Paper

Aidan Fray

June 3, 2019

1 Introduction

The increasing use of public key cryptography by instant messaging and secure email means key fingerprint verification is an ever more important task. One of the biggest risks to the security of the communication channel is a Man-in-the-middle (MiTM) attack . A successful MiTH attack can circumvent the encryption as it allows an attacker to read all the encrypted data. A countermeasure for this is the verification of each parties' fingerprint. **[...]**

2 Literature Review

2.1 Fingerprint representation comparison

Fingerprints are used to quickly compare the similarity of two keys. The comparison occurs between the digest of a strong one-way hash function. Historically these have been represented as a hexadecimal string and users were tasked with manually comparing fingerprints between two substrates, for example, a monitor screen and a business card. We will refer to this process throughout the paper as the *"authentication ceremony"*.
It has been shown that the average human can only hold 7-digits worth of data in our working memory[1]. This rules out the possibility of comparing full digests. For examples SHA-1 is 40 hex digits (160-bit) and, therefore, will make it difficult for effective comparison. Therefore , if the process is to involve humans interaction, there is a need for schemes that work effectively with our abilities and senses.

Research in this area has taken various encoding schemes and compared their fallibility to impersonated attacks. Impersonation of a key would involve brute force of the encoding schemes through the mass creation of keys. The main elements of comparison have been accuracy of attack detection and time to complete the comparison. In this context we will consider the metrics a measure of security and usability respectively.

Effective schemes

Results from the literature consistently show the effectiveness of language based encodings (Words or Sentences). This was shown in [2], [3] and [4] with the accuracy of these schemes all being above 94.00%. In all cases these were the best scemes in terms of accuracy. The exception to this is the work performed by **Hsiao, *et al.***[5] in 2009 with Words achieving an abnormal accuracy of 63.00% (Comparisons of methodology will be discussed in later sections).

Aside from textual representations were graphical schemes. Examples of schemes assessed were: Random Art[6], Flag[7], T-Flag[8], Vash[1], OpenSSH visual host key and Unicorns[2] to name a few. Accuracy from these schemes were mixed with positive results obtained by **Hsiao, *et al.***[5] where their best accuracy was achived by Random Art (94.00%) and Flag Ext.[3](88.00%). On the other hand, the worst accuracy was obtained by Flag(50%). The only other paper assessing graphical representations was the work of **Tan *et al.***[3] where they also achieved mixed results with accuracies ranging from 46% to 90%.
In terms of usability of graphical schemes the literature concurred on their ease-of-use. In terms of speed of comparison all schemes consistently obtained some of the quickest comparisons times. This is also supported indirectly from the the comparison results of alternative literature [4][2] where with these in consideration graphical schemes still remain the quickest overall. In terms of research into the performance of graphical schemes, the literature does not contain an extensive review with only two papers including graphical schemes in their comparison. This is, therefore, a candidate for further research.

Experimentation comparison

2.2 Fingerprint representation schemes

Another area of research is investigations into the actual physical encodings of the hash digest. This section will briefly discuss the current research available on the creation and security of actual encoding schemes. The actual details of the operation of the schemes are outside the scope of this literature review, therefore, minimal attention will be allocated to these details.

Some of the oldest preliminary work into visual encoding schemes was performed

[1] https://github.com/thevash/vash

[2] https://unicornify.pictures/

[3] The authors' own improvement on Flag

by **Adrian Perrig *et al***[6]. in the creation of their scheme "Random Art" in 1999. The motivation for creating such a scheme was the perceived flaws in the ways humans verify and compare written information. As mentioned in previous sections visual encoding schemes have been shown to have mixed success, with low security being one of their most alarming flaws. This research laid the foundation for further work in analysing the security of visual encoding schemes.

Further research into the creation of unique visual hash schemes have been performed by **C Ellison *et al.*** [7] (Flag), **Yue-Hsun Lin *et al.***[8] (T-Flag) and work by **M. Olembo *et al.***[9]. Each publication has provided a new way to visually represent a key fingerprint. Alongside the academic literature, there are more informally presented methods of visual fingerprints such as Unicorns[4] and Robots[5]. This list is by no means exhaustive but is used to depict the amount of research and work invested into graphical hash representations.

One paper of note is the preliminary work performed by **D Loss *et al.***[10] in their *"An analysis of the OpenSSH fingerprint visualization algorithm"* where their aim was to spur on further research with their initial findings into the security of the OpenSSH scheme. The authors claim that the use of the algorithm in OpenSSH is only heuristically defined and there is a need for a formal proof of its security.
The paper proposed a number of ways to generate similar fingerprints. The methods proposed were: Naive brute force, Graph Theory, and brute force of a full visual set. They were only able to produce only very basic results and have proposed a large amount of potential further work. Since the paper's publication in 2009, there seems to have been no research building on the work of the authors. This highlights a current gap in the available literature.

Minimal research has also focused on basic textual fingerprint representations and their respective security. Work by **A. Karole** and **N. Saxena**[11] looked into ways to improve the security of a textual representation. This research aim was to improve the secure device pairing process of comparing two numerical values. The devices used (Nokia 6030b; Mid-range devices at the time of publication) and the SAS compared results in findings that are not directly applicable in a fingerprint comparison context.

A more specific subsection of textual fingerprints is the use of words and sentences to encode hash digests. Some of the first work in this area was produced by **Juola** and **Zimmermann** [12] and their work in generating a word list where phonetic distinctiveness was prioritised. Each word is mapped to a single byte. The unique aspect of the word list is the separation of "even" and "odd" words where "even" byte positions are sample from the even-list and "odd" from the odd-list. This effectively creates two sub-word lists. The maximisation of linguistic distinctiveness of these word lists were maximised through the use of

[4] https://unicornify.pictures/
[5] https://github.com/e1ven/Robohash

a Genetic algorithm. The paper also includes a study on effective measures of "linguistic distances" of words and provided an in-depth discussion into these areas.

Overall the paper provides a foundation for formalising the creation of effective wordlists. A limitation is the lack of empirical data gathered on the performance. However, this was later evaluated in work by Dechand *et al.* [2] and shown to be an effective encoding scheme.

Other research of note is work by **M. Goodrich *et al.***[13] called *Loud and Clear: Human-Verifiable Authentication Based on Audio.* As the name suggests the authors were researching ways to improve current methods of secure device pairing. The unique aspect of this work is the use of a Text-to-Speech system reading out syntactically correct English sentences. The sentences are based on a MadLibs[6] where static placeholders were replaced with potential words.
The work into a potential wordlist can be seen as an extension to the work performed by Juola and Zimmermann[12] as they aimed to emulate the techniques used in PGPfone. The paper's finding are limited by the lack of empirical data backing up claims made by the author as the systems performance and security are only theoretically assessed.

Aside from this research, there have been further informal implementations of fingerprint encodings. The first being by **Michael Rogers**[7]. Rogers' implementation is a program designed to map fingerprints to pseudo-random poems. This implementation was again, empirically evaluated by Dechand *et al.*[2]. Older work by **N. Haller** with the S/KEY[14] shows the implementation of a system designed to represent a hash as a series of six short words. However, this system is designed for a one-time-password purpose and only provides word mappings for basic human usability of the password and not within a fingerprint verification context. Therefore, the wordlist has not been designed with pronounceability in mind.
A very recent implementation of a word list can be found in Pretty Easy Privacy (pEp) implementation of TrustWords[8]. The unique aspect of TrustWords is its mapping of a single word to 16-bits. In comparison to other literature, this is the highest number of bits-per-word seen. Full mappings (no duplication of words) would, therefore, require 2^{16} words in the dictionary and arguably is higher than most users vocabulary. this deviation from the norm has not been currently backed up by research. Moreover the main RFC documentation still remains in a draft stage and states *"It is for further study, what minimal number of words (or entropy) should be required."*. These aspects clearly highlight on a gap in the current literature.

[6] https://en.wikipedia.org/wiki/Mad_Libs

[7] https://github.com/akwizgran/basic-english

[8] https://tools.ietf.org/html/draft-birk-pep-trustwords-03

2.2.1 Topic conclusion

In conclusion to this topic, the current research has primarily focused on the research and creation of visual representations. Research for textual fingerprints is fragmented and incomplete with work Juola and Zimmermann [12] and M. Goodrich *et al.*[13] providing meaningful research to build upon in terms of word a sentence based encodings. The fragmentation of this research leaves room for further work into this topic area. Alongside this, findings from the previous sections research shows that human language based encodings provided the best usability and, therefore, should be a target for further research looking to improve upon their security and usability.

2.3 Fingerprint representation attacks

This area of research studies ways to physically execute attacks on fingerprint encoding schemes. This differs from previously examined work due to papers discussing the performance and fallibility of encoding schemes simulated the attack without consideration for how the attack would be performed. Research in this area is scant, with lots of research attention being directed towards the security of Man-in-the-Middle (MITM) attacks and not the encoding schemes themselves.

Research in 2002 by **Konrad Rieck**[15] is the first formalisation of attacks on fingerprint representations. The paper titled *"Fuzzy Fingerprints Attacking Vulnerabilities in the Human Brain"* aimed to look into ways users check hexadecimal encoded OpenSSH fingerprint representations. The author created an elegant way to 'weight' more important chunks of the digest. The bytes furthest to the right and left of the digests provided the highest weight. The weight was the smallest in the centre of the digest. This provides a way to score digests and determine the best partial collisions found. For example with the target fingerprint: `9F23` a partial match `9313` is given a score of 45% even though only two characters were matching. This is due to the weightings.

The paper contains an implementation with a "1.2GHz CPU" being able to obtain 130,000 H/s (With MD5). In comparison to this, a mid-range Intel i5-3320M CPU can today obtain 111,700,000 MD5 H/s. This shows that the results obtained from the paper are significantly outdated. However, even with the low hash rate, the author was able to obtain some promising results. Figure 1 contains the best example used.

```
TARGET: d6:b7:df:31:aa:55:d2:56:9b:32:71:61:24:08:44:87
MATCH:  d6:b7:8f:a6:fa:21:0c:0d:7d:0a:fb:9d:30:90:4a:87
```

Figure 1: Best match obtained after a few minutes of hashing

Overall the paper shows an interesting way to create partial fingerprint matches but is not quantified by any empirical evidence gathered on real world users. This, therefore, highlights on gaps in the coverage of this literature.

The only other relevant research on this topic is the work by **M Shirvanian *et al.***[16] and their paper *"Wiretapping via Mimicry: Short Voice Imitation Man-in-the-Middle Attacks on Crypto Phones"*. Further research in the area of "human voice impersonation" has received lots of attention [17][18][19]. This paper was chosen over other alternatives due to is specific use of encoding schemes in its evaluation.

In this paper, the authors develop a way to impersonate users when authenticating Short-authentication-Strings (SAS) in pairing of Crypto-phones. To achieve this impersonating they propose two methods: "Short voice reordering attack" where an arbitrary SAS string is recreated by re-ordering snipped obtained from eavesdropping a previous connection and "Short voice morphing attacks" whereby the use of previously eavesdropped audio snippets the attacker can morph their own voice to match that of the victim. With these methods, they aimed to attack encodings of Numbers, PGP word list (previously discussed work by Juola and Zimmermann [12]) and MadLib (M. Goodrich *et al.*[13] work also previously discussed). The effectiveness of these attacks were evaluated with a study involving 30 participants.

Results from the paper show the effectiveness of these methods. Compared to the baseline of the attacker's voice replacing the victim where this performed with a ∼18% success rate. Morphing gained an overall success rate of 50.58% and Reordering a very impressive 78.23% success rate. Showing that these attacks provide an improvement on top of the naive implementation.

One of the biggest limitations addressed by the authors was the reduction in success rates as the size of the authentication string grew. The morphing and reordering attacks become increasingly ineffective as the user has more time to detect imperfections. This is not quantified by the author and the extent of this degradation is never empirically discussed. Therefore, the results from this study are only effective and applicable in a SAS context.

2.3.1 Topic Conclusion

Overall the literature for this subtopic remains sparse and incomplete. Further suggested work could look into the feasibility of generating partial collisions for all textual representations alongside quantified effectiveness on users. With the possibility to concentrate on a few selected implementations. The work would aim to focus on the various physical methods used and their feasibility. This is one area the previous literature has failed to cover and has only theoretically quantified attacker strength without consideration for the actual real-world cost of these attacks.

3 Overall Summary

TODO: Create an overall summary of all the gaps identified

4 Research Questions

TODO: - Backup choice of questions up using my previous discussion.

References

[1] G. A. Miller, "The magical number seven, plus or minus two: Some limits on our capacity for processing information." *Psychological review*, vol. 63, no. 2, p. 81, 1956.

[2] S. Dechand, D. Schürmann, K. Busse, Y. Acar, S. Fahl, and M. Smith, "An empirical study of textual key-fingerprint representations," in *25th {USENIX} Security Symposium ({USENIX} Security 16)*, 2016, pp. 193–208.

[3] J. Tan, L. Bauer, J. Bonneau, L. F. Cranor, J. Thomas, and B. Ur, "Can unicorns help users compare crypto key fingerprints?" in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems.* ACM, 2017, pp. 3787–3798.

[4] R. Kainda, I. Flechais, and A. Roscoe, "Usability and security of out-of-band channels in secure device pairing protocols," in *Proceedings of the 5th Symposium on Usable Privacy and Security.* ACM, 2009, p. 11.

[5] H.-C. Hsiao, Y.-H. Lin, A. Studer, C. Studer, K.-H. Wang, H. Kikuchi, A. Perrig, H.-M. Sun, and B.-Y. Yang, "A study of user-friendly hash comparison schemes," in *2009 Annual Computer Security Applications Conference.* IEEE, 2009, pp. 105–114.

[6] A. Perrig and D. Song, "Hash visualization: A new technique to improve real-world security," in *International Workshop on Cryptographic Techniques and E-Commerce*, 1999, pp. 131–138.

[7] C. Ellison and S. Dohrmann, "Public-key support for group collaboration," *ACM Transactions on Information and System Security (TISSEC)*, vol. 6, no. 4, pp. 547–565, 2003.

[8] Y.-H. Lin, A. Studer, Y.-H. Chen, H.-C. Hsiao, L.-H. Kuo, J. M. McCune, K.-H. Wang, M. Krohn, A. Perrig, B.-Y. Yang *et al.*, "Spate: small-group pki-less authenticated trust establishment," *IEEE Transactions on Mobile Computing*, vol. 9, no. 12, pp. 1666–1681, 2010.

[9] M. M. Olembo, T. Kilian, S. Stockhardt, A. Hülsing, and M. Volkamer, "Developing and testing a visual hash scheme." in *EISMC*, 2013, pp. 91–100.

[10] D. Loss, T. Limmer, and A. von Gernler, "The drunken bishop: An analysis of the openssh fingerprint visualization algorithm," 2009.

[11] A. Karole and N. Saxena, "Improving the robustness of wireless device pairing using hyphen-delimited numeric comparison," in *2009 International Conference on Network-Based Information Systems.* IEEE, 2009, pp. 273–278.

[12] P. Juola, “Whole-word phonetic distances and the pgpfone alphabet,” in *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP’96*, vol. 1. IEEE, 1996, pp. 98–101.

[13] M. T. Goodrich, M. Sirivianos, J. Solis, G. Tsudik, and E. Uzun, “Loud and clear: Human-verifiable authentication based on audio,” in *26th IEEE International Conference on Distributed Computing Systems (ICDCS’06)*. IEEE, 2006, pp. 10–10.

[14] N. Haller, “The s/key one-time password system,” 1995.

[15] K. Rieck, “Fuzzy fingerprints attacking vulnerabilities in the human brain,” *Online publication at http://freeworld. thc. org/papers/ffp. pdf*, 2002.

[16] M. Shirvanian and N. Saxena, “Wiretapping via mimicry: Short voice imitation man-in-the-middle attacks on crypto phones,” in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014, pp. 868–879.

[17] D. Mukhopadhyay, M. Shirvanian, and N. Saxena, “All your voices are belong to us: Stealing voices to fool humans and machines,” in *European Symposium on Research in Computer Security*. Springer, 2015, pp. 599–621.

[18] S. Chen, K. Ren, S. Piao, C. Wang, Q. Wang, J. Weng, L. Su, and A. Mohaisen, “You can hear but you cannot steal: Defending against voice impersonation attacks on smartphones,” in *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2017, pp. 183–195.

[19] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, “Spoofing and countermeasures for speaker verification: A survey,” *speech communication*, vol. 66, pp. 130–153, 2015.