SCIS 2001 The 2001 Symposium on
Cryptography and Information Security
Oiso, Japan, January 23-26, 2001
The Institute of Electronics,
Information and Communication Engineers

Visible Watermarking using Verifiable Digital Seal Image

Hyuncheol Park* Kwangjo Kim*
hcpark@icu.ac.kr kkj@icu.ac.kr

Abstract---Many watermarking methods for protecting the intellectual property right and authenticating multimedia data have been proposed in recent years. It is possible to guarantee the authenticity of data using readable and fragile watermarking. In this paper, we propose a visible watermarking technique using verifiable digital seal image that is based on the concept of fragile watermarking. Seal image is just a binary (black or white) information after it converted to digital data by scanning. But, by adding digital signature to the seal image with a watermarking technique, we can get a verifiable digital seal image. The digital seal image can be used as an authentication for a digital image as well as a digital document. Inserted to digital image as a visible watermark, it guarantees the authenticity of cover image. This paper shows how to construct the verifiable digital seal image and how to apply it to cover image. Results are demonstrated by simulations.

Keyword : watermarking, signature, verifiable digital seal image

1. Introduction

1.1. Motivation

With the increased use of multimedia data, such as images, video and audio, the concern about intellectual property right (IPR) and data authentication has been raised. Digital watermarking scheme plays an important role in guaranteeing IPR and data authentication since it provides a mechanism that hides some information in cover data – images, video, audio and so on. This information is used for IPR and cover data authentication.

Current watermarking schemes are divided into readable watermarking and detectable watermarking. In readable watermarking, we extract watermark bit by bit from cover data and read its content. But, in detectable watermarking, we just verify the existence of watermark at cover data. Readable watermarking has more applications and cryptographic primitives can be applied more easily than detectable watermarking [1]. If readable watermarking algorithm is ideal, cryptographic primitives can provide confidentiality and authenticity to watermarked data. For example, encrypted user ID can be inserted into cover data and it can be used to detect illegal user [2]. For authenticity, the authentic information of copyright owner or cover data themselves can be inserted as watermarks [2][3][4].

But, readable watermarking algorithms are to be imperfect in a sense that if a attack succeeded to break watermark, the inserted watermark may be extracted with some error bits according to attack level [5][6]. From this reason, the encrypted ID and the authentic information for copyright owner may not be recovered, so the confidentiality and authenticity for copyright owner are not guaranteed. But, we can guarantee the authenticity of cover data, because the damaged authentic information represents that cover data was modified.

The watermarking scheme for confidentiality and authenticity of copyright owner is classified as robust watermarking and the scheme for authenticity of cover data is classified as fragile watermarking [4].

In this paper, we consider the application of cryptographic primitives to fragile watermarking and propose the construction of verifiable digital seal image and visible watermarking. Our schemes are based on the fragile watermarking and content-based digital signature proposed in [7].

1.2. Verifiable Digital Seal Image

A seal image means a stamped or written information into a document for the authentication of signer or document in real world such as a stamp, a hand written signature (an autograph) and a logo. After converted to digital data by scanning, it is just a binary image. But, by adding digital signature to it with a watermarking technique, we can get verifiable digital seal

* Information Security Group, Information and Communications Univ.,
58-4 Hwaam-dong, Yusong-gu, Taejon, 305-732, Korea

image (VDSI). VDSI is considered to be a digital seal which has the same validity as a seal image in real world and can be further used as visual authentication for digital documents and images.

We also get a visible watermarked image by applying VDSI to cover image. VDSI acts as a visual copyright mark and authentication for the image. Intuitively speaking, the visible watermarked image is like a poster with an approved stamp.

1.3. Outline of this paper

The paper is organized as follows. In Section 2, basic building blocks used for our scheme – watermarking algorithm and Elliptic Curve Digital Signature Algorithm(ECDSA) – are described. In Section 3, construction and verification process of VDSI are proposed. visible watermarking using VDSI is proposed in Section 4 and simulation results are described in Section 5. Conclusion of this paper follows in Section 6

2. Building Blocks

In this section, we review basic building blocks used for our proposed scheme.

Firstly, the generation and verification procedures of digital signature that is inserted into the scanned seal image is described. ECDSA is selected as digital signature algorithm among the various schemes for the view point of ease implementation and high efficiency. In digital watermarking system, the length of watermark should be limited because watermark damages the quality of watermarked data and a large watermark needs a large area of watermarked data. But, the forgery of watermark should be impossible [8]. In this point of view, inserted signature should be short and secure, so ECDSA fit to this requirement.

Secondly, watermark insertion method in frequency domain is reviewed. Many insertion methods have been proposed after the advent of watermarking concept. In this paper, a simple and well-known method[9] is reviewed and applied to our scheme. JPEG compression which is a kind of attack alters the cover image, but JPEG image is widely used. So, we should consider fragile watermarking which is robust against JPEG compression [4].

2.1. ECDSA [10]

ECDSA is the elliptic curve analogue of the digital signature algorithm (DSA) and is based on the elliptic curve discrete logarithm problem (ECDLP). Since ECDLP appears to be significantly harder than DLP(discrete logarithm problem in prime-order subgroups of Z_p^*), the strength-per-key-bit is substantially greater in elliptic curve systems than in conventional discrete logarithm system. Thus, smaller parameters can be used in ECC than with discrete logarithm (DL) systems but with equivalent levels of security [11].

2.1.1. Notations

n : order of an elliptic curve
G : base point of an elliptic curve
d : signer's private key
Q : signer's public key
$H()$: one way hash function

2.1.2. ECDSA Signature Generation

To sign a message m, an entity A does the following :

① Select a random or pseudorandom integer k, $1 \le k \le n-1$
② Compute $kG = (x_1, y_1)$ and convert x_1 to an integer x_1'
③ Compute $r = x_1' \bmod n$. If $r = 0$ then go to step ①
④ Compute $k^{-1} \bmod n$
⑤ Compute $H(m)$ and convert this bit string to an integer e
⑥ Compute $s = k^{-1}(e + dr) \bmod n$.
If $s = 0$ then go to step ①
⑦ A's signature for the message m is (r, s)

2.1.3. ECDSA Signature Verification

To verify A's signature (r, s) on m, B obtains A's public key Q. B then does the following :

① Verify that r and s are integer in the interval $[1, n-1]$
② Compute $H(m)$ and convert this bit string to an integer e
③ Compute $w = s^{-1} \bmod n$
④ Compute $u_1 = ew \bmod n$ and $u_2 = rw \bmod n$
⑤ Compute $X = u_1G + u_2Q$
⑥ If $X = O$, then reject the signature. Otherwise, convert the x-coordinate x_1 of X to an integer x_1', and compute $v = x_1' \bmod n$
⑦ Accept the signature if and only if $v = r$

2.1.4. Proof that Signature Verification Works

If a signature (r, s) on a message m was indeed generated by A, then $s = k^{-1}(e + dr) \bmod n$. Rearranging gives

$$k \equiv s^{-1}(e + dr) \equiv s^{-1}e + s^{-1}rd \equiv we + wrd \equiv u_1 + u_2d \pmod{n}.$$

Thus $u_1G + u_2Q = (u_1 + u_2d)G = kG$, so $v = r$ as required.

2.2. Watermark Insertion

2.2.1. Transform Domain Method

Many transform domain variations exist. One method is to use the discrete cosine transform(DCT) as a vehicle to insert watermark in an image [9]. DCT is important part of the most

popular lossy digital image compression system, the JPEG system. The JPEG system consists of three steps which are DCT computation, quantization of the DCT coefficients and variable-length code assignment [12]. In DCT computation step, the image is subdivided into 8×8 blocks and each block is transformed to frequency domain by DCT. Each watermark bit is inserted to each DCT coefficient block (see Figure 1). It means that minimum N blocks are needed to insert N bits of watermark.

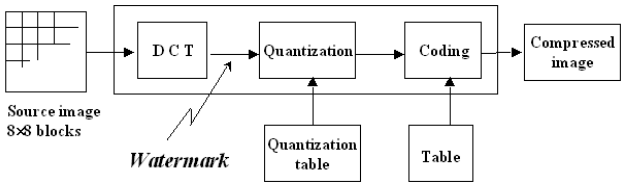


Figure 1. JPEG system and watermark insertion

Watermark insertion and extraction algorithms in frequency domain are shown in Figures 2 and 3. In these algorithms, two coordinates (u_1, v_1), (u_2, v_2) of DCT coefficient block B are selected and a watermark bit is represented by the relationship between their values. That is to say, if $B(u_1, v_1) > B(u_2, v_2)$, the block has a watermark bit '1', otherwise '0'. The two coefficients are swapped if their relative size does not match with the bit to be inserted. The watermark insertion algorithm must consider the robustness of watermark because DCT coefficient blocks are damaged in quantization step. So, the algorithm ensures that $|B(u_1, v_1)-B(u_2, v_2)| > x$ for some $x > 0$. We determine the minimum value of x to guarantee watermark extraction (See Appendix 1).

w : watermark

$l(w)$: length of w

```
for  i = 1 ,…, l(w)  do
    if  w_i == 1  then
        if  B_i(u_1, v_1) < B_i(u_2, v_2)  then
          swap ( B_i(u_1, v_1) , B_i(u_2, v_2) )
        end if
    else
        if  B_i(u_1, v_1) > B_i(u_2, v_2)  then
            swap ( B_i(u_1, v_1) , B_i(u_2, v_2) )
        end if
    end if
    adjust | B_i(u_1, v_1) - B_i(u_2, v_2) | > x
  end for
```

Figure 2. Insertion algorithm in frequency domain (F_{ins}())

```
for  i = 1 ,…, l(w)  do
    if  B_i(u_1, v_1) > B_i(u_2, v_2)  then
        w_i = 1
    else
        w_i = 0
    end if
end for
```

Figure 3. Extraction algorithm in frequency domain(F_{ext}())

3. Verifiable Digital Seal Image

In this section, the construction and verification of VDSI are described. Seal image in real world has validity for a document, a signer who stamped or wrote it and seal itself which is printed on the document. VDSI also has the same validity. The signature inserted to scanned seal image is generated from a digital document, signer's private key and seal image information. Seal image information is a binary matrix that consists of the binary information (black:0, white:1) of each pixel in the image. In JPEG image, the value of each pixel is changed after a quantization step. But, the binary matrix can be extracted by the threshold which is a border between black and white. Binary matrix extraction algorithm is shown in Figure 4.

```
for  j= 0,…, Image_height  do
    for  i= 0,…, Image_width  do
        if  Pixel_value[i,j] > Threshold  then
            Binary_matrix[i,j] = 1
        else
            Binary_matrix[i,j] = 0
        end if
    end for
end for
```

Figure 4. Binary matrix extraction algorithm

3.1. Construction of VDSI

VDSI is created by inserting ECDSA signature to seal image using watermark insertion method which is discussed in section 2.2. The concatenation of two hash values that came from a digital document and image information is applied to ECDSA as a message with signer's private key. The construction process is as follows.

I_s : scanned seal image

b_{mat} : binary matrix

D : digital document (e.g., ordinary text, web document, etc)

F_b() : function for getting binary matrix

F_{ins}() : function for inserting signature (as shown in Figure 2)

$ECDSA$() : ECDSA function

d : signer's private key

s : ECDSA signature

$\|$: concatenation

① $b_{mat} \leftarrow F_b(I_s)$, $h_1 \leftarrow H(b_{mat})$

② $h_2 \leftarrow H(D)$, $m \leftarrow h_1 \| h_2$

③ $s \leftarrow ECDSA(d, m)$

④ $VDSI \leftarrow F_{ins}(I_s, s)$

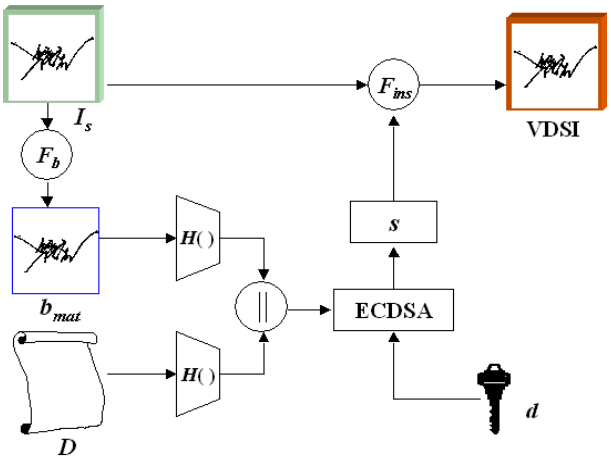


Figure 5. Construction of VDSI

3.2. Verification of VDSI

Signature extraction method is the same as watermark extraction method reviewed in section 2.2. The verification process is done with a signature, the concatenation of two hash values and signer's public key as follows.

$F_{ext}()$: function for extracting signature (as shown in Figure 3)

Q : signer's public key

① $b_{mat} \leftarrow F_b(\text{VDSI})$, $h_1 \leftarrow H(b_{mat})$

② $h_2 \leftarrow H(D)$, $m \leftarrow h_1 \| h_2$

③ $s \leftarrow F_{ext}(\text{VDSI})$

④ $ECDSA(Q, s, m) \rightarrow$ Accept / Not

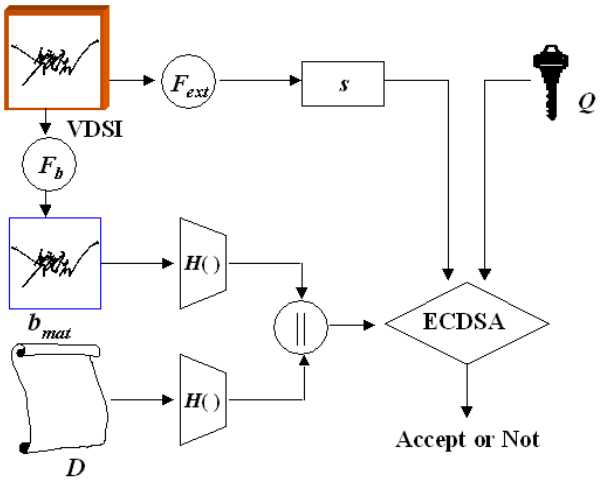


Figure 6. Verification of VDSI

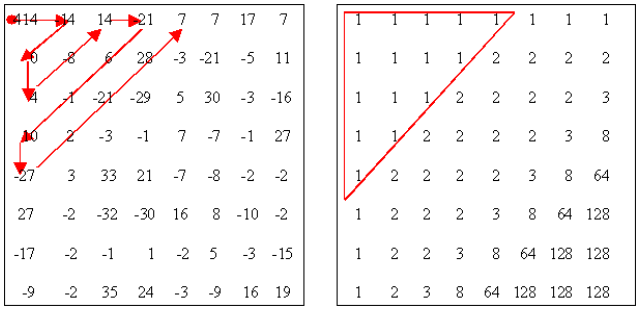
4. Visible Watermarking using VDSI

Visible watermarking is to print a visual watermark such as a logo on a cover image. In our scheme, VDSI printed on a cover image as a visual and verifiable watermark. The digital document used to generate signature is replaced by cover image and a binary matrix must be replaced by other data. Because general images don't have binary value - black and white, it's difficult to get a binary matrix from VDSI printed on a cover image. And a cover image as a digital document is changed by applying VDSI and JPEG compression, so its information that represents an important feature of its own is used instead. In this section, the construction of visible watermarking scheme applying to a JPEG image is proposed.

4.1. Getting Image Information

For a given image, it is assumed that we can get DCT coefficient blocks like Figure 7 (a). In general, DCT coefficients in low frequency area of each block capture the "essence" of the image while the values in high frequency area capture the fine detail and "noise" [13]. DCT coefficients in low frequency area can't be changed after quantization step if quantization table value corresponded to each coefficient was '1' as shown Figure 7 (b). Therefore, the important part of image information consists of DCT coefficients by zigzag scanning in low frequency area of all blocks (see Figure 7).

The obtained image information of cover image where seal image is printed exhibits the characteristics of the seal image and the cover image simultaneously. It means that we don't have to get 'image information' and 'digital document' separately, while VDSI should get them seperatly.



(a) DCT coefficients block (b) Quantization table

Figure 7. Getting image information

4.2. Construction of Visible Watermarking

I_c: cover image

I_c': cover image where I_s is printed

I_w : visible watermarked image

M : image information

$F_m()$: function for getting image information stated as Section 4.1

$\oplus$: function for image overlay

① $I_c' \leftarrow I_c \oplus I_s$: make 'black' pixels of I_s transparent and overlay I_s on I_c

② $M \leftarrow F_m(I_c')$

③ $s \leftarrow ECDSA(d, M)$

④ $I_w \leftarrow F_{ins}(I_c', s)$

Signature is inserted to the area where seal image is printed, because it minimizes damage by signature insertion.

4.3. Verification

① $M \leftarrow F_m(I_w)$, $s \leftarrow F_{ext}(I_w)$

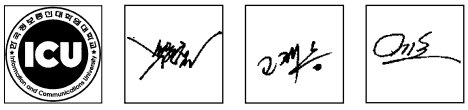
② $ECDSA(Q, s, M) \rightarrow$ Accept / Not

$F_m()$ gets M from DCT blocks after DCT computation step of JPEG system in construction process but after inverse quantization step in verification process.

5. Simulations and Results

We tested 144×144 pixels scanned seal images and 400×400 pixels grayscale images. For generating ECDSA signature, Elliptic Curve Cryptography (ECC) library based on OpenSSL 0.9.5a [14] and Rosing's ECC codes [15] was written and used. The curve order is 131 bits, so signature (r, s) has 262(= 131+131) bits long. But, we put each component of signature to 160 bits space and generated 320 bits signature because 131 bits curve has a low security level, so 160 bits order curve will replace it later. Our implementation was designed to use 320 bits signature for the ease of further extension.

5.1. VDSI test



(A) (B) (C) (D)

Figure 8. Seal images

The binary matrix extraction should be tested with threshold and components which affect the pixel values. Table 1 shows the extraction result by selected coordinates that signature bit is inserted to. The ***x***-value is determined by value of coordinates. The ***x***-value guarantees signature extraction. We select 0x80 as threshold because it is median between 0x00(black) and 0xFF(white). Quantization table is shown in Figure 9. As the coordinates and x-values vary, the quality of seal image was changed. But, the binary matrix extraction was possible at all cases.

Table 1. Binary matrix extraction test

coordi-nates	***x*** value	PSNR (dB)				result
		(A)	(B)	(C)	(D)	
(4,3),(3,4)	3	24.5	30.7	31.7	31.0	success
(5,3),(3,5)	3	25.5	30.1	32.4	32.5	success
(5,4),(4,5)	3	26.1	31.2	32.1	33.4	success
(6,4),(4,6)	3	25.6	31.9	32.5	32.6	success
(6,5),(5,6)	5	26.5	31.8	33.0	32.6	success
(7,5),(5,7)	15	26.6	31.5	33.1	33.2	success
(7,6),(6,7)	127	26.1	29.3	30.1	29.8	success

1	1	1	1	1	1	1	1
1	1	1	1	2	2	2	2
1	1	1	2	2	2	2	3
1	1	2	2	2	2	3	8
1	2	2	2	2	3	8	64
1	2	2	2	3	8	64	128
1	2	2	3	8	64	128	128
1	2	3	8	64	128	128	128

Figure 9. Quantization table

5.2. Visible watermarking

As shown in Figure 10, image (c) is created with an ICU mark (a) and ELAINE image (b), and then image information is extracted from (c). The image information includes seal image information and cover image information. 320 bits ECDSA signature was generated with image information and signer's private key and then inserted to the ICU mark area of (c) through JPEG compression process. (d) is a visible watermarked JPEG image using VDSI.

(a) Seal image (b) Cover image

(c) Seal image + Cover image (d) Signature + JPEG

Figure 10. Visible watermarking using VDS

We can get image information clearly after JPEG process and signature insertion because the corresponded quantization table values are '1's.

5.3. Analysis

VDSI is based on fragile watermarking. Table 2 shows the feature of VDSI whether the requirements of fragile watermarking [4] are met or not. In VDSI, we can determine whether a seal image has been altered or not, but can't locate any alteration made on seal image. A signature is inserted into seal image and it is invisible under normal viewing condition. And, VDSI is robust against JPEG compression.

Table 2. The feature of VDSI

Requirement	State
Determine whether image is altered or not	Satisfied
Locate alteration	Not
Integrate authentic data with cover image	Satisfied
Invisible authentic data	Satisfied
Robust against JPEG compression	Satisfied

VDSI on visible watermarked image represent IPR visual-apparently and can be verified for authenticity of IPR owner (signer) under JPEG compression. But, under other attacks, only authenticity of watermarked data can be guaranteed.

6. Conclusion

In this paper, we proposed a construction method of VDSI and visible watermarking using VDSI. To generate digital signature inserted to seal image, we adapted ECDSA. These schemes were based on fragile watermarking and content-based digital signature [7].

The concept of seal image for digital document was also represented in [16]. The seal image was used for guaranteeing document integrity and recovery. VDSI can't recover the alteration of digital document, but can guarantee the authenticity for document provider (signer) as well as document itself.

VDSI applied to cover image represents the IPR visual-apparently and guarantees the authenticity for cover image. VDSI is considered to be one way for applying cryptographic primitives to watermarking. As future works, application of VDSI for other data type such as video should be studied and more extensive research is required to guarantee the authenticity for copyright owner and cover data simultaneously regardless of attacks.

References

[1] "Watermarking Technology for Copyright Protection: General Requirements and Interoperability", http://www.imprimatur.alcs.co.uk/download.htm, 18 May, 1998

[2] Hiroyuki Inaba, Masao Kasahara, "Notes on Privacy Enhanced Protocol for Digital Watermark" , Proc. of SCIS'99, T4-2.4, Kobe, Japan, January, 1999

[3] A.Herrigel, J.Ruanaidh, H.Petersen, S.Pereira, T.Pun, "Secure Copyright Protection Techniques for Digital images", Proc. of IH'98, pp.169-189, April, 1998

[4] Min Wu, Bede Liu, "Watermarking for Image Authentication", Proc. of ICIP'98, vol.2, pp.437-441, 1998

[5] G..C.Langelaar, R.L.Lagendijk, J.Biemond, "Watermarking by DCT Coefficient Removal:A Statistical Approach to Optimal Parameter Settings", Proc. of SPIE'99, vol.3657, pp.2-13, 1999

[6] J.Fridrich, M.Goljan, "Comparing robustness of watermarking techniques", Proc. of SPIE'99, vol.3657, pp.214-225, 1999

[7] M.Schneider, S.F.Chang, "A Robust Content Based Digital Signature For Image Authentication", Proc. of ICIP'96, 1996

[8] "Research on digital watermarking at Aristotle university of Thessaloniki", http://poseidon.csd.auth.gr/signatures/report.html, 1997

[9] S.Katzenbeisser, F.A.P.Petitcolas, *Information Hiding Techniques for Steganography and Digital Watermarking*, Artech House, 2000

[10] D. Johnson, A. Menezes, S. Vanstone, "The Elliptic Curve Digital Signature Algorithm (ECDSA)", Certicom Whitepaper, http://www.certicom. com

[11] "Remarks on The Security of The Elliptic Curve Cryptosystem", Certicom Whitepaper, http://www.certicom. com

[12] A.M.Tekalp, *Digital video processing*, Prentice Hall, pp.394-403, 1995

[13] C.W.Brown, B.J.Shepherd, "Graphics File Formats", Manning, pp.220-229, 1995

[14] http://www.openssl.org/

[15] M.Rosing, *Implementing Elliptic Curve Cryptography*, Manning, 1999

[16] M.Iwakiri, Y.Murakami, W.Piyapisuit, Y.Nakamura, K.Matsui, "Signature Seal with recoverable Function of Text for Electronic Documents", SCIS2000 - D54, Okinawa, Japan, January, 2000

Appendix 1 : Choice of x

q : quantization table value corresponded with $B(u_1,v_1)$ and $B(u_2,v_2)$, $q > 0$

$X = B(u_1,v_1)$, $Y = B(u_2,v_2)$

a : quotient of (X / q), b : quotient of (Y / q)

$X' = aq$, $Y' = bq$

Proposition) $x \geq (q + (q\text{-}1))$ for $X > Y \rightarrow X' > Y'$

Proof)

$Y = qk_1 + r_1$, $X = Y + x = (qk_1 + r_1) + (qk_2 + r_2)$

(k_1, k_2, r_1, r_2 : integer , $0 \leq r_1, r_2 \leq q\text{-}1$)

1. $0 \leq r_1 + r_2 \leq q\text{-}1$

$X' = q(k_1+k_2)$, $Y' = qk_1$

$k_2 \geq 1$ for $X' > Y'$, $\therefore x \geq (q + r_2)$ …… (1)

2. $r_1 + r_2 > q\text{-}1$

$X' = q(k_1+k_2+\alpha)$, $Y' = qk_1$ (α : integer, $\alpha > 0$)

$k_2 \geq 0$ for $X' > Y'$, $\therefore x \geq r_2$ …… (2)

From eqs (1), (2) $x \geq (q + r_2)$

Therefore $x \geq (q + (q\text{-}1))$ because $0 \leq r_2 \leq q\text{-}1$