In-game item validity check using digital signature

Christopher Chandrasaputra - 13519074
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13519074@std.stei.itb.ac.id

***Abstract*—With the grows of gaming industry, the number of players keep increasing. With many varieties of player, game developer needs to increase their game security. One of the problems that developers face are illegal in-game items. To solve the problem, a cryptographic system can be implemented, to be precise, digital signatures. This paper will discuss about implementation of digital signature to increase game security, particularly in-game items. Digital signature will be used to check in-game item validity to prevent any illegal in-game items.**

Keywords—game; security; item; validity; signature;

I. INTRODUCTION (*HEADING 1*)

As the gaming industry grows, the number of players continues to increase. This result often benefits game developers since they will be able to promote their game to more potential player and if they are lucky, the player will start to play their game. However, not all players have good intention to enjoy the hard work of game developers. Sometimes, players have ill intention toward some games. This ill intention's usually gives benefits to the player and loss to other player or the developers.

One of the main concerns about game development is game exploitation. Game exploitation is a way to break the game mechanics with the purpose of gaining advantages inside the game without playing as the game designers intended. There are many types of game exploitation. Some of them involve the game character by removing hitboxes or increasing character speed while some other involve gaining something that should have not been able to be obtained at the current phase of the game.

Game exploits not only ruin the game experience, but it can also ruin the game market. In some cases, exploiters may control the game market with in-game items they obtained from exploiting the game. There are also cases where game exploiters use a third-party program to exploit the game vulnerabilities so they may gain useful stuffs for them. By doing this, players who played the game may feel that it is unfair and in result, the game may lose its players.

To prevent game exploitation, game developers can create a certain mechanic to prevent any exploitation from happening. Due to the increases of players in this industry, there will be more people with experiences involving exploiting digital security to found game vulnerabilities. With those things in mind, it will result to a never-ending exploitation prevention and finding game vulnerabilities cycle.

One of the things game developers can do to prevent game exploitation is to implement a cryptographic system in their game. By implementing this system, game developers can check their whether it is account information, redemption code, or even to check in-game items validity.

In this paper, we will be focusing on checking in-game item validity using digital signature. The process will consist of two main parts, signing and validating. Both processes will involve two fields of knowledge in cryptography that is, public-key cryptography and hash.

II. THEORETICAL BASIS

A. Cryptography

Cryptography is a study that mainly focus on security of a data transmission between sender and receiver. The word "cryptography" was taken from Greek language, κρυπτός (*kryptós*) and γράφειν (*gráphein*) which mean secret and write respectively. With both words combined, "cryptography" means secret writing. There are four types of security that cryptography can offer.

The first one is confidentiality. Confidentiality gives a data privacy so that only the sender and receiver know the data being transferred without any third-party knowing the information contained inside the data. This type of security mostly used to prevent any third-party from any sensitive information that is being transferred or stored.

The second one is integrity. Integrity gives the truth of a data so that the data is still whole and remain intact without any modification. This type of security makes sure that any data that is being transferred or stored is whole, it is crucial to make sure that a data is not altered. If a data that is being transmitted is altered by receiver, for example a money transaction that should only send 1 dollar is altered to send 1001 dollar, it will cause trouble not only to the sender, but also to the third-party that provide services for transferring the data.

The third one is authentication. To make sure data is being sent from the right person and received by the right person, authentication makes sure that the data is not being sent or received by a third-party and can only be sent and received by the sender and receiver.

The last one is nonrepudiation. To make sure that any receiver cannot repudiate what they sent. With this, receiver can know who sent data and the person who sent the data cannot claim that they have never sent the data.

The study of cryptography has been around for a long time ago even before computers were invented. In result, cryptography can be split into two, classic cryptography and modern cryptography. Classical cryptographies are cryptographies that manipulates traditional characters and mainly focus on techniques used for encrypting and decrypting data. This, however, does not offer securities such as integrity, authentication, and nonrepudiation. On the other hand, Modern cryptographies utilize computing power. Therefore, it manipulates data bit by bit. Some of the method might uses traditional techniques but it mainly utilizes mathematical algorithms for encryption and decryption. By utilizing mathematical algorithms for its encryption and decryption, it may offer new securities such as integrity, authentication, and nonrepudiation.

There are two main methods used in classical cryptography algorithms. The two methods are the following:

- Substitution

  This method works by replacing a character to another character. For example, for every character in alphabet, the character will be replaced by the character next to it. In result, the encrypted message “FOX” will be “GPY”. To revert the encrypted message back to the original message, simply every character back.

- Transposition

  This method works by rearranging characters position inside the message. For example, the message “THEQUICKBROWNFOX” can be encrypted by swapping every two characters. In result, it becomes “HTQEIUKCRBWOFNXO”. to revert the encrypted message back to the original message, simply swap every two characters back.

B. Symmetric Key Cryptography

Symmetric key cryptography is a modern cryptography that uses key to encrypt and decrypt a data. There only exist one key in symmetric key cryptography for its encryption and decryption. It also manipulates data bit by bit rather than manipulating it character by character. To see how symmetric key cryptography works, you can see the image below.

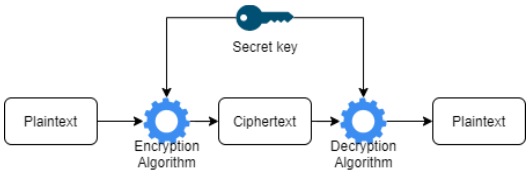

Figure 1. Symmetric Key Cryptography (author illustration)

Here are algorithms that implement symmetric key cryptography:

- Stream Cipher

  This algorithm uses a stream of bits or bytes as key; therefore, it is called stream cipher. The stream usually generated by a generator. The generator is expected to be able to generate a random output so cryptanalysts will find it more difficult to find out the data. To understand it better, you can see the illustration below.

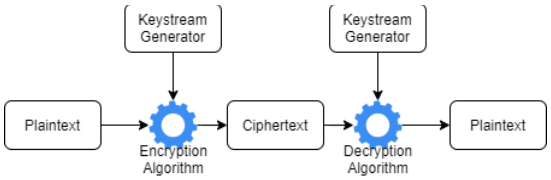

Figure 2. Stream Cipher (author illustration)

- Block Cipher

  Block cipher is another modern cryptography. Unlike stream cipher, block cipher works by grouping many bytes into blocks. Encryption and decryption process will be done to each block grouped from the message. there are a few block cipher algorithm such as Electronic Code Book (ECB), Cipher Block Chaining (CBC), Cipher Feedback (CFB), Output Feedback (OFB) and, Counter Mode.

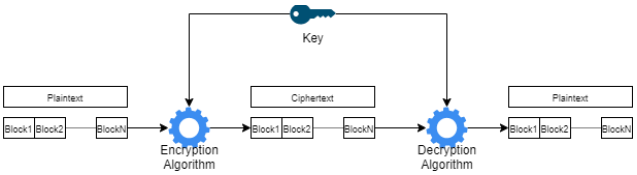

Figure 3. Block Cipher (author illustration)

C. Public-Key Cryptography

Public-key cryptography is a modern cryptography that uses different key to encrypt and decrypt a data. It is possible for the keys to be different because it involves mathematic algorithms.

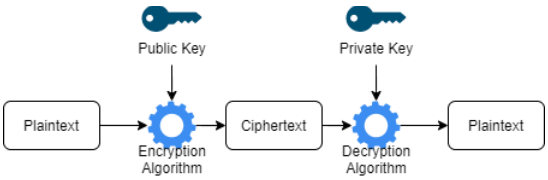

Figure 4. Symmetric Key Cryptography (author illustration)

There are many public-key cryptography algorithms, but for the purpose of this paper, we will only discuss two public-key cryptography, RSA and Elgamal.

- RSA

  RSA is a public-key algorithm invented by Ronal Rivest, Adi Shamir, and Len Adleman. This algorithm is very popular and has a wide range of applications. The algorithm has a few properties:

  1. $p, q;\ private,\ prime\ number$
  2. $n = p \cdot q;\ public$
  3. $\Phi(n) = (p-1)(q-1);\ private$
  4. $e;\ public\ key$

5. $d;\ private\ key$
6. $m;\ plaintext$
7. $c;\ ciphertext$

To generate a pair of keys, the following steps must be done:

1. Pick two prime number p, q.
2. Calculate $n = p \cdot q$
3. Calculate $\Phi(n) = (p-1)(q-1)$
4. Choose an integer e that is relatively prime to Φ(n) for public key
5. Calculate the inverse of e in modulo Φ(n) using the formula $d = e^{-1}\ mod\ (\Phi(n))$

For encryption and decryption, use the following formula:

$$c_i = {p_i}^e mod(n); for\ encryption$$

$$p_i = {c_i}^d mod(n); for\ decryption$$

- Elgamal

This public-key algorithm was invented by Taher Elgamal. The security of the algorithm lies in the difficulty of calculating discrete logarithm. The algorithm has a few properties:

1. $p;\ public,\ prime\ number$
2. $g,\ g < p;\ public,\ random\ number$
3. $x,\ x < p;\ private\ key$
4. $y = g^x\ mod\ p;\ public\ key$
5. $m;\ plaintext$
6. $a, b; ciphertext$

To generate a pair of keys, the following steps must be done:

1. Pick random prime number p.
2. Pick two random number $g, x;\ g < p;\ 1 \leq x \leq p-2$
3. Calculate $y = g^x\ mod\ p$

For encryption, follow these steps:

1. Split plaintext into blocks with the following rules

$$0 < block < p-1$$

2. Pick a random number k with the following rules

$$1 \leq k < p-2$$

3. Encrypt every block using the following formula

$$a = g^k\ mod\ p$$

$$b = y^k m\ mod\ p$$

For decryption, follow these steps:

1. Calculate $(a^x)^{-1} = a^{p-1-x}\ mod\ p$
2. Calculate the plaintext using the following formula

$$m = b/a^x\ mod\ p = b(a^x)^{-1}\ mod\ p$$

D. Hash Function

Hash function is a function that compresses message M with random length to a fixed length. The function returns a hash value or a message-digest. Unlike encrypting and decrypting, hash function result cannot be reverted to its original value.

A hash function should have the following properties:

1. Collision resistance

   Any hash function $H(x)$ for any input a and b should rarely satisfy $H(a) = H(b)$

2. Preimage resistance

   For any output y, finding the value of a that satisfy $H(a) = y$ should be hard

3. Second preimage resistance

   For any input a and output $y = H(a)$, finding the value of b that satisfy $H(b) = y$ should be hard

Due to the properties owned by hash functions, hash functions are mainly used to verify data integrity. By normalizing the length of data, hash functions can be used to reduce transfer load for the purpose of verifying data integrity that is located far away from database. It is also possible to create uniform password lengths inside a database. Some of the known hash functions are SHA-256, SHA-512, MD5, RIPEMD, and WHIRPOOL.

E. Digital Signature

Digital signature is a method to authenticate a digital data. It can also be used to check the integrity of a data. Because of the securities that digital signatures offer, it can be applied to a wide range of fields. There are two options to sign a digital data.

The first option is to encrypt the data using symmetric key cryptography or public-key cryptography. Even though this option is a simple solution, it has a downside to it. Using symmetric key cryptography does not offer non-repudiation security. On the other hand, using public-key cryptography does not offer authentication security.

The second option is combining public-key cryptography with hash functions. By using the combination of public-key cryptography and hash functions, this option gives a better security such as authentication, integrity, and nonrepudiation. The process of signing digital data and verifying a digital signature can be broken down into these steps.

- Signing:
  1. message is hashed using a hash function to obtain its message-digest,
  2. message-digest is encrypted using sender's private key to obtain message's signature,

3. the signature is then appended to the message.

- Verifying:
  1. signature is decrypted using sender's public key to obtain message-digest,
  2. message is hashed using the same hash function used when signing to obtain its message-digest,
  3. both message-digest from hashed message and decrypted signature will be compared to determine the sender's identity and message's integrity.

III. Digital Signature Implementation For In-Game Items

As we have discussed before, digital signatures have a wide range of applications. Game is one of the things where digital signatures can be applied to. In this paper, we will be discussing the implementation of digital signatures on in-game items to detect any illegal in-game item creation. We will also discuss about the pros and cons of using digital signature to verify the validity of in-game items.

Firstly, we need to know how illegal in-game items can be created. Usually there are many ways, depending on the game available vulnerability, to obtain an illegal in-game items. There are methods such as duping item, where a single item is duplicated by abusing the game mechanics. Another methods are a little bit straightforward by using a third-party program to create an item.

The usage of third-party program can be easily done in offline games since offline games usually does not really care if the player's experiences are ruined or not because once the game has been bought by the player, the only person that can control the experiences from playing the game is only the player. However, in an online game, game developers may pay more attention to a third-party programs especially if the game has its own market due to the fact that online games let interaction between players. It is possible to ruin another player experiences inside online games.

Third-party program used in obtaining illegal in-game items works in many ways such as modifying game data, modifying game memory, tampering the game connection, or breaking the request system. One way to deal with illegal in-game items is to add detailed item creation descriptions to in-game items. Thus, game developers can track every item in the game ever created by players. For example, see the image below for item without and with detailed item creation description.

```
{
    "name": "Wooden Sword",
    "description": "Your first sword!",
    "stats": {
        ...
    }
}
```

Figure 5. In-Game Item Information Without Creation Description (author illustration)

```
{
    "id": 218319233415,
    "name": "Wooden Sword",
    "description": "Your first sword!",
    "stats": {
        ...
    }
    "original_owner_id": 25183165931286456123,
    "item_creation_datetime": "2021/12/15
    12:32:30",
    "obtaining_method": "Quest Item"
}
```

Figure 6. In-Game Item Information With Creation Description (author illustration)

From the example, by adding information about in-game item creation, it can be determined the information needed. However, even though adding detailed item creation description might be a good and simple solution, it does not prevent the ability of players to manipulate the item creation description itself. Using the given example, we can try to create a new item by modifying the item description.

```
{
    "id": 218319233416,
    "name": "Best Wooden Sword",
    "description": "Your first sword!",
    "stats": {
        ...
    }
    "original_owner_id": 12345678,
    "item_creation_datetime": "2021/12/19
    00:11:22",
    "obtaining_method": "Microtransaction"
}
```

Figure 7. Modified In-Game Item (author illustration)

To prevent players from manipulating it, we can introduce a cryptographic system into the game by implementing digital signatures on every in-game items. With the existence of signatures on every in-game items, game developer can check whether an item is valid or not. A public-key cryptography will be used for encrypting the item's message-digest. The main purpose using a public-key cryptography rather than using the symmetric key cryptography is to check item validity from both sides, player and the game.

```
{
    "id": 1111,
    "name": "Item Name",
    "description": "Item Description",
    "stats": {
        "strength": 0,
        "dexterity": 0,
        "intelligence": 0
    },
    "original_owner_id": 1010,
    "item_creation_datetime": "2021/12/19 00:00:00",
    "obtaining_method": "Method 1",
    "signature":
    "faa3c8cd84f1721b2ed6e8db0e6cfb797dc184bb4e54c85a
    b7c9ed55232ec4f2cea79ae697fad7a55fd819a6a5fd068d1
    d391b69ed270bc3ac3d2e15cb428a9608572257ae2a672873
    1657d02a7cd4a4"
}
```

Figure 8. In-Game Item Information With Digital Signature Added (author illustration)

The process of creating digital signature will be explained in the following steps:

- Signing

  The process of signing will be done twice using server's private key and player's private key. This double-

signing process is used so that other players can also check the item validity particularly the item's data. owner. Signing process can be broken down into these steps:

1. Player send item creation related request to the game server

   This process is automatically carried out by the game mechanic when the player performs certain actions,

2. Server generate an item

   The game server will create an item based on the received request,

3. Generating item's message-digest

   Item's data converted into string and then hashed using a hash function to generate item's message-digest

4. Server sign the item

   The generated message-digest encrypted using server's private key

5. Server sends the signed item to the player

   Player will receive the server's signed item

6. Player encrypts item's signature value

   The signature value generated using the server's private key is then encrypted using the player's private key as a prove of the item's original ownership

7. Player send back the item data to the server

   The item data sent to the server will be stored for further checking

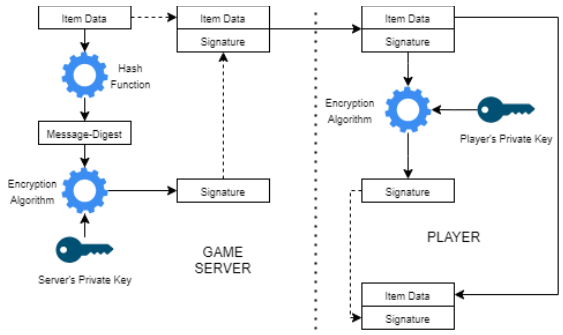

Figure 9. In-Game Item Signing Process (author illustration)

- Verifying

  There are two different ways to verify the item. The first way is verifying the item's and the other one is verifying the item's original owner.

  - Verifying item's original owner

    By comparing the original signature value generated from the server, the original item's owner can be determined.

    1. Obtain item's signature value

       Item's signature that has been encrypted using player's private key is decrypted using player's public key

    2. Calculate server's signature value

       Using the same process from the signing process, raw item data is hashed and then encrypted using server's private key

    3. Match signature values

       The result from step 1 and 2 will be compared, if the result from both steps is the same, then the item's original ownership is verified, otherwise, the item's original ownership has been altered.

    By comparing the original signature value generated from the server, the original item's owner can be determined.

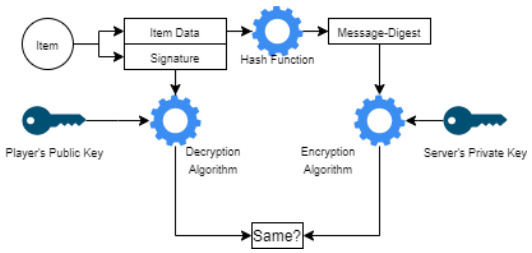

Figure 10. Verifying Item's Original Owner Process (author illustration)

  - Verifying item's data

    Unlike verifying item's original owner, verifying item's data compare the item's message digest with its signature.

    1. Decrypt the encrypted signature value

       Item's signature that has been encrypted using player's private key and server's private key is decrypted using player's public key and server's public key

    2. Obtain item's message-digest

       To obtain the item's message-digest, use the hash function

    3. Match message-digests

       From the message-digest obtained from step 1 and step 2, both of them are compared and if both of the have the same value, the item's validity can be confirmed, otherwise, the item has been tampered

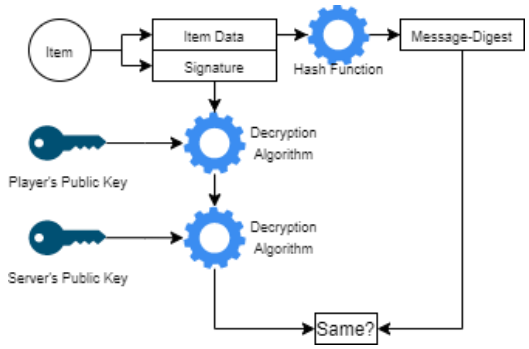

Figure 11. Verifying Item's Data Process (author illustration)

By utilizing digital signature mechanism, there are few in-game item related exploitation that can be avoided. Digital signature allows game developers to keep tracks for all items created inside the game. However, the implementation discussed in this paper does not prevent any connection related exploitation. It mainly focuses on item modification or item duplication prevention.

IV. Conclusion

Cryptography is a very versatile study that can be applied in many applications. Its main purpose is to increase the security of data transfer. Different kinds of cryptography have different functionality depending on their capabilities. A symmetric key cryptography can be used to protect a local data while public-key cryptography can be used to protect data that occasionally move through the network. Hash functions are mainly used to check data integrity. It is also possible to combine different kinds of cryptography to increase data security.

In this paper, we have discussed about implementing one of the implementations in cryptography, digital signature, for increasing game security. By introducing a digital signature to game, an item validity can be checked. This can prevent game exploitation such as illegally generated item or item duplication that could ruin the game market.

Acknowledgment

Author would like to give thanks to the God Almighty that this paper can be written without any hindrance. Thanks to author's teacher, Dr. Ir. Rinaldi Munir, MT., for teaching author for the whole semester. Author would also like to give thanks to everyone around author, including family, friend, and everyone that support the author. Lastly, author would like to thank everyone that read this paper. May this paper give the reader some useful information to be used in the future.

References

[1] Ferguson, Niels, and Schneier, Bruce, Practical Cryptography, Wiley, 2003

[2] William Stalling, Cryptography and Network Security, Principle and Practice 5rd Edition, Pearson Education, Inc., 2015

[3] Hans Delfs, Helmut Knebl, Introduction to Cryptography Principles and Applications, Second Edition, Springer

[4] Douglas R. Stinson, Maura B. Paterson, Cryptography Theory and Practice, Fourth Edition

[5] Rinaldi Munir, Kriptografi, Edisi Kedua, Penerbit Informatika

[6] Menezes, Alfred J., Paul C van Oorschot, dan Scott A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1996. (e-book)

[7] Schneier, Bruce, Aplied Cryptography 2nd, John Wiley & Sons, 1996

STATEMENT

With this, I declare that this paper that I wrote is my own writing, not a summary, or a translation from other people's paper, and not a plagiarism.

Bekasi, 20 December 2021

Christopher Chandrasaputra
13519074