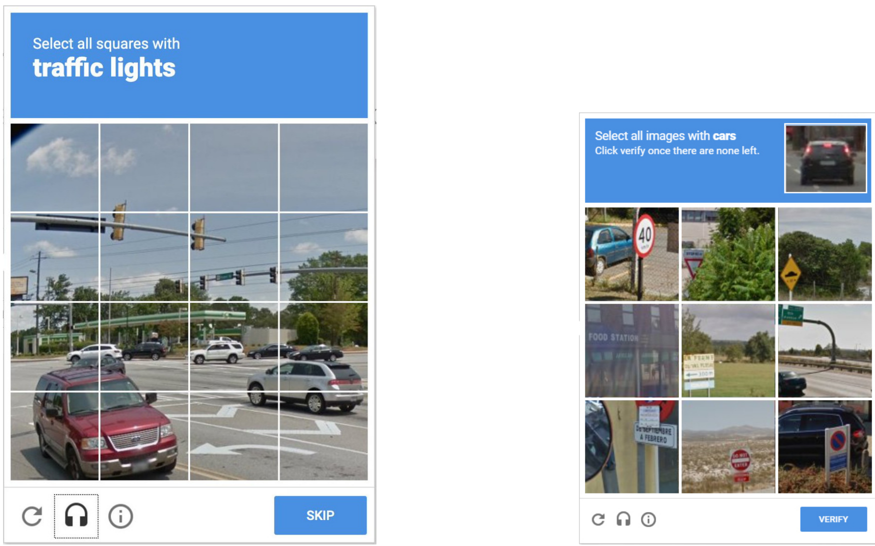
(a) Completing the puzzle.

(b) Inserting only some pieces.

Figure 2.5: Examples of puzzle-based CAPTCHAs.

- **Selection-based CAPTCHAs**
  the user usually needs to select the images that contain a requested subject. The set of images, on which the user needs to identify the subject, can be implemented in different ways, for example:
  - An image is divided into a set of sub-squares and each of them is a candidate image2.6a
  - There are many images, each one with a unique different subject (see Figure 2.6b)

  This type of CAPTCHAs is vulnerable to different Object Recognition techniques developed for Computer Vision purposes.
  An extension of this type of CAPTCHAs, called *FaceDCAPTCHA*, has been introduced[11]. It incorporates elements of face detection. The human brain is very effective in the process of natural face segmentation even if there are complex backgrounds. This approach increases the security efficiency because the Computer Vision programs can easily detect if there is a face, e.g. Viola-Jones algorithm[12], but have many problems differentiating real and non-real photographs of faces.
  Face, fingerprint and eye detections in images remain also a difficult challenge to be performed by computers. For this reason these results were used to develop a new variant of image-based CAPTCHA called *MB CAPTCHA*[13].



(a) With an image divided in sub-squares. (b) With several images.

Figure 2.6: Examples of selection-based CAPTCHAs.

- **Interactive-based CAPTCHAs**
  the user needs to discover a secret position in an image using mouse movement or swiping gesture

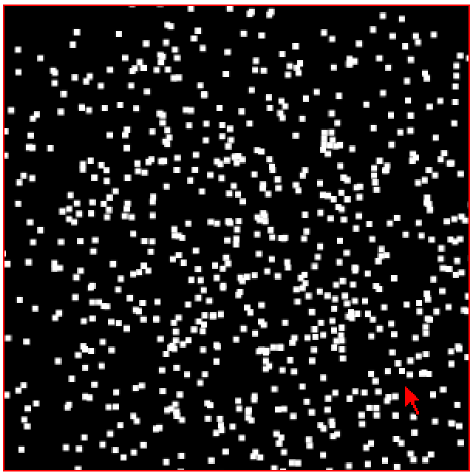
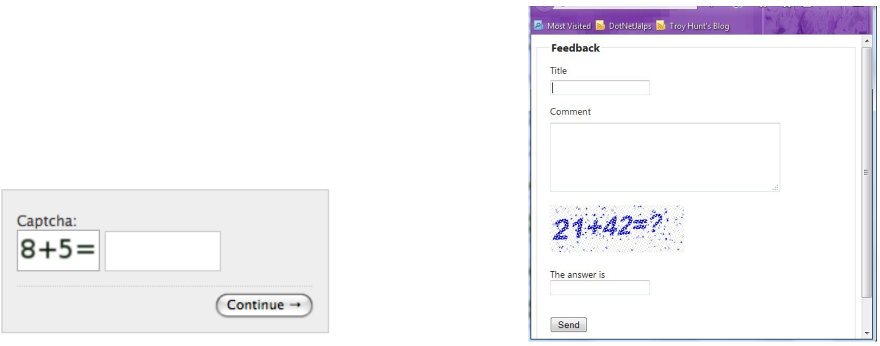


Figure 2.7: Example of interactive-based CAPTCHA.

2.1.4 Math CAPTCHAs

Looking to an operation specified in a frame, the user needs to insert the result in a text field. The operation is written in plain text or, to improve the security of this challenge, it's warped like text-based CAPTCHAs (Figure 2.8). These classical math-CAPTCHAs, also known as *arithmetic CAPTCHAs*, are vulnerable to OCR (Optical Character Recognition) techniques.

An advanced version of this CAPTCHA is used in the Quantum Ran-



(a) With plain text. (b) With wrapped text.

Figure 2.8: Example of arithmetic CAPTCHAs.

dom Bit Generator Service (QRBGS) sign-up Web Page[14] (see Figure 2.9). This type of CAPTCHA asks user to solve an advanced math expression. It prevents the use of free or commercial OCRs because many mathematical symbols are not considered in their detection algorithm. However, it's vulnerable to side-channel attack [14].
Hence many math symbols are wrongly translated by bot programs and the challenge is very secure. The only problem is that this CAPTCHA is very complex for normal users and many of them could not solve the challenge correctly.

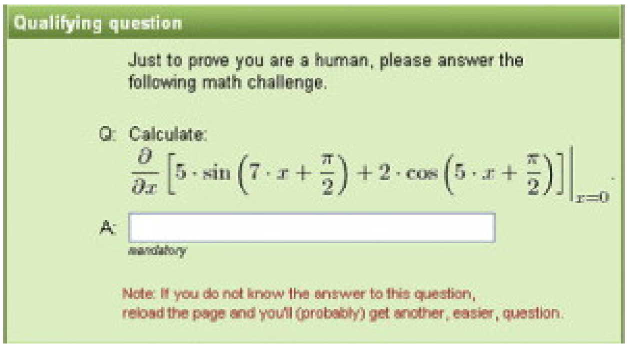


Figure 2.9: Example of Quantum Random Bit Generator Service (QRBGS) sign-up Web Page [14].

2.1.5 Slider CAPTCHAs

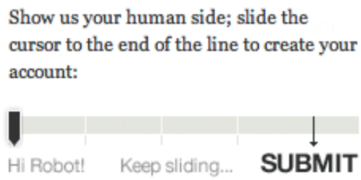
Slider CAPTCHAs only asks users to move the slider across the screen. Hence, image recognition is not part of the challenge to be classified as a human.
The most popular CAPTCHAs are the following:

- **CAPTCHA used by Taobao.com**
  it asks the user to drag the slider from the start to the end of the sliding bar to verify his identity (see Figure 2.10a).

- **CAPTCHA used by TheyMakeApps.com**
  it asks the user to move the slider to the end of the line to submit a form[15] (see Figure 2.10b).

Different variations of this type of CAPTCHAs have been bypassed with a simple JavaScript code and puppeteer.

(a) Example of TaoBao.com CAPTCHA.



(b) Example of TheyMakeApps.com CAPTCHA.

Figure 2.10: Examples of slider CAPTCHAs.

2.1.6 Text-based CAPTCHAs

In text-based CAPTCHA schemes a random series of wrapped characters and/or numbers is displayed on the screen inside an image(see Figure 2.11). The user needs to understand which are the characters that composes the shown sequence and then type them inside a text-field. The text-based CAPTCHAs can be also classified in three main classes looking to the type of wrapped characters:

- **2D**
  the digits are wrapped on a 2D plane, parallel to the screen plane
- **3D**
  the digits are wrapped on a 3D plane oriented in the space and then a 2D image is taken from a specific point of view

This type of CAPTCHAs is vulnerable to several type of attacks, related to Computer Vision techniques, that are:

- OCR techniques[16]
- Segmentation techniques (e.g. DECAPTCHA[17])
- Machine Learning and Deep Learning techniques

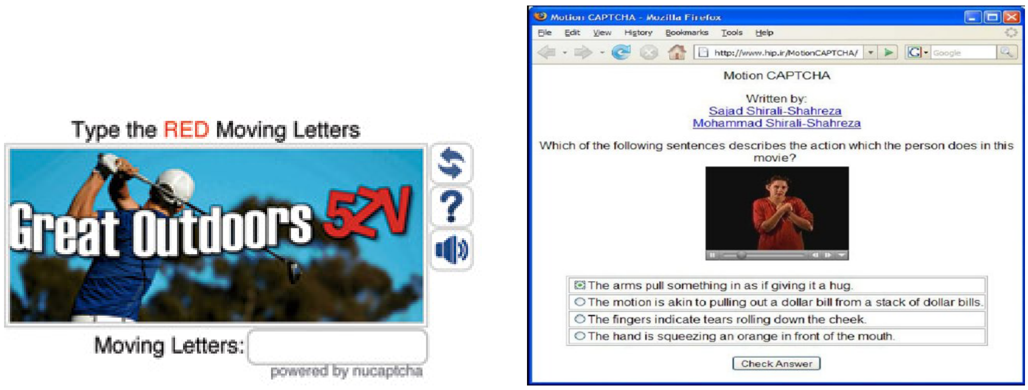
In the design phase of a text-based CAPTCHA there are many issues, related to Computer Vision techniques, to be considered. For each of them, there is usually a solution in the design phase of the CAPTCHA that reduces the possibility that the challenge would be broken by a bot[17].

2.1.7 Video-based CAPTCHAs

This type of CAPTCHA is not very common because of the weight of the file to be downloaded[3]. The traditional video-based CAPTCHA is composed

Figure 2.11: Example of text-based CAPTCHA.

by a video in which a sliding text is shown (see Figure 2.12a). The user needs to type this message in a text field to pass the challenge. Some implementations of this type of CAPTCHAs are vulnerable to machine learning attacks. Another variant of this CAPTCHA is the *Motion CAPTCHA*[18], developed by M. Shirali-shahreza and S. Shirali-shahreza, in which the user needs to watch a video. Then he needs to select which action was performed in the played file, choosing it from multiple answers (see Figure 2.12b). The strength of these implementations of CAPTCHAs depends on the relationship between the multiple choices submitted to the user[19].
A similar implementation of the previous variant, it's the one developed by Kluever et al. in which the user watches a video and needs to write three words that describe what he sees[20]. The same authors also performed a tag frequency-based attack to evaluate the security of their CAPTCHA schemes achieving a success rate of 13%.



(a) Example of sliding text in video. (b) Example of Motion CAPTCHA[18].

Figure 2.12: Examples of video-based CAPTCHAs.

2.2 Modern CAPTCHAs

The type of CAPTCHAs and authentication mechanisms described in the following section are far from traditional CAPTCHAs and aren't based on cognitive knowledge of the human user but on other parameters, such as behavioural analysis and sensors readings. In the following sections there is a summary of the most known CAPTCHA schemes of this type.

2.2.1 Biometrics-based CAPTCHAs

The most known authentication mechanisms, that use biometric parameters of the user, are based on the following CAPTCHA schemes:

- **Bio-CAPTCHA voice-based Authentication**
  This authentication method was developed starting from good results reached in the authentication phase of cloud systems (Alexa for Amazon, Siri for Apple, Cortana for windows)[21]. This particular implementation uses a random voice-based password challenge. This password changes at every login of the user and this method uses CAPTCHA challenge to provide unpredictability and ambiguity to the authentication process. The experiments reveals that unauthorized access probability decreases, while it keeps high usability because it needs only a mic.

- **rtCAPTCHA**
  this type of authentication method is a Real-time CAPTCHA that asks users to perform some tasks like smile, blink or nod in front of the camera of the mobile phone. The recorded video is sent to the service provider that checks if in the file, there is the expected user performing the required action.
  This implementation of CAPTCHA solves many problems of similar CAPTCHAs that are also based on liveness mechanisms and video capture. The attackers can extract patterns or features from existing or captured images and embed them into a new generated video in attack in the compromised device.
  In the work of Erkam Uzun, Simon Pak Ho Chung, Irfan Essa and Wenke Lee[22], there is a detailed comparison between other similar authentication mechanisms and rtCAPTCHA, looking to all possible Computer Vision attacks.

2.2.2 Behavioural-based CAPTCHAs

In 2014 Google announced that today's Artificial Intelligence can solve even the most difficult variant of text-based CAPTCHAs at 99.8% accuracy[23]. For this reason, the company develops the following CAPTCHA schemes:

- **Google no CAPTCHA**
  Google developed in 2015 a new CAPTCHA system that is simpler than traditional CAPTCHAs in terms of user interaction[24]. This CAPTCHA system is composed by two layers of protection:

  1. Checkbox *"I'm not a robot"* to be clicked by user as in Figure 2.13 (or image-based CAPTCHA on mobile devices)

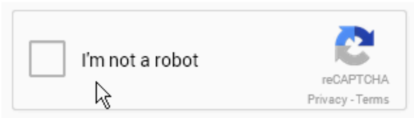
  

  Figure 2.13: Example of Google no CAPTCHA checkbox.

  2. Traditional text-based CAPTCHA with two warped words

  The second layer is reached only if the user doesn't succeed in the first one. For the checkbox step, the application evaluates in background the user's behaviour (e.g. the mouse movement, where the users click, how long they linger over a checkbox). Then the program performs an *advanced risk analysis*, by looking results of first step but also spam traffic and passed/failed CAPTCHAs. It understands in this way if the test is passed or not.
  The tests done confirms that this phase was very inefficient and many times the first layer failed even if a human user was performing correctly the task. A problem of this type of CAPTCHA is that many attacks exploits the image-based CAPTCHA and text-based CAPTCHA using attacks based on known Computer Vision techniques or their variants (e.g. CAPTCHA breaker made by Suphannee Sivakorn, Jason Polakis and Angelos D. Keromytis[25]).

- **Google Invisible ReCAPTCHA**
  It's a top layer over the *Google noCAPTCHA v2.0*, adding the option to bind directly to the form's submit element[24]. It usually requires the use of cookies, used to track the user's behaviour. There exist two version of this CAPTCHA:

  - **ReCAPTCHA v2.0**
    it was developed in 2017. It's not really invisible because Privacy & Policy badge must be included on every page of app or website in which the CAPTCHA is used. Computer Vision and Artificial Intelligence algorithms can break the challenges by recognizing object in the pictures in the image-based CAPTCHA phase.
  - **ReCAPTCHA v3.0**
    it was developed in 2018. With constantly analyzing human behavior, mouse movements, typing speed and other features incorporated into NO CAPTCHA technology, Google collected enough sample data to perfectly fine-tune their Google invisible reCAPTCHA v2.0 with this new version. This type of CAPTCHA uses Artificial Intelligence and Machine Learning probability scores, hostname, timestamp and anction validations.
    Google removes image recognition and looking only the score, it evaluates if the user is a human or a bot. The main difference w.r.t. previous versions is that this CAPTCHA returns a probability score (*risk score*) in the range $[0.0, 1.0]$: *0.0* if the user is a bot, *1.0* otherwise. The administrator of the website can decide what range of scores he wants to manage, declaring the the site is under attack and what actions need to be performed.
    Some characteristics related to this version of *Invisible ReCAPTCHA* are:
    * If a user accesses a Web page using incognito mode or private mode, he is classified with a very low score (*high risk*).
    * If a human is wrongly classified as a bot, the user can login into its Google account to increase its score. If this doesn't change the classification, you cannot do anything else.

2.2.3 Sensor-based CAPTCHAs

This type of devices have natively many sensors, like gyroscope and accelerometer, and the CAPTCHA schemes, described in the following sections, exploit their presence to improve security of the authentication.

- **Completely Automated Public Physical test to tell Computer and Humans Apart (CAPPCHA)**
  this is a way to enforce the PIN authentication phase by mobile phone[26]. The user needs to tilt the device to a specified angle specified on the screen (see Figure 2.14). The CAPPCHA security is based on the *Secure Element* (*SE*) present in the device. It prevents brute force, side

channel and recording attacks. The usability results are good and then some of the comments made by users were considered in the implementation.

Figure 2.14: CAPPCHA and PIN authentication[26].

- **Invisible CAPPCHA**
  It will be described in details in Chapter 3.

2.3 CAPTCHA security

The process used for breaking CAPTCHAs is organized into the following phases[27]:

1. **Pre-processing phase**
   In this phase, several techniques are applied to remove background, separate foreground from the background, to delete noise and to remove some particular pattern (e.g. Canny Detection and Scale-Invariant Feature Transform (SIFT) application).

2. **Attack phase**
   the following techniques are usually applied:

   - ***Object Segmentation attacks***
     Segmentation techniques (e.g. vertical histogram, colorfilling, snake segmentation and JSEG) are used to split the CAPTCHA image into segments to facilitate recognition
   - ***Object recognition attacks***
     The most used techniques are pattern matching (e.g. shape context matching, correlation algorithm]), OCR recognition, SIFT and machine learning.

- ***Random Guess Attacks***
The attacker's program tries to break the CAPTCHA scheme by guessing the correct answer. This attack is effective on CAPTCHAs with few number of different challenges.
- ***Human Solver Relay Attacks***
The bot forwards the CAPTCHA challenge to a remote human worker that will solve it.

Many CAPTCHAs have yet the following known issues:

- **Session issue**
Some types of CAPTCHAs have a big issue because they don't destroy the session, after the correct answer is inserted by the user[1].
Hence, the hacker can crack following accesses using the same session id with the related solution of the challenge, after connecting to the web page of CAPTCHA. In this way the attacker can make hundreds of requests before the session expires and the previous operation must be computed again.

- **Resilience to both automated and human solver relay attacks**
Many CAPTCHA schemes are designed to be robust against a possible AI attack but the new generation of CAPTCHA involves the use of remote bot or human solver. Traditional CAPTCHA schemes are vulnerable to this type of attacks.
Invisible reCAPTCHA and other academic proposals haven't been attacked yet, but works over thousands of different IP addresses and simulate the human behavior. Sensor-based CAPTCHAs are also vulnerable to solver relay attacks. An exception of this issue is the *Invisible CAPPCHA*, that will be analysed in following sections and it's designed to block this type of attacks.

- **Limited number of challenges**
An issue of sensor-based CAPTCHA schemes is the limited number of challenges because the design of many usable gestures is very hard. This problem could be solved relying on trusted hardware.

- **Trade-off between Friction-heavy and Frictionless CAPTCHAs**
A trade-off between usability and security aspects is always considered analysing CAPTCHA schemes. This condition is highlighted in behavioural and sensor-based CAPTCHAs.

- **User's privacy**
Sensor-based and behavioural CAPTCHAs usually send useful infor-

mation to a remote server that analyses it to establish if the user was a human or a bot. If an hacker attacks the server side of this application, he can access to users' private data.
In some CAPTCHAs, the information are evaluated on the client side by a trusted hardware and the server receives only the results of the analysis. In this case, we need to be sure that trusted hardware is secure enough to guarantee privacy of user's information.

- **Compatibility with different devices**
  Many CAPTCHA schemes, e.g. behavioural ones, use specific forms factors but a good challenge should be compatible with different factors.

Type	Scheme	Usability issues	Security
Audio	*Audio reCAPTCHA*	Issues of recognition: • Knowledge of English dictionary by the user. • Some character sounds very similar to others.	It's vulnerable to: • ASR programs. • Deep Learning and ML techniques.
Game	*Dynamic Cognitive Game (DCG)*	Comprehension of rules.	Vulnerable to Stream Relay Attack.
Image	*Click-based* *Drag & Drop-based* *Sliding-based* *Selection-based* *Interactive-based*	Difficulty in identification of images caused by: • Blur of images. • Low vision condition.	Vulnerable to: • Segmentation techniques • Deep Learning and ML techniques • OCR techniques
Math	*Arithmetic* *QRBGS*	It requires basic or advanced math knowledge.	Vulnerable to: • OCR techniques • Side-channel attacks
Slider	Taobao.com TheyMakeApps.com	Simple and intuitive interaction.	Simple bypassed by Javascript code and pupeteer.
Text	*2D* *3D*	Many problems have to be solved by user: • Multiple fonts • Font size • Blurred Letters • Wave Motion	It can be identified by: • OCR technique • Segmentation techniques • Deep Learning and ML techniques
Video	*Motion CAPTCHA*	Heavy file to be downloaded	

Table 2.1: Survey of main types of traditional CAPTCHAs[10].

Type	Scheme	Usability issues	Security
Audio	*Audio reCAPTCHA*	Issues of recognition: • Knowledge of English dictionary by the user. • Some character sounds very similar to others.	It's vulnerable to: • ASR programs. • Deep Learning and ML techniques.
Game	*Dynamic Cognitive Game (DCG)*	Comprehension of rules.	Vulnerable to Stream Relay Attack
Image	*Click-based* *Drag & Drop-based* *Sliding-based* *Selection-based* *Interactive-based*	Difficulty in identification of images caused by: • Blur of images. • Low vision condition.	Vulnerable to: • Segmentation techniques • Deep Learning and ML techniques OCR techniques
Math	*Arithmetic* *QRBGS*	It requires basic or advanced math knowledge.	Vulnerable to: • OCR techniques • Side-channel attacks
Slider	Taobao.com TheyMakeApps.com	Simple and intuitive interaction	Simple bypassed by Javascript code and pupeeteer
Text	*2D* *3D*	Many problems have to be solved by user: • Multiple fonts • Font size • Blurred Letters • Wave Motion	It can be identified by: • OCR technique • Segmentation techniques • Deep Learning and ML techniques
Video	*Motion CAPTCHA*	Heavy file to be downloaded	

Table 2.2: Survey of main types of traditional CAPTCHAs[10].

Chapter 3

Invisible CAPPCHA

The *Invisible CAPPCHA* is an evolution of CAPPCHA, in terms of usability[28]. The main difference with respect to CAPPCHA is that the challenge isn't explicitly submitted to user but it's hidden behind the PIN authentication phase. This type of challenge works only on smartphones as its ancestor.
The micro-movements of the device, generated by the interaction of the user with the touch-screen, are evaluated by the *Secure Element* (*SE*). Then credentials are shared with the remote Service Provider if the input is inserted by a human or not.

3.1 Motion detection

3.1.1 Secure Element

3.2 Communication between Client and Server

3.2.1 Elliptic Curve Digital Signature Algorithm (ECDSA)

3.3 Security

3.3.1 Threat model

3.3.2 Strength against known attacks

Chapter 4

Side-channel attacks

A side-channel attack is an attack in which the malicious user exploits a side-information of transmitted encrypted data, to give access to user private data. This type of extra information is usually: timing information, power consumption, electromagnetic radiations, sound and so on.
The first types of side-channel attacks requires the physical access to the victim's device. Nowadays, side-channel attacks are evolved and can be conducted by remote hackers using malicious code (e.g. cache-timing attacks, DRAM row buffer attacks), even exploiting information from sensors on mobile devices[33].
Side-channel attacks can be classified as:

- **active**
  the hacker influences the behaviour of the victim's device

- **passive**
  the attacker only analyses the leaking information

Another categorization can be the following one:

- Local attacks

- Vicinity attacks

- Remote attacks

In the following sections there is a survey of the most popular attacks, organized with respect to the previous classifications (see Table 4.1).

	Local	Vicinity	Remote
Passive	Power Analysis Differential Computation Analysis Shoulder Surfing and Reflections Hand/Device Movements	Network Traffic Analysis USB Power Analysis Wi-Fi Signal Monitoring	Linux-inherited procfs Leaks Data-Usage Statistics Page Deduplication Microarchitectural Attacks Sensor-based Keyloggers Fingerprinting Devices/Users Location Inference Speech Recognition Soundcomber
Active	Clock/Power Glitching Electromagnetic Fault Injection (EMFI) Laser/Optical Faults Temperature Variation Differential Computation Analysis NAND Mirroring	Network Traffic Analysis	Rowhammer

Table 4.1: Survey of the most popular side-channel attacks[33].

4.1 Local side-channel attacks

The attacker needs to get the target device or to be very near to it. In many cases the hacker physically needs to manipulate the the device or to obtain access to the chip.

4.1.1 Passive

The following attacks are used to break cryptographic system implementations:

- **Power Analysis**
  this type of attacks are based on the analysis of the power variations in transistors. There exist several attacks[30]:
  - **Simple Power Analysis (SPA)**
    the attacker analyses the power consumption of the system, that depends on the microprocessor used. This analysis can be useful to understand which operations are performed by different implementations of cryptographic algorithm (e.g. RSA, DES).

  - **Differential Power Analysis (DPA)**
    these attacks collect data and then makes statistical analysis and error correction techniques from data to extract information correlated to secret keys.
  - **High Order DPA (HO-DPA)**
    While DPA obtains information across a single event, HO-DPA correlates between multiple cryptographic sub-operations.

- **Differential Computation Analysis**
- **Shoulder Surfing and Reflections**
- **Hand/Device Movements**

4.1.2 Active

- **Clock/Power Glitching**
- **Electromagnetic Fault Injection (EMFI)**
- **Laser/Optical Faults**
- **Temperature Variation**
- **Differential Computation Analysis**
- **NAND Mirroring**

4.2 Vicinity side-channel attacks

The attacker needs to wiretap or eavesdrop the network communication of the victim or to be in the neighbourhood of the target.

4.2.1 Passive

- **Network Traffic Analysis**
- **USB Power Analysis**
- **Wi-Fi Signal Monitoring**

4.2.2 Active

- **Network Traffic Analysis**

4.3 Remote side-channel attacks

4.3.1 Passive

- **Linux-inherited procfs Leaks**
- **Data-Usage Statistics**
- **Page Deduplication**
- **Microarchitectural Attacks**
- **Sensor-based Keyloggers**
- **Fingerprinting Devices/Users**
- **Location Inference**
- **Speech Recognition**

- **Soundcomber**

4.3.2 Active

- **Rowhammer**

4.4 Popular side-channel attacks

Through the use of side-channel information, the attacker can also detect keys used to encrypt the communication in the most know cryptographic models.

- **Analysis of information from sensors**
  The main sensors, that are usually exploited in an attack by an hacker, are[]:
  - Location sensors (e.g. GPS, proximity)
  - Motion sensors (e.g. accelerometer, gyroscope, magnetometer)
  - Environmental sensors (e.g. for ambient light, temperature, barometer)
  - Biometric sensors for wearable devices (e.g. heart rate sensor, ECG)
  - Audio sensors (microphone)
  - Video sensors (camera)

For example a Web-application works between two parties: the client and the server. For this reason the communication channel is usually encrypted and the requests made by the user work through the *HTTPS* protocol. This solution isn't enough to prevent an attacker to exploit reserved data because each web page has a distinct size, loads resources of different sizes. Hence the attacker can fingerprint the page even if HTTPS protocol is used.
Another cause of these attack on Web-services is given by the trend of Web to work on Stateful Protocols, providing better performance to the client by keeping track of the connection information. TCP session for example works on Stateful Protocol because both systems maintain information about the session itself during its life[29].

4.5 Authentication

Bibliography

[1] Sarika Choudhary, Ritika Saroha, Yatan Dahiya, and Sachin Choudhary, "Understanding CAPTCHA: Text and Audio Based Captcha with its Applications" in *International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3(6), pp. 106-115, 2013.*

[2] Darko Brodić, Alessia Amelio, Radmila Janković, "Exploring the influence of CAPTCHA types to the users response time by statistical analysis" in *Multimedia Tools and Applications, vol. 77, pp. 12293–12329, 2017.*

[3] Walid Khalifa Abdullah Hasan, "A Survey of Current Research on Captcha" in *International Journal of Computer Science Education in Schools (IJCSES), vol. 7, pp. 141–157, 2016.*

[4] Ruti Gafni, Idan Nagar, "CAPTCHA – Security affecting user experience" in *Issues in Informing Science and Information Technology, vol. 13, pp. 63-77, 2016.*

[5] G. Sauer, J. Holman, J. Lazar, H. Hochheiser, and J. Feng, "Accessible privacy and security: A universally usable human-interaction proof tool" in *Univers. Access Inf. Soc., vol. 9, no. 3, p. 239–248, Aug. 2010.*

[6] Jennifer Tam, Jiri Simsa, David Huggins-Daines, Luis von Ahn, Manuel Blum, "Improving Audio CAPTCHAs" in *Symposium On Usable Privacy and Security (SOUPS), 2008.*

[7] William Aiken, Hyoungshick Kim, "POSTER: DeepCRACk: Using Deep Learning to Automatically CRack Audio CAPTCHAs" in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security (ASIACCS), 2018.*

[8] Jennifer Tam, Jiri Simsa, Sean Hyde, Luis von Ahn, "Breaking audio CAPTCHAs" in *Advances in Neural Information Processing Systems, pp. 1625–1632, 2009.*

[9] Manar Mohamed, Song Gao, Nitesh Saxena, Chengcui Zhang, "Dynamic cognitive game captcha usability and detection of streaming-based farming" in *The Workshop on Usable Security (USEC), co-located with NDSS, 2014.*

[10] Ved Prakash Singh, Preet Pal, "Survey of Different Types of CAPTCHA" in *International Journal of Computer Science and Information Technologies (IJCSIT), vol. 5(2), 2014.*

[11] Goswami G, Powell BM, Vatsa M, Singh R, Noore A, "FaceD-CAPTCHA: face detection based color image CAPTCHA" in *Future Generation Computer Systems, vol. 31(2), pp. 59–69, 2014.*

[12] Wen-yao Lu, Ming Yang, "Face Detection Based on Viola-Jones Algorithm Applying Composite Features" in *International Conference on Robots & Intelligent System (ICRIS), pp. 82-85, 2019.*

[13] Brian M. Powell, Abhishek Kumar, Jatin Thapar, Gaurav Goswami, Mayank Vatsa, Richa Singh, Afzel Noore, "A multibiometrics-based CAPTCHA for improved online security" in *IEEE 8th International Conference on Biometrics Theory, Applications and Systems, 2016.*

[14] Carlos Javier Hernandez-Castro, Arturo Ribagorda, "Pitfalls in CAPTCHA design and implementation: The Math CAPTCHA, a case study" in *Computers & Security, vol. 29(1), pp. 141-157, 2010.*

[15] Luke Wroblewski, "A Sliding Alternative to CAPTCHA?", *2010.*

[16] Silky Azad, Kiran Jain,, "Captcha: Attacks and weaknesses against OCR technology" in *Global Journal of Computer Science and Technology, 2013.*

[17] E. Bursztein, M. Martin, J. Mitchell, "Text-based CAPTCHA strengths and weaknesses" in *Proc. 18th ACM Conference on Computer and Communications Security (CCS), pp. 125–138, 2011.*

[18] M. Shirali-Shahreza and S. Shirali-Shahreza, "Motion CAPTCHA" in *2008 Conference on Human System Interactions, Krakow, pp. 1042-1044, 2008.*

[19] Kameswara Rao Kavya Sri, Gnana Sai, "A Novel Video CAPTCHA Technique to Prevent BOT Attacks" in *International Conference on Computational Modeling and Security (CMS 2016), Procedia Computer Science, vol. 85, pp. 236–240, 2016.*

[20] K. A. Kluever, R. Zanibbi, "Balancing usability and security in a video captcha" in *Proceedings of the 5th Symposium on Usable Privacy and Security, 2009.*

[21] Omar Ahmed Hedaia, Ahmed Shawish, Hana Houssein, Hala Zayed, "Bio-CAPTCHA Voice-Based Authentication Technique for Better Security and Usability in Cloud Computing" in *International Journal of Service Science Management Engineering and Technology, vol. 11(2), pp. 59-79, 2020.*

[22] Erkam Uzun, Simon Chung, "rtCaptcha: A Real-Time Captcha Based Liveness Detection System" in *The Network and Distributed System Security Symposium (NDSS), Georgia Institute of Technology, 2018.*

[23] Vinay Shet, "Are you a robot? Introducing "No CAPTCHA reCAPTCHA"", *2014.*

[24] TehnoBlog, "Google no Captcha + INVISIBLE reCaptcha–First Experience Results Review", *2019.*

[25] Suphannee Sivakorn, Jason Polakis, Angelos D. Keromytis, "I'm not a human: Breaking the Google reCAPTCHA" in *Black Hat, pp. 1–12, 2016.*

[26] Meriem Guerar, Alessio Merlo, Mauro Migliardi, "Completely automated public physical test to tell computers and humans apart: A usability study on mobile devices" in *Future Generation Computer Systems, vol. 82, pp. 617–630, 2018.*

[27] Narges Roshanbin, James Miller, "A survey and analysis of current CAPTCHA approaches" in *Journal of Web Engineering, vol. 12, issue 1–2, pp. 1–40, 2013.*

[28] Meriem Guerar, Alessio Merlo, Mauro Migliardi, Francesco Palmieri, "Invisible CAPPCHA: A usable mechanism to distinguish between malware and humans on the mobile IoT" in *Computers & Security, vol. 78, pp. 255–266, 2018.*

[29] Shuo Chen, Rui Wang, XiaoFeng Wang, Kehuan Zhang, "Side-Channel Leaks in Web Applications: A Reality Today, a Challenge Tomorrow" in *2010 IEEE Symposium on Security and Privacy, pp. 191-20, 2010.*

[30] Paul Kocher, Joshua Jaffe, Benjamin Jun, "Introduction to Differential Power Analysis and Related Attacks" in *Cryptography Research, 1998.*

[31] Kazuo Sakiyama, Takanori Machida, Arisa Matsubara, Yunfeng Kuai, Yu-ichi Hayashi, Takaaki Mizuki, Noriyuki Miura, Makoto Nagata, "Authentication Using Side-Channel Information." in *IACR Cryptol. ePrint Arch., 2015, 834.*

[32] A. Nahapetian, "Side-channel attacks on mobile and wearable systems" in *13th IEEE Annual Consumer Communications & Networking Conference (CCNC), pp. 243-247, 2016.*

[33] R. Spreitzer, V. Moonsamy, T. Korak and S. Mangard, "Systematic Classification of Side-Channel Attacks: A Case Study for Mobile Devices" in *IEEE Communications Surveys & Tutorials, vol. 20, no. 1, pp. 465-488, 2018.*

Acknowledgements

Professor Migliardi

I would like to express my very great appreciation to Dr Guerar

I would like to express my gratitude to University of Padova for the study path I performed. The uncertainty about the future and the idea of being far from needed cyber security skills have become a stimulus to improve myself. I learn a lot and I got hooked on the programming, starting from zero level of it, thanks to the professors' professionalism and knowledge. During the last five years, I've changed and now I spend programming all my free time. Thanks to University because professors follow my thirst of knowledge and I grew up, living alone and really becoming an adult.

Thanks to staff,

Thanks to other company,

I would like to express my gratitude to my family that taught me to never give up. In particular thank to my sister that, with her great experience in University course, has been a reference for my study attitude and perseverance.

Thanks to my grandmother Concetta that, with her smiles, has left to me happiness during my darker and difficult periods.

Thanks to my grandmother Maria that, was always worried for my health issues and tries always to give me up using her food.

Thanks to Cristina,

Thanks to Francesca, because even if we had many commitments we have always found 5 minutes to stay together and to support the other one in his/her choices.

Thanks to Davide,

Thanks to Giuseppe, Aurora, Alessia and Sara,

Thanks to Elia,

Thanks to Lorenzo,