KNOW-WHY

XiyuanFeng(Huey) 8591-9234
Yinan Zhang 8781-8979
Jiayu Zhou 6549-2156

Hybrid Cryptography

In cryptography, a hybrid cryptosystem is one which combines the convenience of a public-key cryptosystem with the efficiency of a symmetric-key cryptosystem. A hybrid cryptosystem can be constructed using any two separate cryptosystems:
1. a key encapsulation scheme, which is a public-key cryptosystem
2. a data encapsulation scheme, which is a symmetric-key cryptosystem

In our project, a public private key pair is generated for every physician beforehand. We use symmetric-key cryptosystem(AES) to encrypt the patients' information. And then encrypt the AES key with public key. Then send the encrypted AES key and encrypted information to physican. Then physician can use their private key to decrypt the AES key and then use the AES key to decrypt the ciphertext. The process is as followed:

RSAKeyGen: Randomly generate RSA public key $rsapk$ and RSA private key $rsask$. $RSAGenKey(randompool) \rightarrow rsapk, rsask$. And send private key $rsask$ to physician.

AESencrypt: Randomly generate AES key $aesk\_id, aesk\_data$.
$AESGenKey(randompool) \rightarrow aesk\_id, aesk\_data$. Encrypt information of patients by AES. $AESenrypt(aesk\_id, aesk\_data, patient\_id, patient\_data) \rightarrow ciphertext\_id, ciphertext\_data$

RSAencrypt: When we receive the request of physician, we'll judge their attribute and decide if they have the authority to get all the information or part of them. Then use RSAencrypt those part of informaion.
$RSAencrypt(rsapk, aesk\_data, (aesk\_id)) \rightarrow aesk\_data\_cipher, (aesk\_id\_cipher)$. And then send $(ciphertext\_id), ciphertext\_data, (aesk\_id\_cipher), aesk\_data\_cipher$ to the physician.

RSAdecrypt: When physician receive the information. They can first decrypt the

AES key by their RSA private key. $RSAdecrypt(rsask, (aesk\_id\_cipher), aesk\_data\_cipher) \rightarrow (aesk\_id), aesk\_data.$
AESdecrypt: Then the physician can use the AES key to decrypt.
$AESdecrypt((aesk\_id), aesk\_data, (ciphertext\_id), ciphertext\_data) \rightarrow (patient\_id), patient\_data$

The advantage to use Hybrid Cryptosystem is that compared to only use public, private key strategy, it is faster. But if we only use symmetric encryption then we have to change the symmetric key all the time to prevent leak of information. Then every time we change the symmetric key, we need to let all the physician know. It's not convenient. And in Hybrid Cryptosystem, physician are not necessary to keep the symmetric key. So changing the key will be more convenient.

And we have successfully implemented this part under the Linux environment. And in HC_REDAME.txt shows how to run the implementation of this part.

Pairing Based Cryptography(PBC)

Another research we are doing is the Pairing Based Cryptography. This is widely used in e-health security model. Paring-based Cryptography is the use of a pairing between elements of two cryptographic groups to a third group with a mapping $e : G_1 * G_2 \rightarrow G_T$ to construct or analyze cryptographic. The advantage of using PBC is if it pairings are symmetric then it can be used to reduce a hard problem in one group to a different, usually easier problem in another group.

There are several concept of PBC. First we let $(G1, +)$ and $(G2, +)$ be two additive cyclic groups of (nearly) prime order q with $G1 =< P >$ and $G2 =< Q >, (GT, .)$ be a multiplicative cyclic group of order q with $GT =< g >$. We write as usual 0 for the identity elements of $G1, G2$ and 1 for GT . A pairing or a bilinear map is a map $e : G1 * G2 \rightarrow GT$ satisfying the following properties:

Bilinearity: For all $P_1, P_1 \in G_1, Q_1, Q_1 \in G2, e$ is a group homomorphism in each component, i.e.
1. $e(P_1 + P_1', Q_1) = e(P_1, Q_1) * e(P_1', Q_1),$
2. $e(P_1, Q_1 + Q_1') = e(P_1, Q_1) * e(P_1, Q_1').$

Non-degeneracy:e is non-degenerate in each component, i.e.
1. For all $P_1 \in G_1, P_1 \neq 0$, there is an element $Q_1 \in G_2$ such that $e(P_1, Q_1) \neq 1$,
2. For all $Q_1 \in G_2, Q_1 \neq 0$, there is an element $P_1 \in G_1$ such that $e(P1, Q1) \neq 1$.
Computability: There exists an algorithm which computes the bilinear map e efficiently.

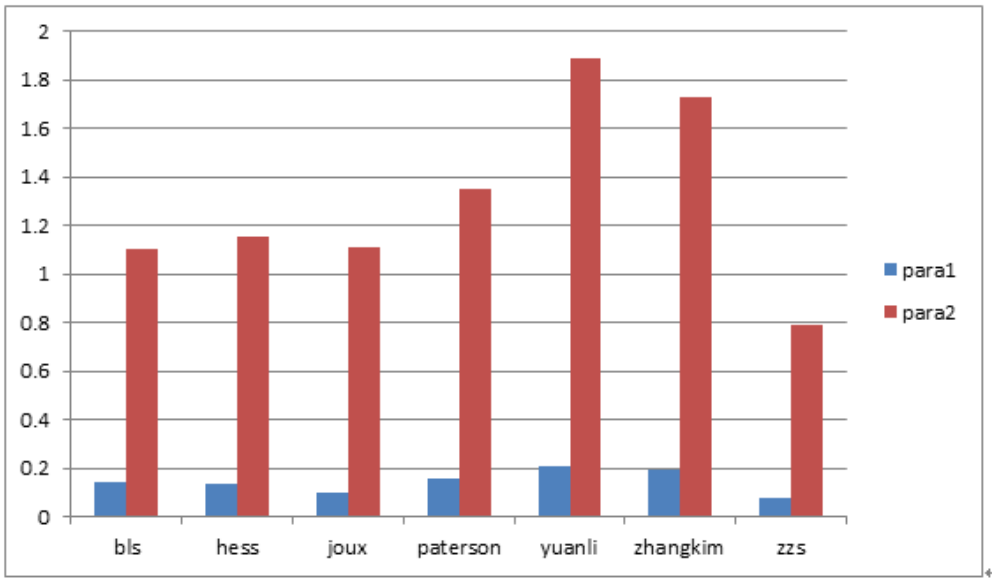
In our experiments, we just use the pbc library and try to run several paring based signature or negotiation keys structure to see how it works. The code of this part are all downloaded from the Internet. And we just analysed them. The structure includes:
1. Boneh-Lynn-Shacham short signatures
2. Hess identity-based signatures
3. Joux tripartite Diffie-Hellman
4. Paterson identity-based signatures
5. Yuan-Li identity-based authenticated key agreement
6. Zhang-Kim identity-based blind/ring signatures
7. Zhang-Safavi-Naini-Susilo signatures

Then we'll give the analysis of Yuan-Li identity-based authenticated key agreement as an example. The code mainly achieve the function as followed:
SETUP: KGS chooses $G_1, G_2, e : G1 * G1 \rightarrow G_2, P, H : \{0,1\}* \rightarrow G1, s, H-$ some function for key calculation. KGS calculates $Ppub = s * P$, publishes $G_1, G_2, e, P, Ppub, H1, H$ and saves s as master key.
EXTRACT: For the user with ID public key can be calculated with $Qid = H1(ID)$. KGS generates bound public key $Sid = s * Qid$.
1. A chooses random a from Z_p*, calculates $Ta = a * P$.
   $A \rightarrow B : Ta$
2. B chooses random b from Z_p*, calculates $Tb = b * P$.
   $B \rightarrow A : Tb$
3. A calculates $h = a * Tb = a * b * P$ and shared secret key $Kab = e(a * Ppub + Sa, Tb + Qb)$
4. B calculates $h = b * Ta = a * b * P$ and shared secret key $Kba = e(Ta + Qa, b * Ppub + Sb)$

Session key is $K = H(A, B, h, Kab)$.

In addition, we input different length of parameter and record the running time

of each structure to see the efficiency of each structure. The result is stored in folder "PBC Running Time" and we get the graph as followed, the y-direction is time(s):

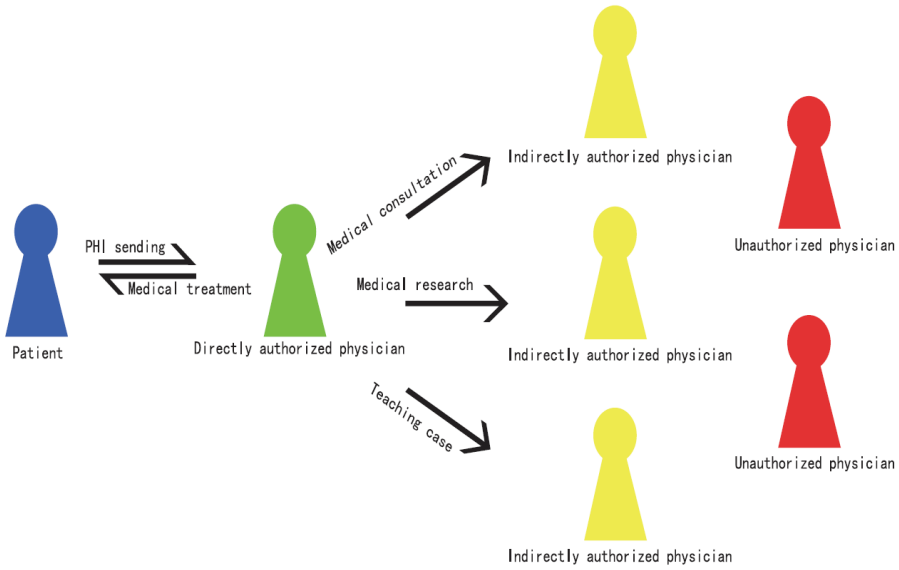


Attribute Based Access Control

In our project, patients' data is required to be encrypted for mutiple receivers who are identified by their roles, so that only those receivers whose attributes satisfy specific policy can perform decryption successfully. In this case, we have three different kinds of people who may access patients' data, primary physicians, secondary physicians and unauthorized ones. An access policy defined over a set of attributes is associated with each encrypted data, and each user in the system has a private key obtained from an authority corresponding to the users attributes.

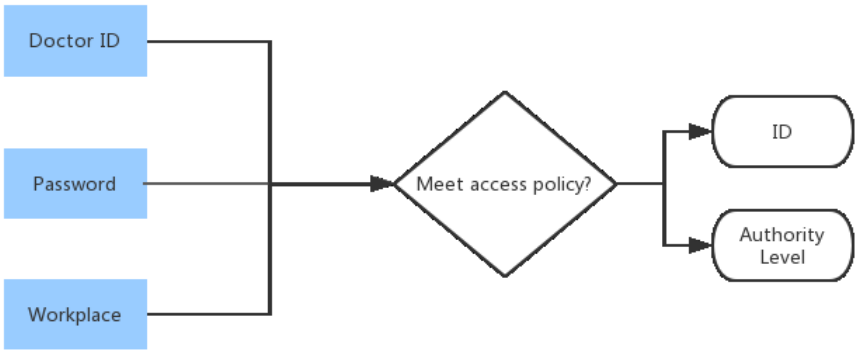
Access Policy

Let $(P1, P2, ..., Pn)$ be a set of parties. A collection $A \subset 2^{(P1,P2,...,Pn)}$ is monotone if $\forall B, C : if B \in A$ and $B \subset C$ then $C \in A$. An access structure (respectively, monotone access structure) is a collection (respectively, monotone collection) A of non-empty subsets of $(P1, P2, ..., Pn), i.e., A \subset 2^{(P1,P2,...,Pn)} \varnothing$.



The sets in A are called the authorized sets, and the sets not in A are called the unauthorized sets.

If a doctor requests to access patient's information, He needs to provide three credentials including doctorID, password and workplace. And then an attribute authority will check out whether he meets the access policy and if yes, what is his authority level which determinds how much data he can access.



Document Digital Watermark

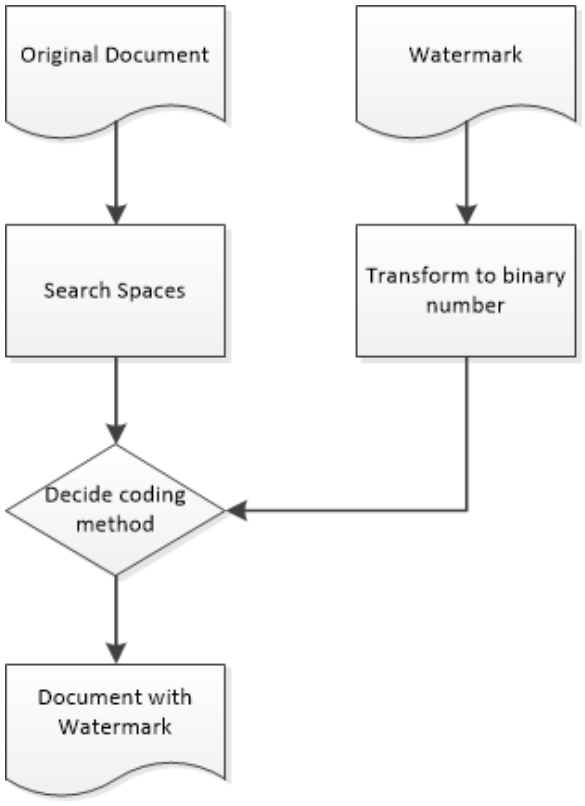
Introduction

Digital watermark is about exploiting the redundancy of file to add extra content into it.The most important attribute of watermark is robustness.Besides, keep-

ing the original content to be distinguishable is also significant.Most watermark algorithms are based on image, audio and video data which have a lot of redundancy to be exploited.Although text is the most popular way to record information, there are little watermark algorithm for text documents due to the lack of redundancy.According to the method of embedding watermark, algorithms are divided into three kinds: format, syntax and semantics.Due to the attribute of English, it's hard to apply syntax methods.So I designed a format-based and a semantics-based watermark algorithm for electronic medical record.

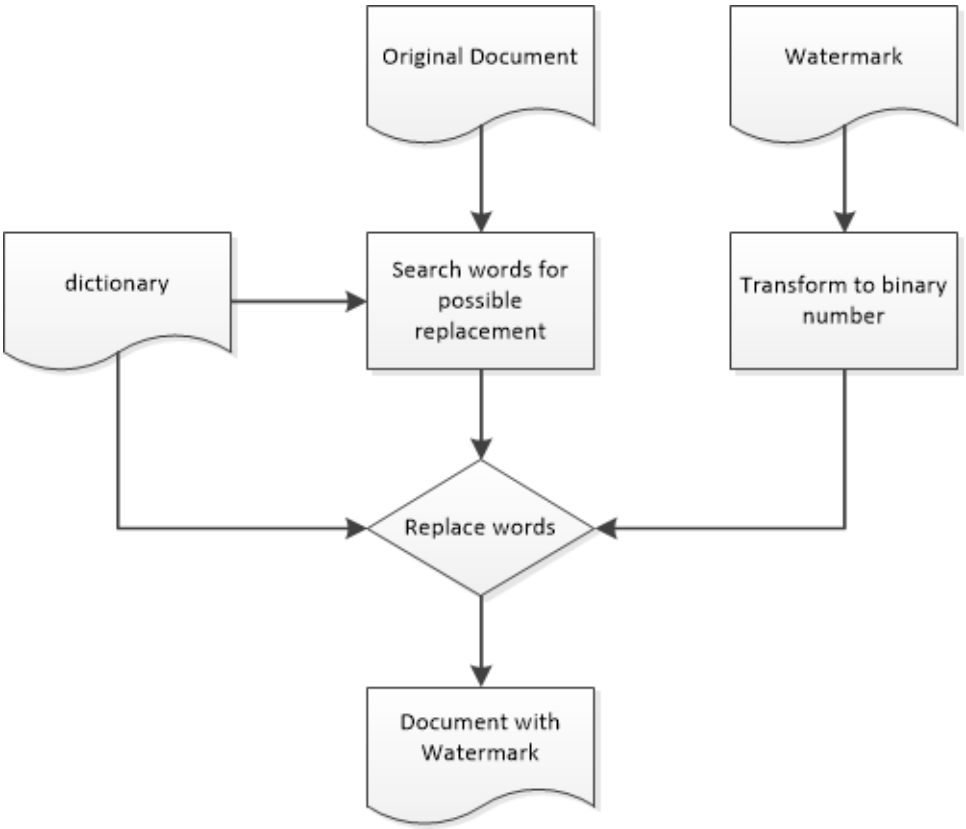
Format

Format-based watermark algorithms use color, size and other attribute of documents to embed the watermark.Since we are using plaint text documents based on ASCII codes, I decides to use the codes which don't represent alphabets to embed watermark. The watermark message is transformed into a binary number.Then each bit of the number will decide whether each space(' ') in document is represents by code 0(null) or code 32(space).



Semantics

Semantic-based watermark algorithms analyze the structure of sentences and try to modify nouns, verbs and other words in sentences to embed the watermark.Because medical records are not like articles.Most sentences in medical records are incomplete while most information are represent by words but not sentences. I design a word-based algorithm to embed the watermark.In this algorithm, a digital dictionary is used to store synonyms and possible represent forms of words. Before embedding the watermark, words in the documents will be analyzed for whether they can be replaced by other words or not.After the watermark message is transformed into a binary number, each bit of the number will decide whether each possible word is replaced or not.



Conclusion

After experiments, the format-based algorithm is fast while not changing the original content.Although the watermark will not be destroy by clipboard cutting and pasting, it can still be easily changed if the coding method is known.The

semantics-based watermark has much stronger robustness since it's impossible to tell which words are replaced without the original document.However, the semantic-based watermark algorithm is slow while changing the original content.A bad dictionary of replacement may change the meaning of the document.