



ELTE

EÖTVÖS LORÁND
UNIVERSITY

TKP2021-NVA-29 presentation - The 3PC protocol in a nutshell

Norbert Tihanyi, PhD

OSEP, OSCE, OSCP, OSWP, CRTP, CEH, ECES, ISO27001 LA

✉ ntihanyi@inf.elte.hu

🐦 @TihanyiNorbert

Project no. TKP2021-NVA-29 has been implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund, financed under the TKP2021-NVA funding scheme.

The 3PC protocol

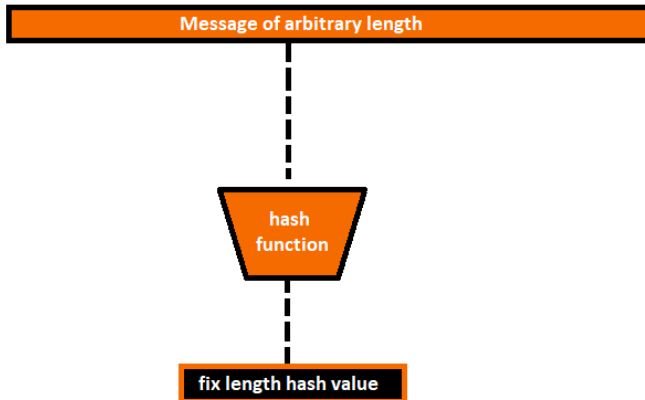
3PC (The 3PC Protocol)

- **Journal:** *IEEE Transactions on Information Forensics and Security*
- **Category:** D1 / IF 6.8
- **Publication date:** 14 January 2024
- **Title:** *Privacy-Preserving Password Cracking: How a Third Party Can Crack Our Password Hash Without Learning the Hash Value or the Cleartext*
- **DOI:** 10.1109/TIFS.2024.3356162

What is hash function?

Definition

A **hash function** is a computationally efficient function mapping binary strings of arbitrary length to binary strings of some fixed length, called *hash-values*. $H : \{0, 1\}^* \rightarrow \{0, 1\}^k$



What is Cryptographic hash function?

Definition (Cryptographic Hash Function)

A cryptographic hash function is a mathematical algorithm that maps data of arbitrary size to a bit string of a fixed size (a hash function) which is designed to also be **one-way function**, that is, a function which is **infeasible to invert**.

The input data is often called the **message**, and the output (the hash value or hash) is often called the **message digest** or simply the digest.

Properties of cryptographic hash functions

The ideal cryptographic hash function has four main properties:

Definition (Properties of cryptographic hash functions)

- it is quick to compute the hash value for any given message
- it is infeasible to generate a message from its hash value except by trying all possible messages
- a small change to a message should change the hash value so extensively that the new hash value appears uncorrelated with the old hash value
- it is infeasible to find two different messages with the same hash value

Avalanche effect

Definition (Avalanche effect)

An important and desirable feature of a good hash function is the non-correlation of input and output, or so-called **avalanche effect**, which means that a small change in the input results in a significant change in the output, making it statistically indistinguishable from random.

Example:

- NTLM(**TKP12345**) — > 13BEE93BE76A7D569D05659D912CB512
- MD5(**TKP1234**) — > AF9FF9AB67EADD8C33681346509FB7C3

Privacy-Preserving Password Cracking:

How a Third Party can Crack our Password Hash, Without Learning the Hash Value or the Cleartext

Asbtract

*Using the computational resources of an untrusted third party to crack a password hash, can pose a high number of privacy and security risks. The act of revealing the hash digest, could in itself negatively impact both the data subject who created the password, and the data controller who stores the hash digest. **This paper solves this currently open problem** by presenting a Privacy-Preserving Password Cracking protocol (3PC), that prevents the third party cracking server from learning any useful information about the hash digest or the recovered cleartext.*

Main contributions of the research

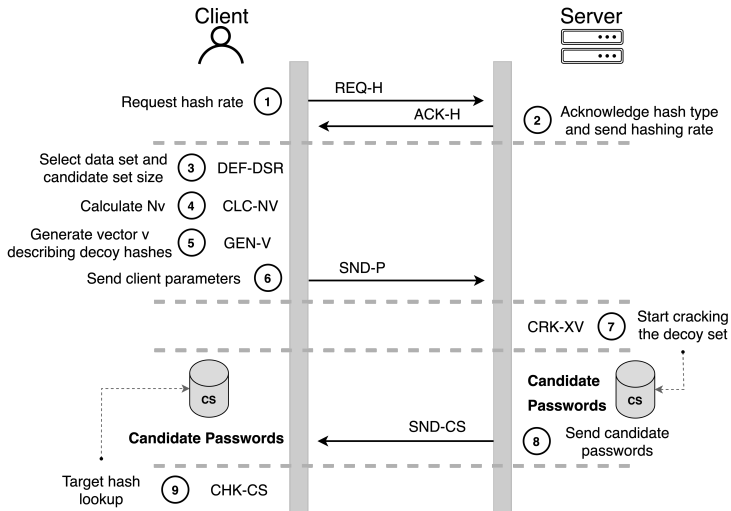
- 1 We introduce the idea of a *Privacy-Preserving Password Cracking* (3PC) protocol
- 2 We extend the concept of *predicate functions* to evaluate decoy hash digests that make up our anonymity set, thereby resolving data transfer and performance issues
- 3 We show that the *probabilistic information* the *cracking server* learns is not practically useful, and the protocol is resistant against attacks and *foul play*
- 4 The protocol provides *plausible deniability*, where the client can claim to have aimed for a different target
- 5 Demonstrations of the implemented protocol, both with “*toy examples*”, and realistic use cases, showing the scalability and efficiency of the protocol
- 6 The protocol ensures the client, using *proof of work*, that the server exhausted the agreed search space

The 3PC algorithm pseudocode

Algorithm 3PC protocol Client-Server exchange

```
1: procedure 3PC( $t, r, \mathcal{DS}$ )
2:   REQ-H( $h$ )                                ▷ Request hash information
3:    $i \leftarrow \text{ACK-H}(h)$                     ▷ Acknowledge hash info request
4:    $\mathcal{DS}, r \leftarrow \text{DEF-DSR}(i)$           ▷ Define data set and  $r$ 
5:    $N_v \leftarrow \text{CLC-NV}(r, |\mathcal{DS}|)$           ▷ Calculate  $N_v$ 
6:    $v \leftarrow \text{GEN-V}(t, N_v)$               ▷ Generate decoy hashes
7:   SND-P( $\mathcal{DS}, v$ )                          ▷ Send parameters to server
8:    $\mathcal{CS} \leftarrow \text{CRK-XV}(v, \mathcal{DS})$         ▷ Cracking decoy set
9:   SND-CS( $\mathcal{CS}$ )                             ▷ Send candidate passwords
10:  CHK-CS( $\mathcal{CS}, t$ )                          ▷ Check target hash in candidate set
11: end procedure
```

The 3PC algorithm



The predicate function

Definition (Predicate function): The set of all strings of length l over the alphabet Σ is denoted by Σ^l . Let Σ be the well-ordered set of hexadecimal characters. For a vector $v = (v_1, \dots, v_{2l}) \in \Sigma^{2l}$ we define a predicate function P_v over Σ^l as follows: Given an input vector $x = (x_1, \dots, x_l) \in \Sigma^l$, $P_v(x) = 1$ if $(v_{2i-1} \leq x_i \leq v_{2i})$ for all $1 \leq i \leq l$, $P_v(x) = 0$ otherwise.

Example

For example if $x = (3, C) \in \Sigma^2$ and $v = (2, 5, C, D) \in \Sigma^4$, we have $P_v(x) = 1$ since $2 \leq x_1 \leq 5$ and $C \leq x_2 \leq D$ are satisfied for our vector v . For a given $v \in \Sigma^{2I}$, let $\mathcal{X}_v$ be the set of all x vectors which satisfy $P_v(x) = 1$. The number of elements (cardinality) of $\mathcal{X}_v$ is denoted by $|\mathcal{X}_v|$. The size of $\mathcal{X}_v$ can be calculated from the definition of v by the following formula:

$$|\mathcal{X}_v| = \prod_{i=1}^I \max\{v_{2i} - v_{2i-1} + 1, 0\} \quad (1)$$

Toy Example : CRC-32

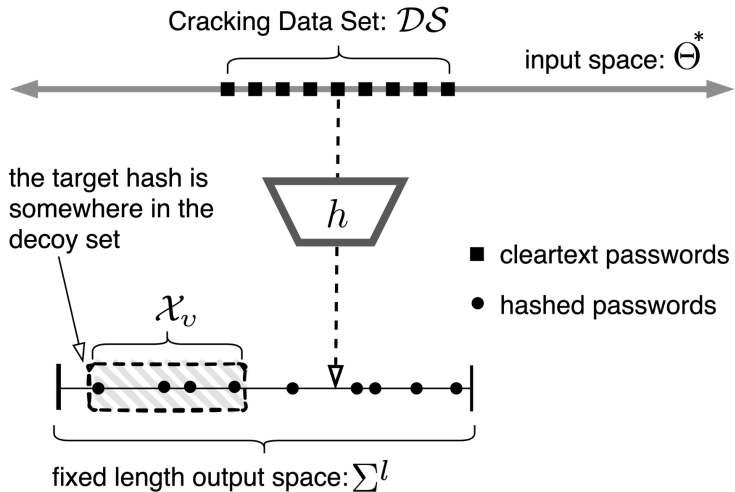
- We have a target hash $t = (C, 6, B, F, A, B, A, 2) \in \Sigma^8$
- The dataset is the Rock-You database,, i.e., $|\mathcal{DS}| = 14\,344\,391$,
- Security parameter, $r = 20$

$$N_v = \frac{r|\Sigma'|}{|\mathcal{DS}|} = \frac{20 \cdot 16^8}{14\,344\,391} \quad (2)$$

The result is $N_v \approx 5988.36$.

We need to construct a vector v where $P_v(t) = 1$ and $|\mathcal{X}_v|$ is around this size. Using the GEN-V algorithm a suitable v can be selected:
 $(C, F, 2, 6, A, B, D, F, 9, F, B, B, A, A, 0, 6) \in \Sigma^{16}$.

The 3PC algorithm



CRC-32 example

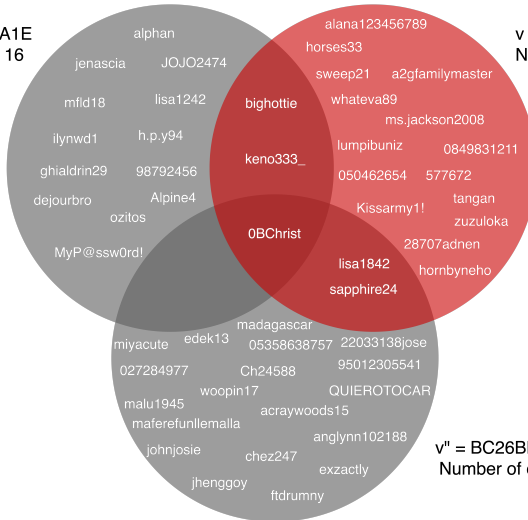
Table: Toy Example 1: Generated candidate hashes

#	password	CRC-32	#	password	CRC-32
1	tangan	C5AEFBA5	11	0849831211	D4BFDBA6
2	hornbyneho	C3AEFBA0	12	lumpibuniz	E3ADEBA4
3	28707adnen	C4AE9BA4	13	sweep21	E3BEEBA6
4	lisa1842	C4BECBA2	14	577672	E3BEFBA5
5	OBChrist	C6BFABA2	15	050462654	E5BDBBA1
6	sapphire24	C6BFDBA2	16	horses33	F2BEBBA0
7	Kissarmy1!	D2ADFBA6	17	zuzuloka	F3AECBA1
8	whateva89	D2BE9BA3	18	ms.jackson2008	F3BEDBA5
9	keno333_	D3BDABA3	19	a2gfamilymaster	F5BDDBA5
10	bighottie	D4AEABA2	20	alana123456789	F6ADABA2

Toy Example

$v' = \text{CD26AB9FACBBAA1E}$

Number of candidates: 16



$v = \text{CF26ABDF9FBBAA06}$

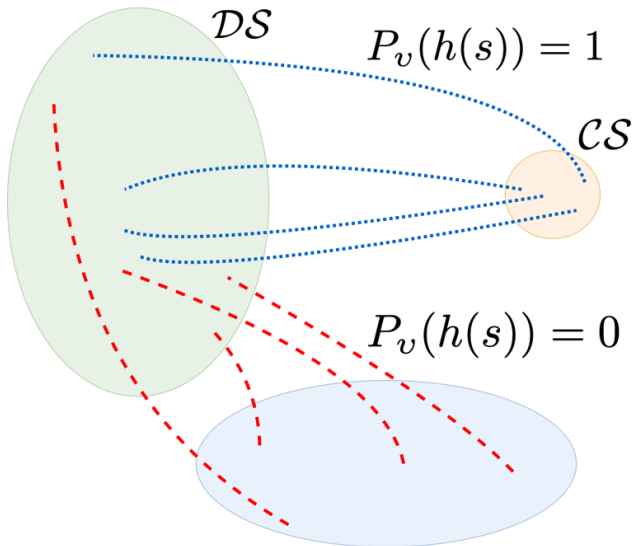
Number of candidates: 20

$v'' = \text{BC26BBDF9FAB9F22}$

Number of candidates: 22

Figure: An example on how a different vector v defines a different random subset of $\mathcal{DS}$

The 3PC algorithm



Real example on RIVYERA

```
3PC — -bash — 42x25
Result found: "5kQC8szzz" (length: 9)
              0x356b514338737a7a7a
    Related target: 8ac50000000000000000000000000000
0000004c/c001/3
| Speed: 190.26G/s | Time elapsed: 19.6h
| Time remaining: ~1s | Freq: 150.00MHz |
Temp: 45°C
| Current ASCII: "00000zzzz" (length: 9)
| Current HEX:   0x30303030307a7a7a7a
Finished after 19.6h with about 191.98G/s.
test$ cat IEEE_example.txt | wc
806834341 806834341 8068343410
test$ cat IEEE_example.txt | tail -n 10
Result found: "00aZytzzz" (length: 9)
Result found: "00aZytzzz" (length: 9)
Result found: "Qg3lyvzzz" (length: 9)
Result found: "wiRnypzzz" (length: 9)
Result found: "XLzuyyzzz" (length: 9)
Result found: "Slkyyqzzz" (length: 9)
Result found: "H000zozzz" (length: 9)
Result found: "7mx3zmzzz" (length: 9)
Result found: "EZx3zxzzz" (length: 9)
Result found: "q4T6zqzzz" (length: 9)
test$
```

Real example

Mask: 8AC500000000000000000000000000004C.

Number of candidates: **806 million**

- **WnbBATman** : This password can be interpreted as the famous superhero of the Warner Brothers. "WnB Batman".
- **MRloNDon6**: Starts with a Title, "MR", followed by a dictionary word, and ends with a number. "Mr London 6" can also be mistaken as human generated.
- **PaYpaLQSC**: What about a password used by employees from the Paypal Quality Service Center?
- **goOgLEFOG**: This can be interpreted as the Google Cloud Platform API solution "Google Fog".

Thank you for your attention!



ntihanyi@inf.elte.hu



@TihanyiNorbert