PIN and Password Security

Michael Amista’ - Gabriel Rovesti

December 7th, 2023

Advanced Topics in Computer Network and Security

2023-2024

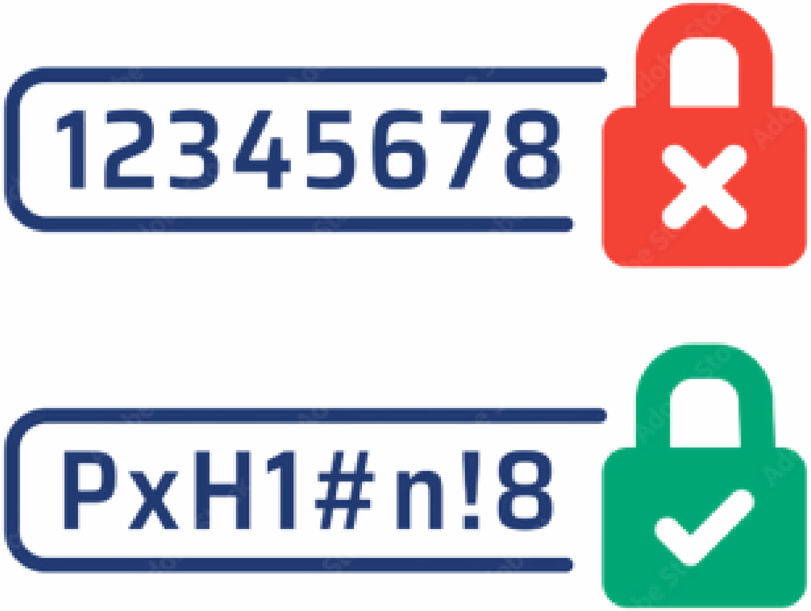
Università degli Studi di Padova

Dipartimento Matematica

PIN sounds good... but is it?

Easy and simple...

But secure?

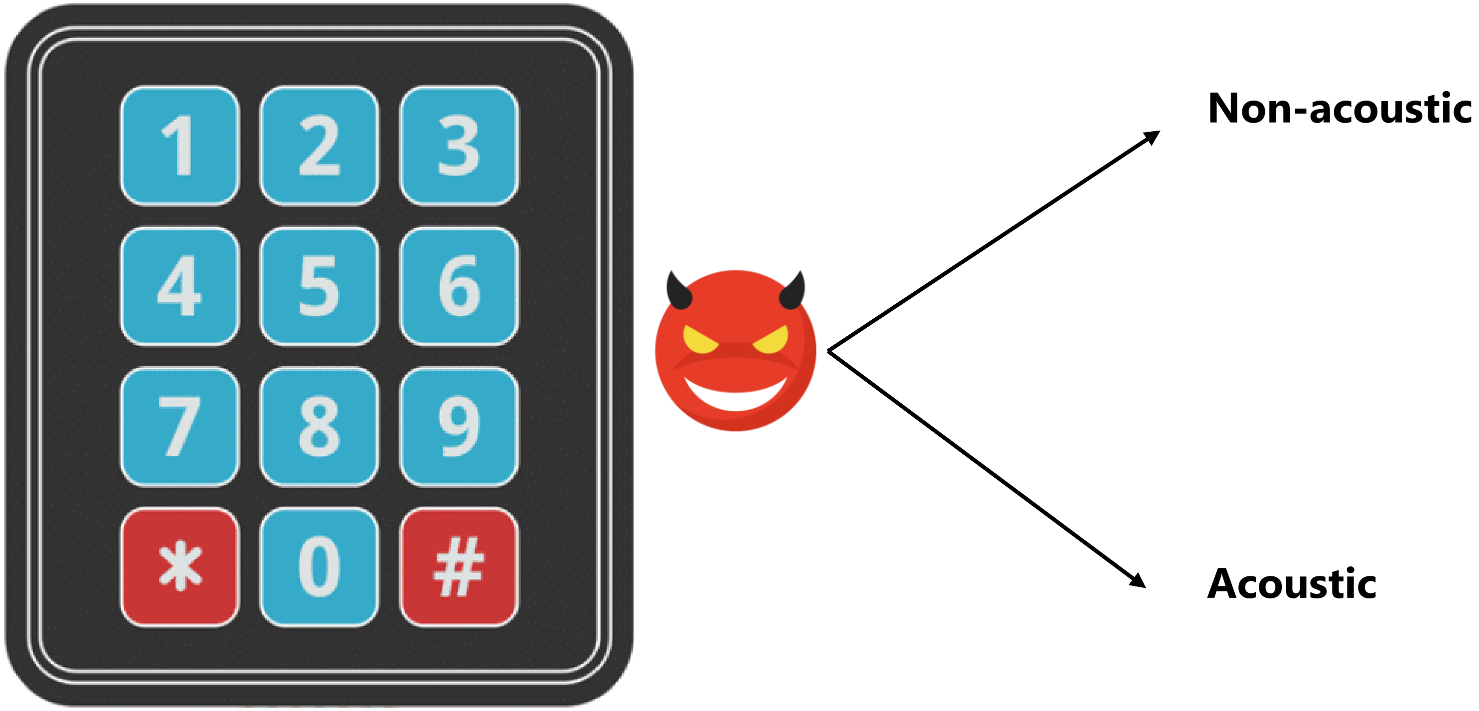


Where are PINs used?

ATMs and PoS

Smartphones

How are ATM PINs exploited?



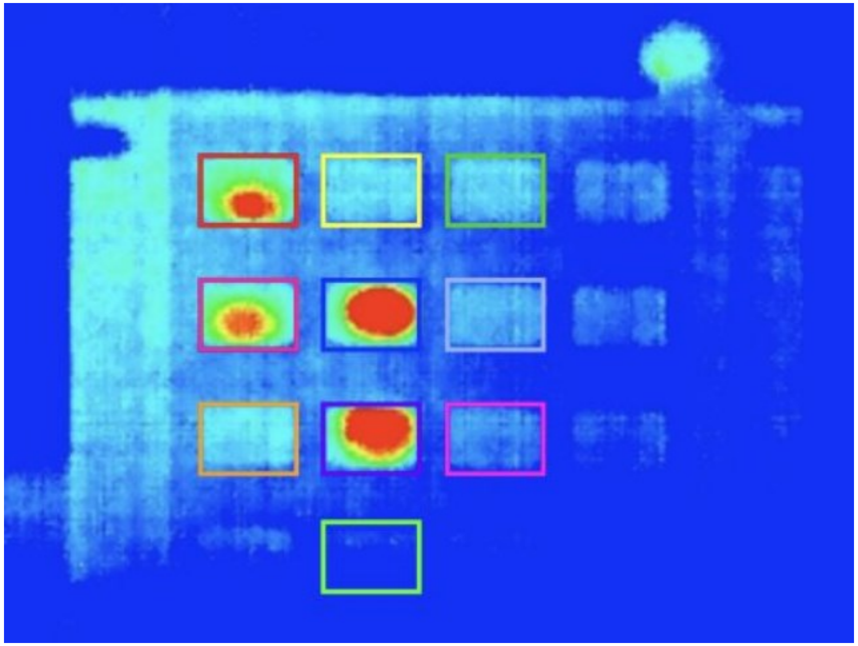
Non-acoustic

Acoustic

How to track and study PINs?

Non-acoustic

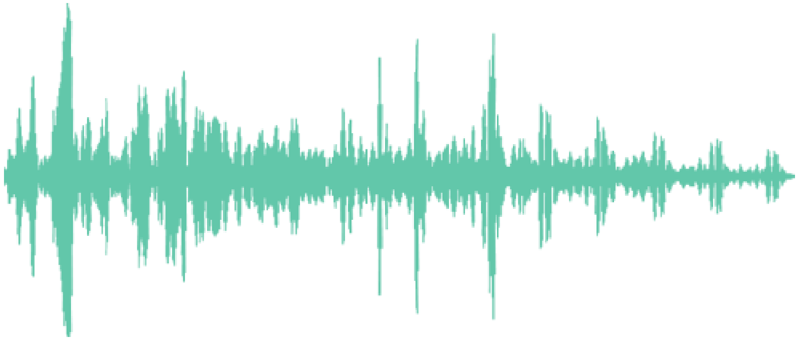
- While the user is typing the password
- Recover keystrokes by electromagnetic emanations
- Observation via a thermal camera to identify key presses on the keypad



How to track and study PINs?

Acoustic

- Each key emits a characteristic sound



- Able to construct a dictionary attack to bruteforce and reconstruct words
- Multiple microphones to triangulate the keys positions
- Alternatives: SSH traffic and video → **measure the timing between presses works better!**

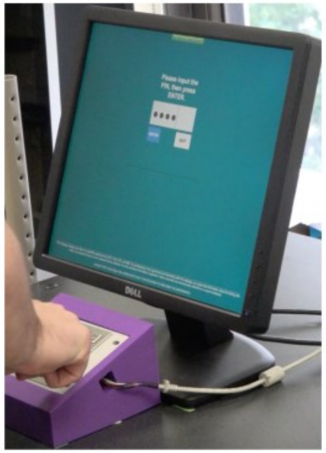
Keystroke Timing

- Measures the distance between consecutive keystrokes of subsequent keys

- The adversary can infer keystroke timings by using audio when the user is typing the PIN

- Filter timestamps of keys pressed from keypad sound normalizing the samples

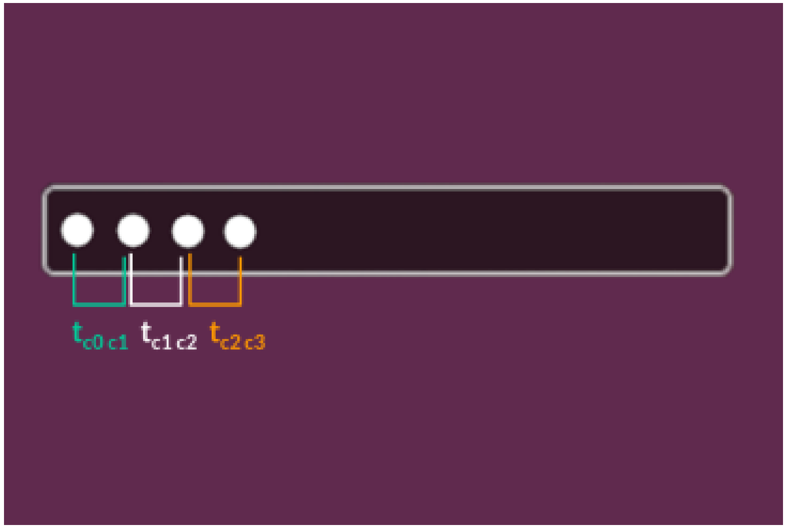
- Timing is also based on *observation (video)*



Keystroke Timing

- Ranking PINs based on the Euclidean distance between subsequent keys in each PIN (but *how many are they typed?*)

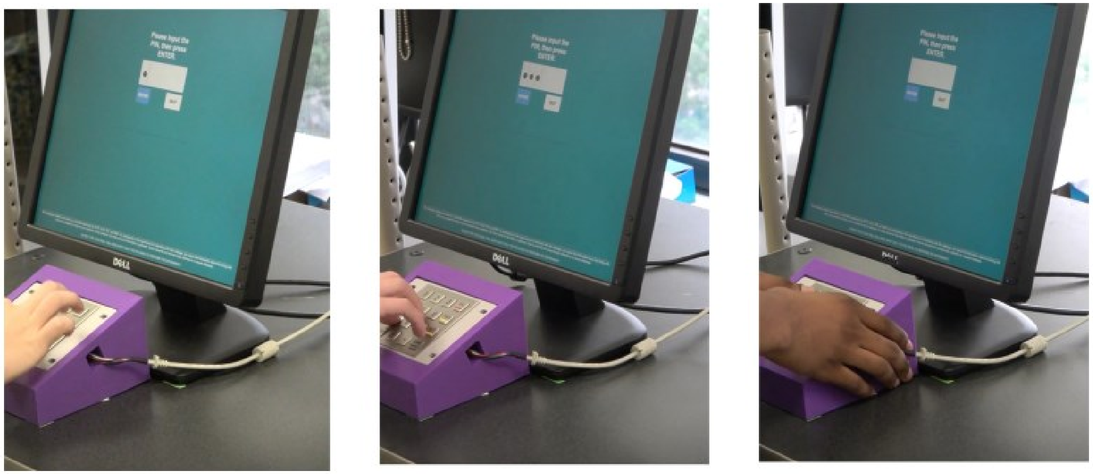
- Distance vector from a sequence of inter-keystroke timings inferred from audio feedback



- Example: PIN 5566 is [0,1,0]

Typing Behavior

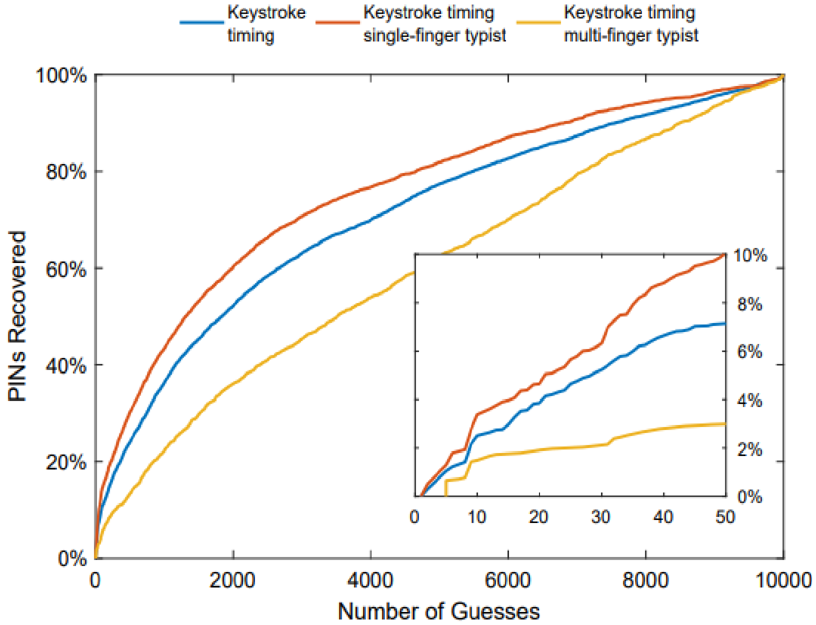
- The adversary can typically observe (video) whether the user is typing with:
  - one finger (*single* typists)
  - more fingers (*multi-finger* typists)



- We can combine thermal camera (*video*) with keystroke timing (*audio*)

Typing Behavior + Keystroke Timing

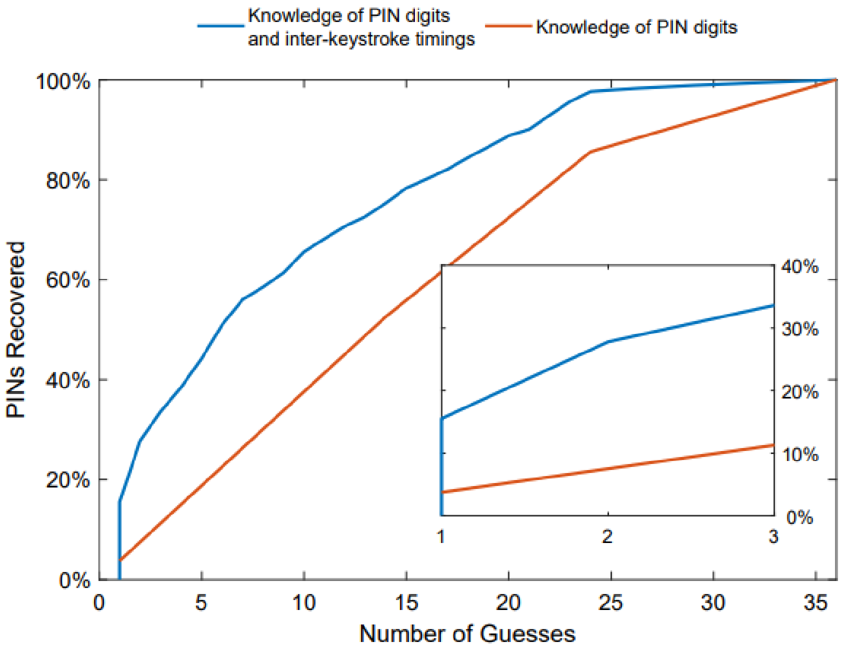
- Single-finger typists:
  - less PINs – more timing
- Two-finger typists:
  - more PINs – less timing
- and so on…



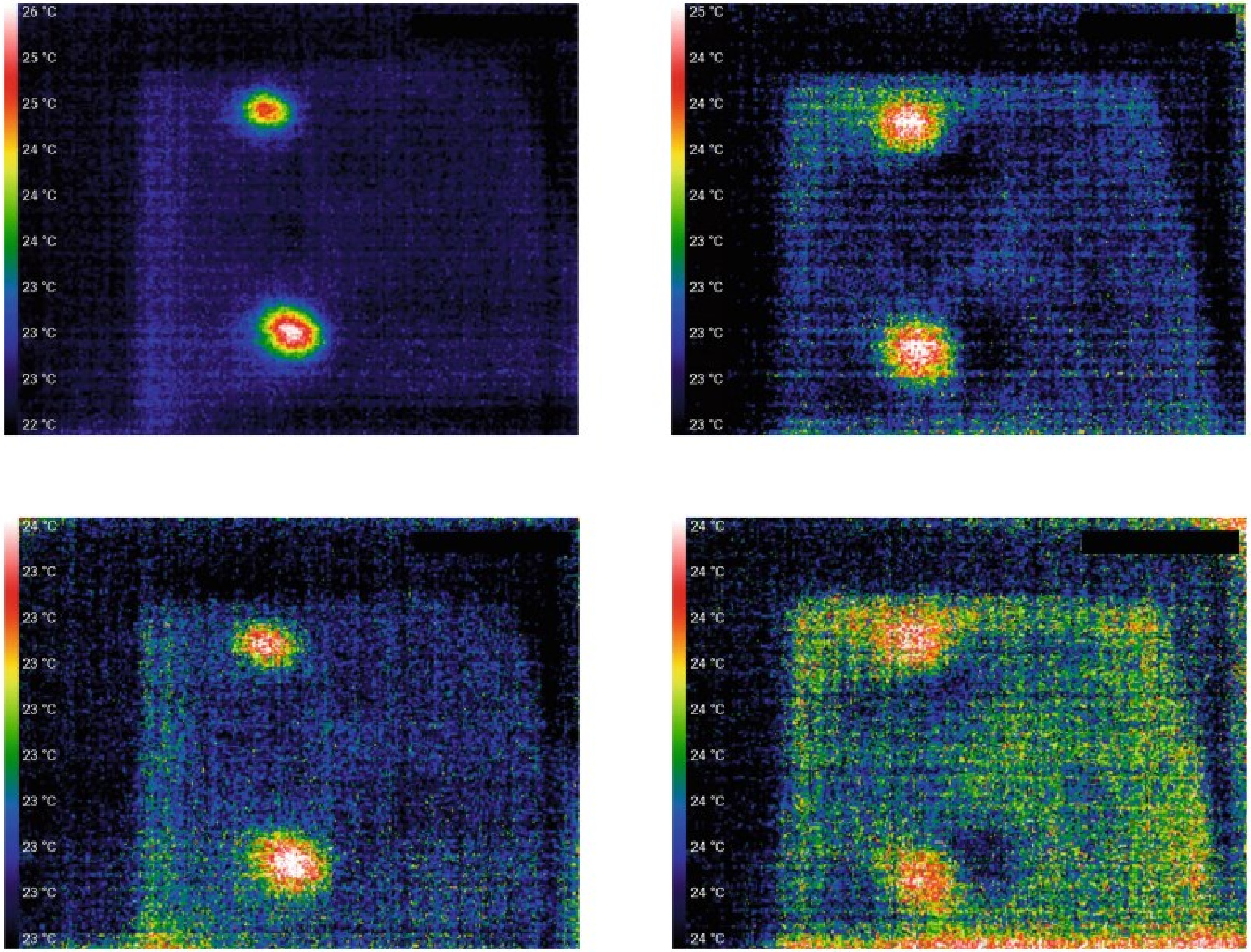
- The inter-keystroke timing (when) is not representative of the Euclidean distance (how many)….
- …which means higher guesses over single-finger compared to multi-finger!

Knowledge of which keys have been pressed

- The adversary may have visibility of the keypad in how the user moves his hand via a thermal camera
- Knowledge of one digit alone reduces the search space by a linear factor (**first** or **last**) over the remaining digits
- The keystroke timing combined with knowledge of keys lead to a higher PIN recover rate



How much time to get the keys pressed?



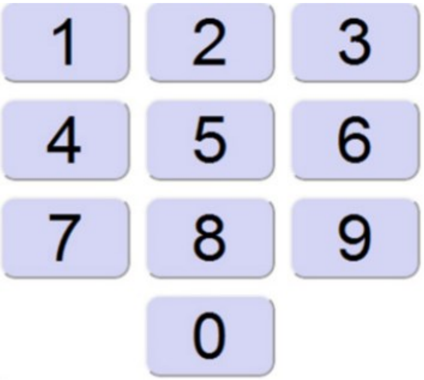
Analysis of PINs Guessing Probability

- Choosing PINs uniformly at random from the entire PIN space is not the best strategy

- The inter-keystroke timing, combined with other sources of information, can lead to higher PINs recovery rate even with same guessing probability thank to audio feedback
  - Keystroke timing + knowledge of first/last digit = x14
  - Keystroke timing + observation of keys = +15%

- Strong correlation between Euclidean distance (which keys) and inter-keystroke timings (how many) lead, combined with previous ones, to better results

PIN Selection Policies



- We said that PINs must be easy to remember....
- ....but how much can we trust users?

- We must combine:
  - **usability** – easy to remember
  - **security** – randomly distributed and difficult to guess

- PINs should have:
  - *totally different* numbers and using last numerical digits (e.g., 7,8,9)
  - different from things *easy-to-remember* (e.g., birthdays, events, etc.)
  - *security-by-design* (a policy which improves the safety without forcing the user to remember something difficult)

How PINs are distributed in the real world

- Occurrence frequency of the PINs (Power law distribution)
  - First digits are pressed more than others
  - Easier for an attacker to guess the first digit than the other digits

- PINs generated from *dates and years*
  - A good portion of selected PINs are made like this

- PINs generated from *arithmetic operations*
  - Some users combine simple things (e.g., addition/multiplication, etc.)
  - Just a minority over the others

- PINs *with close proximity*
  - Consecutive numbers are more selected given they're easy to use
  - Another good portion of PINs are like this

How PINs can be more effectively chosen

- How to balance *easy-to-remember/hard-to-master* in PINs?
  - **Enforce selection policies** (secure but not too strict)
  - Application of said policies in everyday life (e.g., locking mobiles)
  - PINs usable safely and simply in all real-life contexts

- How to balance *easy-to-remember/hard-to-master* in PINs?
  - *4-free*: 4-digit PIN without any restriction
  - *4-short*: 4-digit PIN where the 200 most popular PINs were not allowed
  - *4-long*: same rules as 4-short + without any consecutive number
  - *6-free:* 6-digit PIN without any restriction
  - *6-long:* 6-digit PIN without any consecutive number

- Scale of easy-to-remember (1 = "very easy" up to 5 = "very difficult")

How PINs can be chosen and selected

- The study gave the following results:
  - PINs tend to be chosen between the most popular ones
    - remembrance is worse (*4-short* vs *4-free*)
  - PINs tend to be chosen with consecutive numbers more
    - *4-long* (more secure) vs *4-short* (less secure) = harder for users
  - This holds even In longer PINs
    - remembrance is worse (*6-free* vs *6-long*)
  - **Good compromise**: choose *4-short* PINs over *4-free*

How PINs can be chosen and selected

- In conclusion:
  - Entropy is higher on longer PINs or longer subsequences
  - Odd numbers are usually less frequent
  - It can be useful to analyze more geographical areas and PINs databases

- It would be better to:
  - Useful to enforce a PIN blacklists policy (e.g. avoiding popular PINs)
  - Impose more different chars, easy length and easy words without enforcing the users

Thanks for the attention… but now it’s your turn!

ANY Questions?