**System Walkthrough Doc**

**(*ITS/168/17-CityU*)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Description** |
| 0.1 | 10-Sep-2018 | Helei Cui | Initial draft |

## 

## 

Index

PART I -- Private chain design

[1.](#_bit7idv2ancx) Deploy the contract

2. Store index onto the chain

[3.](#_9mc75gzfq3m3) Search via the search token

PART II -- Public chain design

[1.](#_bit7idv2ancx) Client generates file index

2. Service peer confirms index digest

[3.](#_9mc75gzfq3m3) Client generates search token

4. Service peer performs search operation

## 

PART I

Scripts

## 1. Deploy the contract

Before the contract deployment, make sure the raw contract written in Solidity is compiled with truffle. Also, set up a private blockchain and let rpc listen on 127.0.0.1:8545

Now move to the API folder.

## 1-deploy.sh

**Log result**

Using network 'development'.

Running migration: 1\_initial\_migration.js

Deploying Migrations...

... 0xaffdb05468acc7e186e1a5ada42317d9af7c81bfb3a376749c23a27014a2421c

Migrations: 0x4912e9aa8d0e6d301fd5ece5b704ddb12e8ccb4e

Saving successful migration to network...

... 0x4002c30ed70dda5b686eb0769bd5338548ff1e40d62d3ec4dd34666bc2ec07b8

Saving artifacts...

Running migration: 2\_deploy\_contracts.js

Deploying MyContract...

... 0x965f2c6cb06a329e80606a56f5a708e91292ebb7d602bec245b2ed0888eadf75

MyContract: 0xbd80c7a1d279801e9d77f0ab22cf697fca9e044d

Saving successful migration to network...

... 0x8207fc76a45c774f30969cfb67fb4faa1ddfb9111fd2c502a306b3a0003010eb

Saving artifacts...

## 2. Store Index onto the chain

We now build the encrypted index given the directory name containing all the files to be built and store the index onto the chain.

## 2-storeIndex.sh

**Interaction Interface:**

Please input the directory that contains all the files to be built.

> (User type-in)

Note that the files in the directory should be in certain kind of format and the directory lies in ./DSSE/buildIndex/db

**Log result**

Using network 'development'.

this is total number for label.

814

...

Store Successfully

## 3. Search via the search token

**Interaction Interface:**

Please input the keyword to search.

> (User type-in)

Example: UID:8

**Result**

Files that contain the search token open for review.

PART II

Scripts

## 1. Client generates file index

## 1-fileindex.sh

**Interaction Interface:**

Please input the directory that contains all the files to build file index.

> (User type-in)

Note that the files in the directory should be in certain kind of format and the directory lies in ./db

**Log result**

Building add token :)

File index is built! :)

Creating files for file id, file index, index digest, sethash and checklist...

Contract function call (Storeindex) input:

"859a3bef812f77ef8f1cc2e03623a4b7",[3735,575,914,6225,3950,1092,6141,7827,5311,6369,3257,2705,639,5581,3106,6727,6035,3104,2719,2531,1980,1691,117,5827,263,7694,3253,733,6422,2210,5895,4352,2053,3829,5094,2444,2563,7514,3559,5906,6724,3984,2071,4391,598,5651,4297,4073,6544,3712,5886,7057,6472,2353,1649,6021,6660,4579,666,1090,986,5581,6954,331,6818,6850,3167,1562,947,2580,3450,5362,2848,402,6951,5401,6524,7386,2350,1052,7222,4097,7077,2824,5173,5624,6629,4692,3702,2155]

**Generated files**

file id, file index, index digest, sethash and checklist

**File for service peer**

file index

**Files for public chain**

index digest and sethash

## 2. Service peer confirms index digest

## 1-confirm.sh

Suppose the service peer derives the file index from the client such that the service peer can calculate the same index digest and sethash and log them on chain.

**Log result**

Creating files for index digest and sethash ...

Contract function call (Storeindex) input:

"859a3bef812f77ef8f1cc2e03623a4b7",[3735,575,914,6225,3950,1092,6141,7827,5311,6369,3257,2705,639,5581,3106,6727,6035,3104,2719,2531,1980,1691,117,5827,263,7694,3253,733,6422,2210,5895,4352,2053,3829,5094,2444,2563,7514,3559,5906,6724,3984,2071,4391,598,5651,4297,4073,6544,3712,5886,7057,6472,2353,1649,6021,6660,4579,666,1090,986,5581,6954,331,6818,6850,3167,1562,947,2580,3450,5362,2848,402,6951,5401,6524,7386,2350,1052,7222,4097,7077,2824,5173,5624,6629,4692,3702,2155]

**Generated files**

index digest and sethash

**Files for public chain**

index digest and sethash

## 3. Client generates search token

## 2-searchtoken.sh

Suppose the service peer derives the file index from the client such that the service peer can calculate the same index digest and sethash and log them on chain.

**Interaction Interface:**

Please input the keyword to generate the search token.

> (User type-in)

**Log result**

The query keyword is : b'To:'

The search token is: "d6aeaf48e4262b981d782a7312fd6107"

Contract function call (Search) input:

"d6aeaf48e4262b981d782a7312fd6107",0

Creating a file for the search token ...

**Generated files**

searchtoken

**File for service peer**

searchtoken

**Files for public chain**

searchtoken

## 4. Service peer performs search operation

## 2-searchoperation.sh

Suppose the service peer derives the search token from the client such that the service peer can derive the query answer as well as calculate the correct search result digest and log them on chain.

**Log result**

The search token is: d6aeaf48e4262b981d782a7312fd6107

The result it retrieved optimally from search index!

The query answer is : [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90]

The digest is : 7259

Contract function call (Search) input:

"d6aeaf48e4262b981d782a7312fd6107",7259

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Creating files for the search token and search result digest ...

**Generated files**

search result digest

**Files for public chain**

searchtoken and search result digest