Idea: Suppose there are two people (I call them A, B) who want to pass a message.

Suppose B wants to pass a message to A.

First, A generates two sets of keys (Group1, Group2) through the RSA algorithm, and then sends the public key of each set of keys to B.

Suppose that B sends a message to A (the length is limited to 20, that is, the image used to hide the information needs at least 160 pixels [Note 1]). The first set of public keys is used to encrypt the position of the first pixel [Note 2]. The length of the information is encrypted with the second set of public keys.

Finally, B sends A three files which are new modified image and two numbers(the encrypted position and the encrypted length). Then, A will extract the message from decrypting the image with the two private keys.

Note:

1. First, set the following values:

"a – z" is assigned to "0 - 25" (which means 2 is c, 5 is f , etc.);

"A - Z" is assigned to "26 - 51";

":" is 52; "/" is 52; "?" is 53; "." is 54; " " is 55.

So, to represent a character in a message, a non-negative integer that is less than 56 will be used. Then, all the number will be converted in binary format. There will be 8 bits (actually it's 8 pixels) for each number, which means the value change is hardly possible to be detected by human eyes since the value change for each pixel is "+0 or +1".

In order to simplify the calculation, an integer is considered two part for conversion. For example, 25 is "2" and "5". So, "25" is "0010 0101".

1. To hide the message well, "+0 or +1" will be randomly put on each other pixel. So, in this condition, the position of the message is important. Thus, the position of the first pixel should be sent to A.