Parameterizing Fingerprints to Protect Against "Sniff and Suppress" Attacks

Marcus Aqui
Will Pocklington
Advisor: Dr. Leiss

Final Presentation
August 9th, 2019

UNIVERSITY of HOUSTON

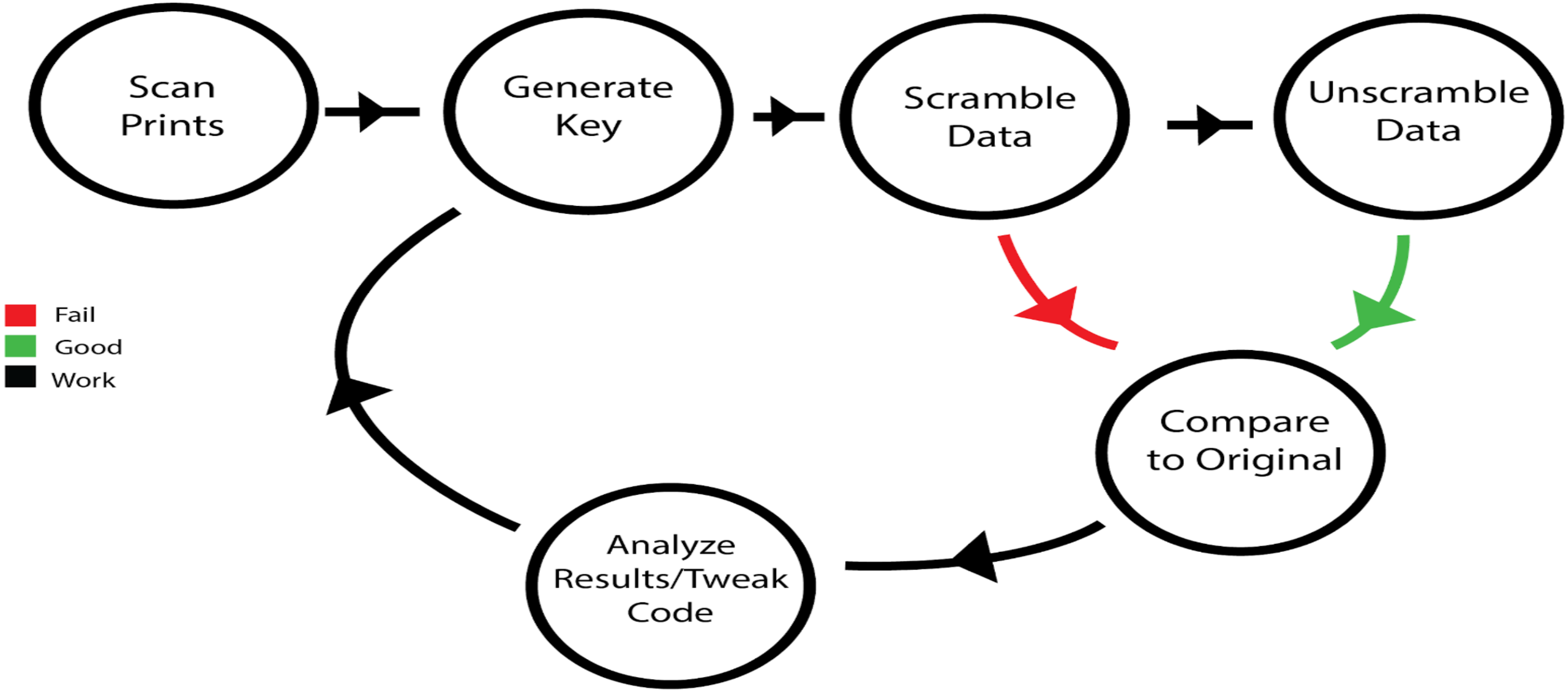
Goal

Prototype a method of protecting fingerprint data in transit to the verification server from malicious actors. By protecting the fingerprint data with a key, any intercepted data will be useless without the original key.

Objectives

1. Determine fingerprint attributes
2. Create a method to scramble fingerprint data
3. Develop prototype software that utilizes the method

Methodology Overview



Objective 1: Tasks

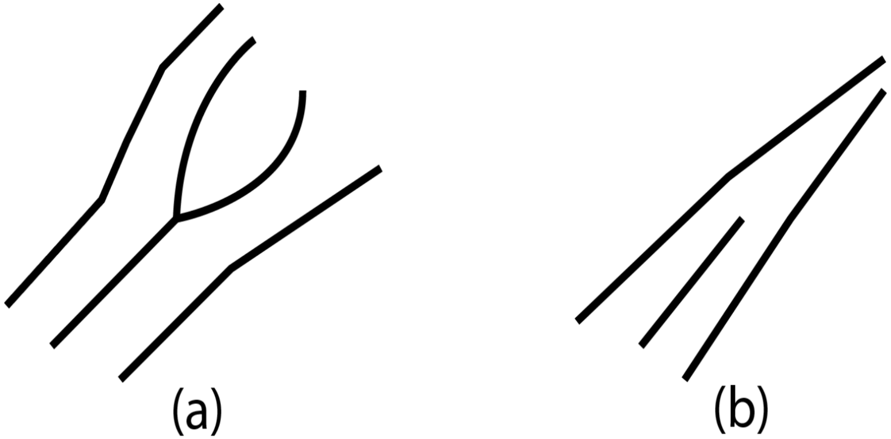
- Learn about fingerprints
- Interpret software documentation

Objective 1: Methodology

- Find documentation
- Generated intermediate files from fingerprint data

Objective 1: Results

- Fingerprint minutiae - two types, measured differently
- Minutiae attributes - spatial coordinates, angle direction, quality
- Derived attributes - fingerprint size, spatial coordinate range

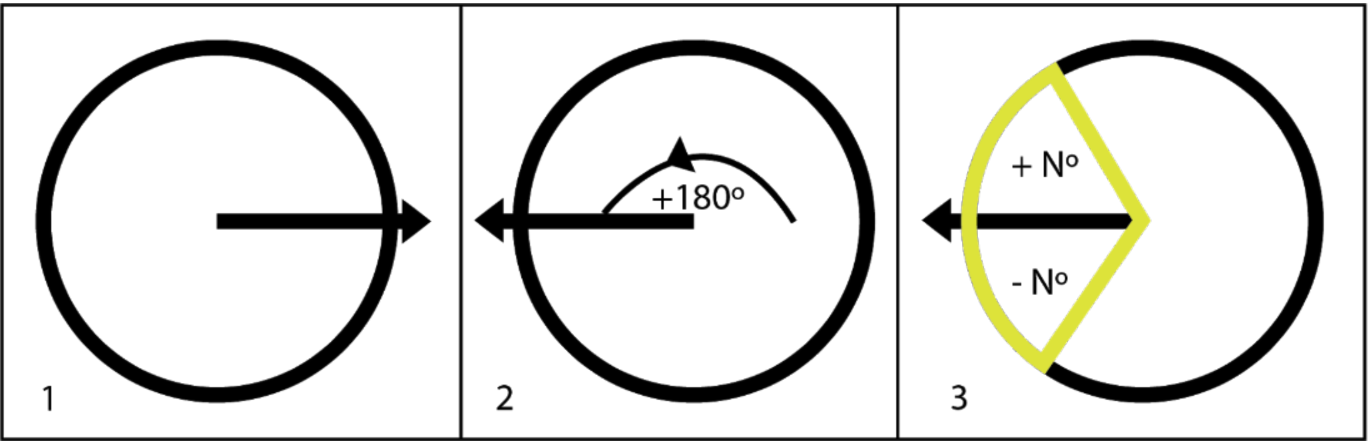


Objective 2: Tasks

- Create a reversible formula for scrambling/unscrambling data

Objective 2: Methodology

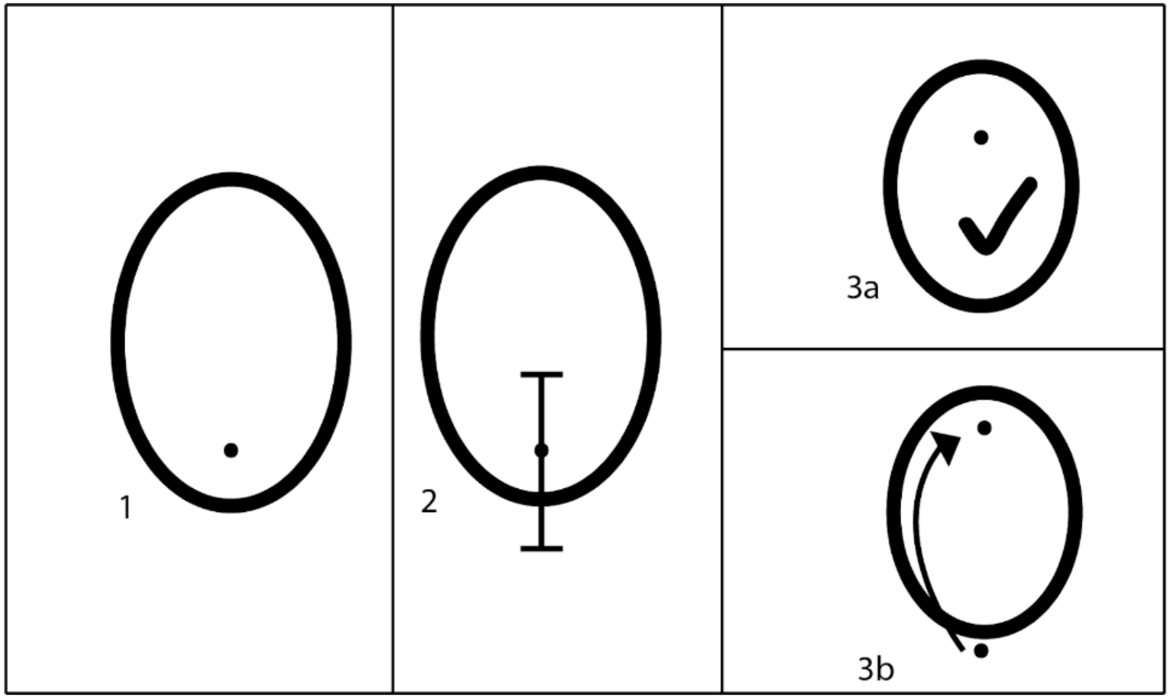
Theta Function



1. Fingerprint angle is found
2. To switch the type of minutia, rotate the angle 180 degrees
3. Based on the key's attributes, add some variance to the angle

Objective 2: Methodology

Coordinate Function



1. Minutia is selected
2. New minutia point is calculated using the formula. It will fall somewhere in a new range determined by the key
3a. If a new coordinate is inside the fingerprint bounds, it is the new coordinate
3b. If it is outside the bounds, it is replaced by an equivalent point inside the bounds. This new point is determined by taking the extrema of the fingerprint as a circular range

Objective 2: Results

- Created a method to scramble/unscramble fingerprint data

Original

X	Y	Angle	Quality
46	280	56	14
51	290	25	33
55	321	214	15
56	352	34	13
…			

Scramble →

Scrambled

X	Y	Angle	Quality
46	236	292	14
265	290	77	33
55	138	34	15
53	352	225	13
…			

Unscramble →

Unscrambled

X	Y	Angle	Quality
46	280	56	14
51	290	25	33
55	321	214	15
56	352	34	13
…			

Objective 3: Tasks

- Write program that calls the scrambling method
- Create testing method to check validity of the parameterization and scrambling

Objective 3: Methodology

- Learned Bash scripting to call executables as needed
- Tests - Original v Scrambled, Original v Unscrambled

Objective 3: Results

- Developed a code suite that can call Bash scripts and receive command line input
- Tested 8000 fingerprints with 17 different keys

Deliverables

1. C++ source code and an executable that scrambles the fingerprint data
2. Various Bash scripts, including:
   a. massScrambler
   b. fingerprintMatcher
   c. prototypeAllInOne.sh:

Limitations

- Tests - only 17 keys tested thus far, due to time needed to develop testing program
- Identification vs Authentication

Future Work

- Test more keys and fingerprints
- Publish a paper

Conclusions

- Scrambling/unscrambling data with keys is successful
- Scrambled prints don’t match original prints
- Unscrambled prints match original prints
- If a fingerprint is compromised by a sniff and suppress attack, a different key can be used, rendering the compromised data useless

References

A. Sankaran, M. Vatsa and R. Singh, Multisensor Optical and Latent Fingerprint Database, IEEE Access, vol.3, no., pp. 653 -665, 2015.

A. Sankaran, M. Vatsa and R. Singh, Latent Fingerprint Matching: A Survey, IEEE Access, vol.2, pp.982-1004, 2014.

Ernst L. Leiss: Safeguarding the Transmission of Biometric Measurements Used for Authenticating Individuals, Proc. IFIP World Computing Congress, NetCon, Santiago, Chile, August 20-25, 2006.

Ernst L. Leiss: Requirements for the Safe Transmission of Biometric Measurements for Authenticating Individuals, CLEI 2008 – Conferencia Latinoamérica de Informática, Sept. 1-5, 2008, Santa Fe, Argentina.

Acknowledgements

The REU project is sponsored by NSF under award NSF-1659755. Special thanks to the following UH offices for providing financial support to the project: Department of Computer Science; College of Natural Sciences and Mathematics; Dean of Graduate and Professional Studies; VP for Research; and the Provost's Office. The views and conclusions contained in this presentation are those of the author and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the sponsors.

Theta Function

Change minutia type: 180 degree rotation of original theta

New Theta = (Rotated Original +- Addon) % 360, where addon is:

$(-1^{Kb + N1} * R * Ks) \% Ms$

Kb = Bit in key position used for this minutia

N1 = Number of 1s in the key

R = Random Number

Ms = Max Shift, sets a limit to addon

Coordinate Function

New Coordinate = Original +- Addon, where Addon is:

$-1^{Kb + minOther} * R$

Kb = Bit in key position used for this minutia

minOther = smallest value of opposite coordinate (x if y is changed, y if x is changed)

R = Random Number

If the new coordinate is outside the range determined by minimum and maximum coordinate values, use circular bounds to get equivalent value within those bounds

Note this function is called only if the original coordinate is not the largest or smallest coordinate for that type

Fingerprint Software and Database

- NIST Biometric Image Software (NBIS)
- MINDTCT
- Bozorth3
- Multi-sensor Optical and Latent Fingerprint Database (MOLF) from the Indraprastha Institute of Information Technology, Delhi