Non-malleable Digital Lockers for Efficiently Sampleable Distributions

PI: Benjamin Fuller, University of Connecticut

Peter Fenteany, University of Connecticut

Link to project: https://eprint.iacr.org/2018/957

Digital lockers: cryptographic primitive that on construction takes in input key and output value and on use performs as so:

$$L_{val,key}(key') = \begin{cases} val & key' = key \\ \perp & otherwise \end{cases}$$

Point function: a digital locker with a single-bit output (i.e., 0 or 1)

Non-malleable: tamper resistance, can be defined for class of functions or generally for error detection/resistance

Previous results in non-malleable point functions rely on unstable assumptions ([KY18], pictured below) or are not composable ([BMZ19]).

$$O(x; r) = (r, r^{g^{h(x)}})$$

(where $h(x) = x + x^2 + x^3 + x^4$)

Both of these prevent construction of non-malleable digital lockers.

We construct a self-composable non-malleable point function, which allows us to construct non-malleable digital lockers

- Biometric authentication
- Password storage

Few key ideas:

- Obfuscating the bit or symbol (from log sized alphabet)
- Use of other cryptographic primitives (non-malleable codes, seed dependent condensers)

One construction given below

$\mathsf{lock}(\mathsf{val}, \mathsf{key})$, input in $\{0,1\}^{\lambda+k}$:

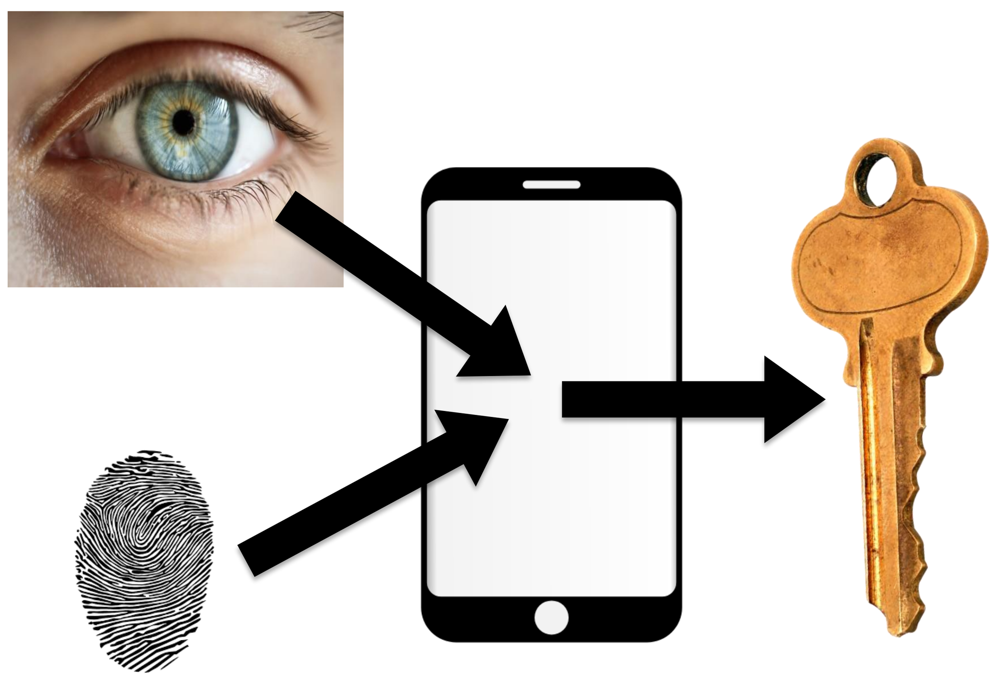
1. Compute $y = \mathsf{cond}(\mathsf{val}, \mathsf{seed})$.
2. Compute $z = \mathsf{Enc}(\mathsf{key}||y)$.
3. Initialize $\mathtt{Out} = \perp$.
4. For $i = 1$ to n compute:
   (a) Sample random generator $r_i \leftarrow \mathbb{G}_{5\lambda}$.
   (b) Compute
   $$\gamma_i = (2\mathsf{val} + z_i)^4 + (2\mathsf{val} + z_i)^3 + (2\mathsf{val} + z_i)^2 + (2\mathsf{val} + z_i).$$
   (c) Append $\mathtt{Out} = \mathtt{Out}||\ (r_i, (r_i)^{g^{\gamma_i}})$.
5. Output $\mathtt{Out}$.

$\mathsf{unlock}(\mathsf{val})$, input in $\{0,1\}^{\lambda}$:

1. Compute $y = \mathsf{cond}(\mathsf{val}, \mathsf{seed})$.
2. For $i = 1$ to n, input r_i, y_i compute:
   $$\gamma_{i,0} = \sum_{j=1}^{4} (2\mathsf{val})^j$$
   $$\gamma_{i,1} = \sum_{j=1}^{4} (2\mathsf{val} + 1)^j$$
   $$P(x, 0, i) = \left(r_i^{g^{\gamma_{i,0}}} \stackrel{?}{=} y_i.\right),$$
   $$P(x, 1, i) = \left(r_i^{g^{\gamma_{i,1}}} \stackrel{?}{=} y_i.\right)$$
   (a) If $P(x, b, i)$ outputs 1, set $z_i = b$. Otherwise output $\perp$.
3. Run decode $\mathsf{key}' = \mathsf{Dec}(z)$.
4. If $\mathsf{key}'_{k\ldots k+n} \neq y$ output $\perp$. Else output $\mathsf{key}'_{0\ldots k-1}$.

Goals and Impact: A more private way to perform iris scan authentication

Currently, devices store biometric data of users in full on device.



Creating and composing non-malleable digital lockers yields biometric authentication from only small subsets of the whole.

The 4th NSF Secure and Trustworthy Cyberspace Principal Investigator Meeting (2019 SaTC PI Meeting)
October 28-29, 2019 | Alexandria, Virginia