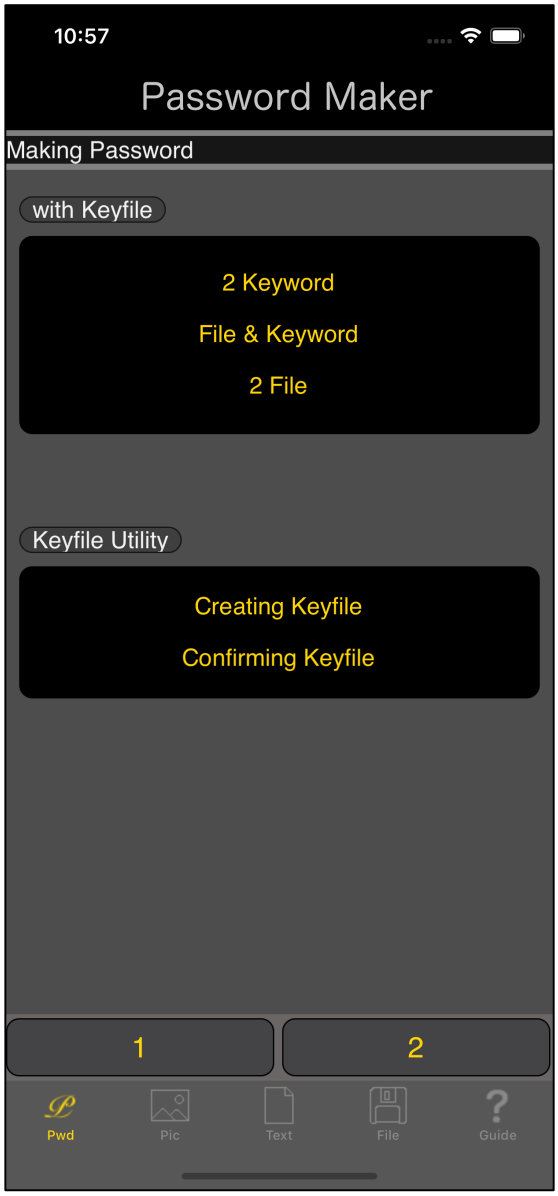
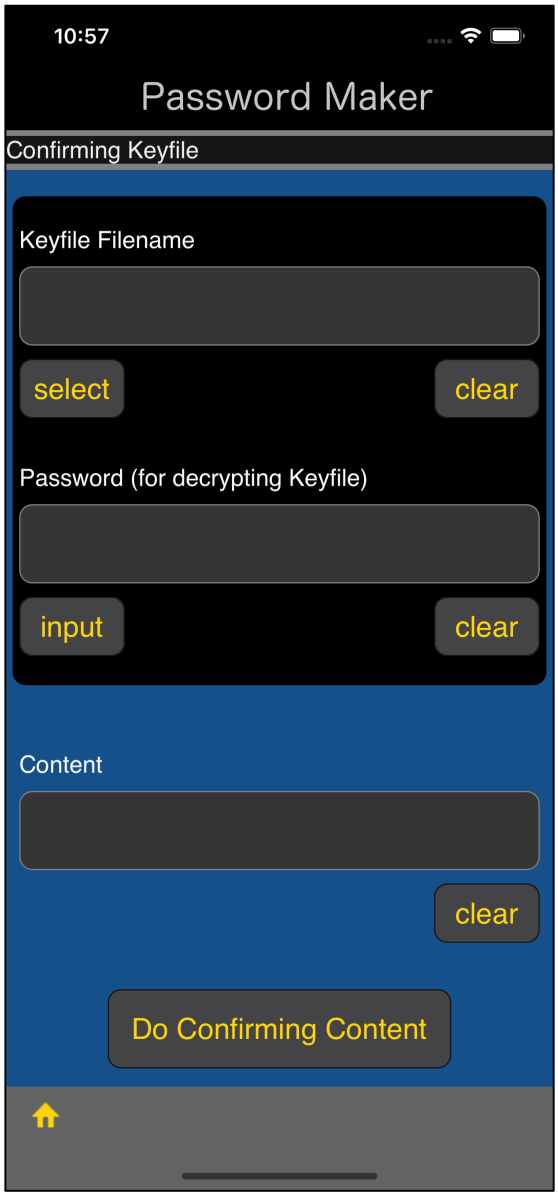
Key file confirmation

This command checks the key file.

It's like checking the contents and checking if the password is correct.



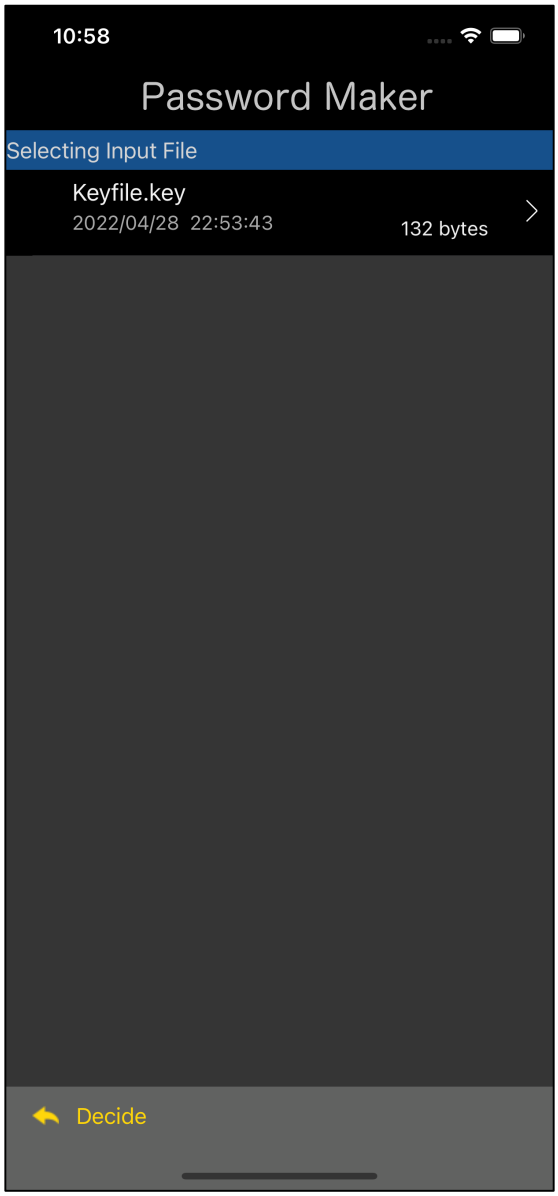
When you press the main "Confirming Keyfile" button, you will see the following view.

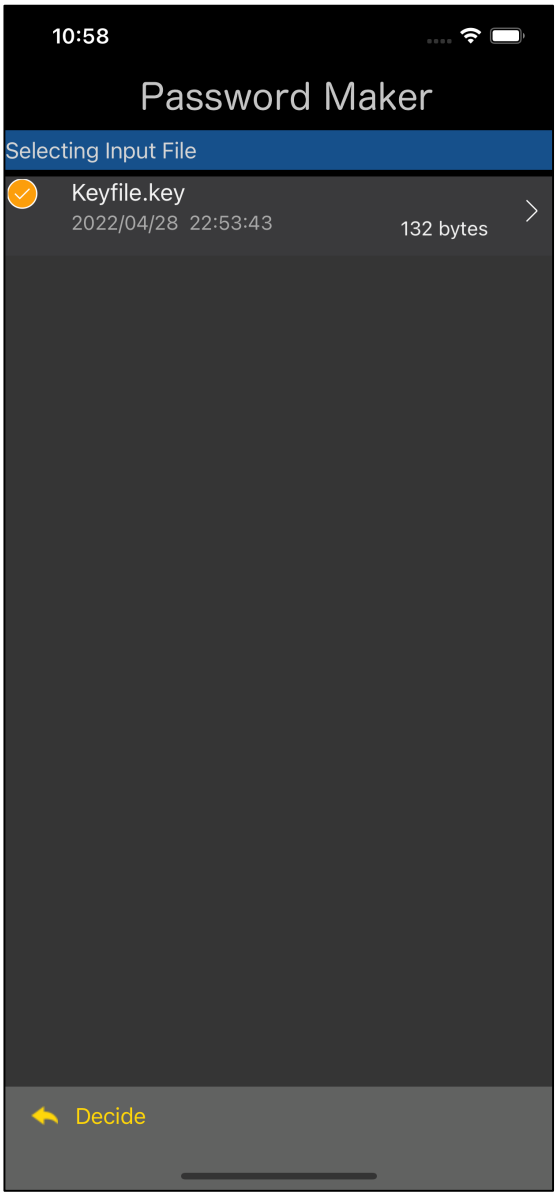


The user interface looks like this.

- Key file name
- Key file decryption password
- Contents value

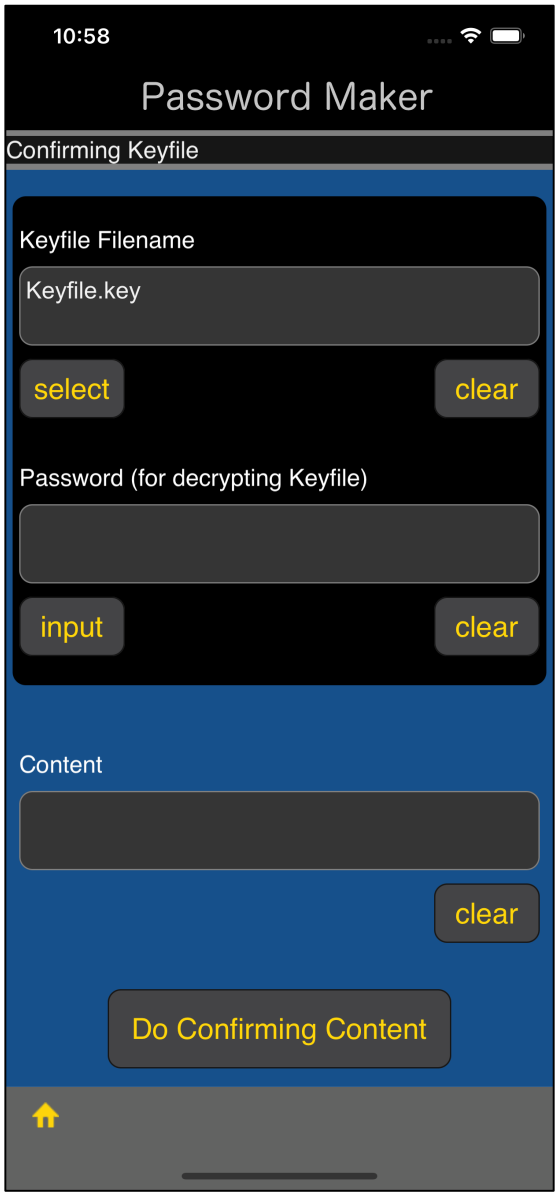
The meaning is like this.



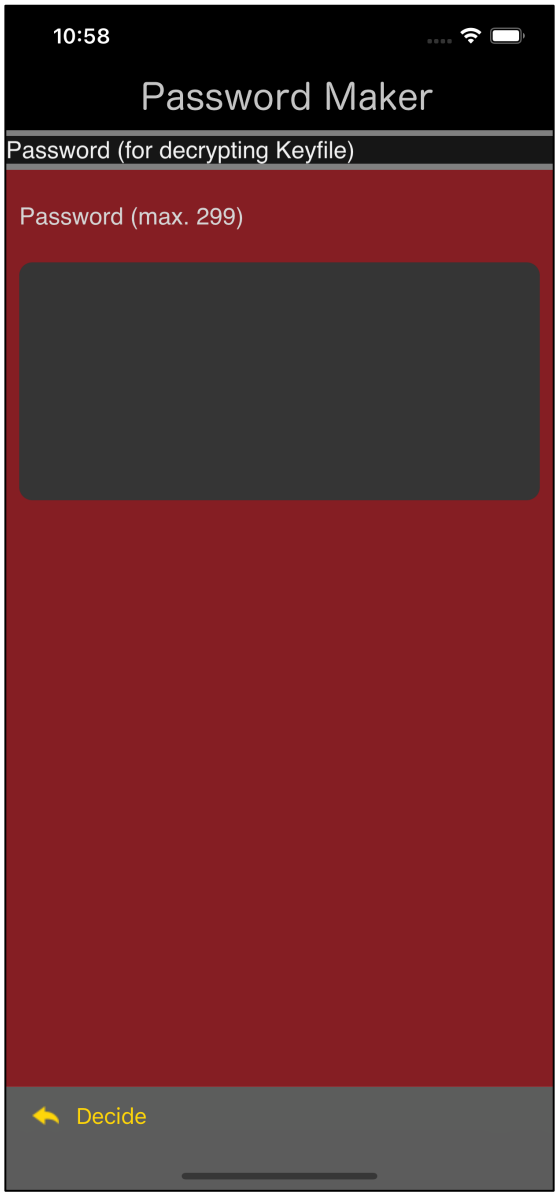


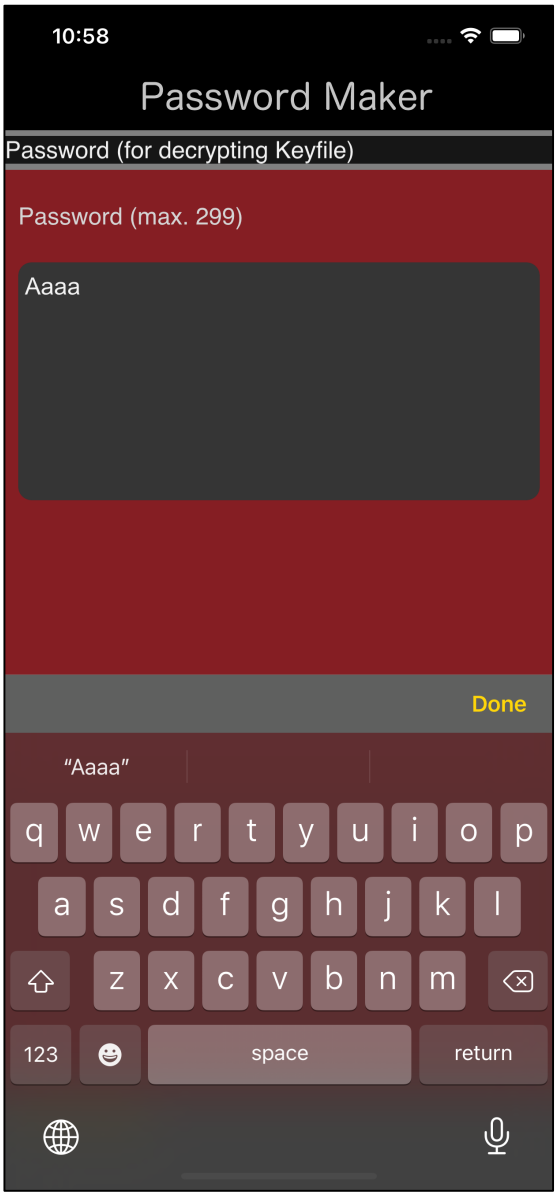
If you press the "Select" button at the bottom left of the text view of the key file file name, a table view like the one on the left will appear.

Select the file and press the "Decide" button on the toolbar to return.

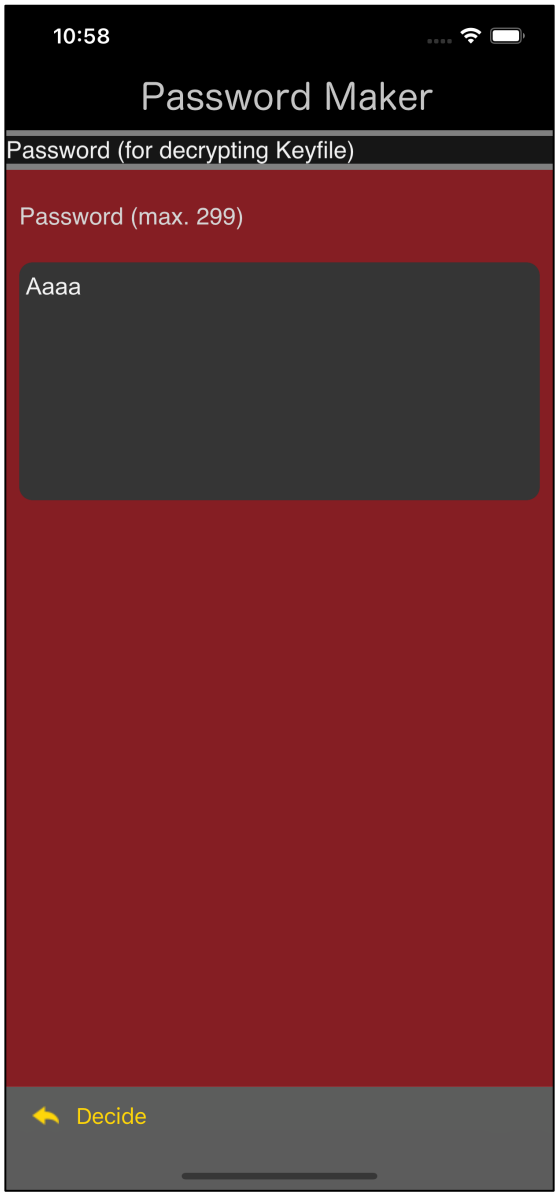


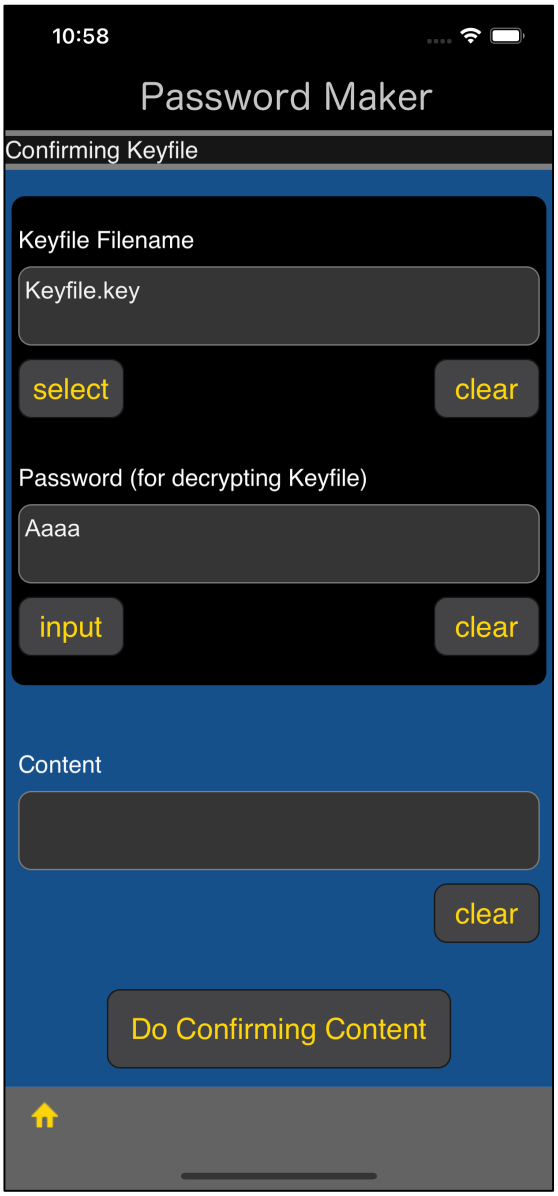
When you come back, it will be like this.



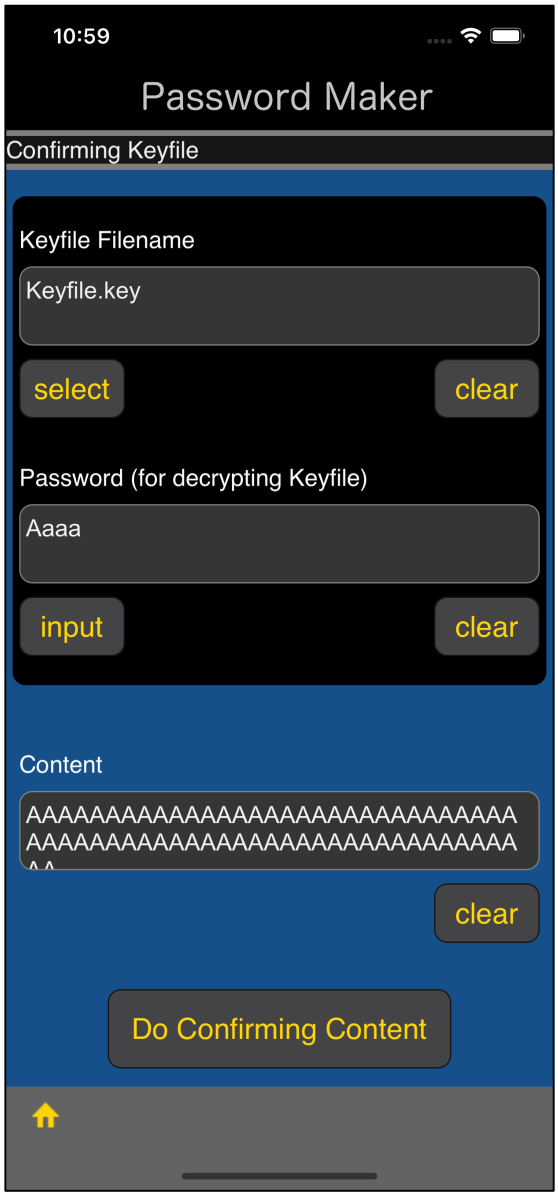


If you press the "Input" button at the bottom left of the text view for displaying the password, the view for entering the password as shown in the figure on the left will appear.

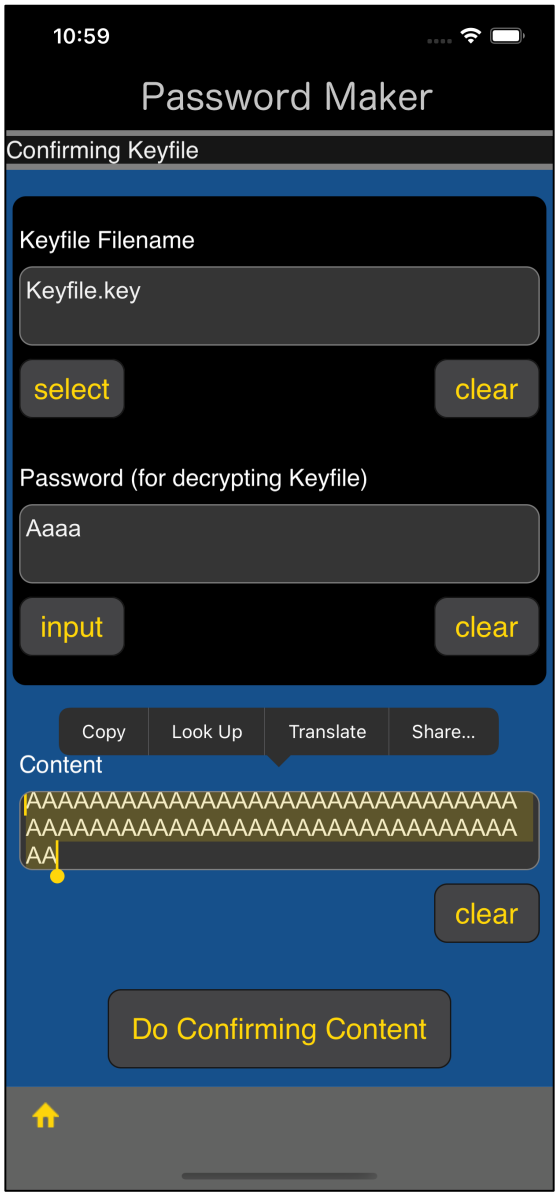




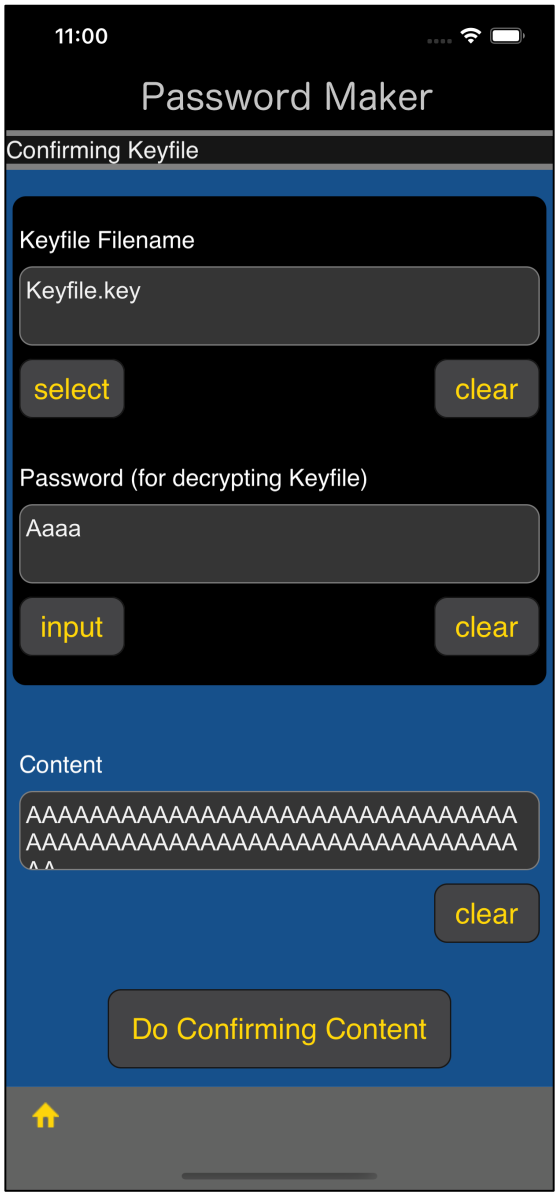
Enter the password and press the "Decide" button on the toolbar to return.



After entering the file name and password of the key file,
press the "Do Confirming Content" button to display the values stored in the key file like this.



You can copy the displayed value this way.



If you press the "Do Confirming Content" button and the value of the contents is displayed like this, then it can be said that

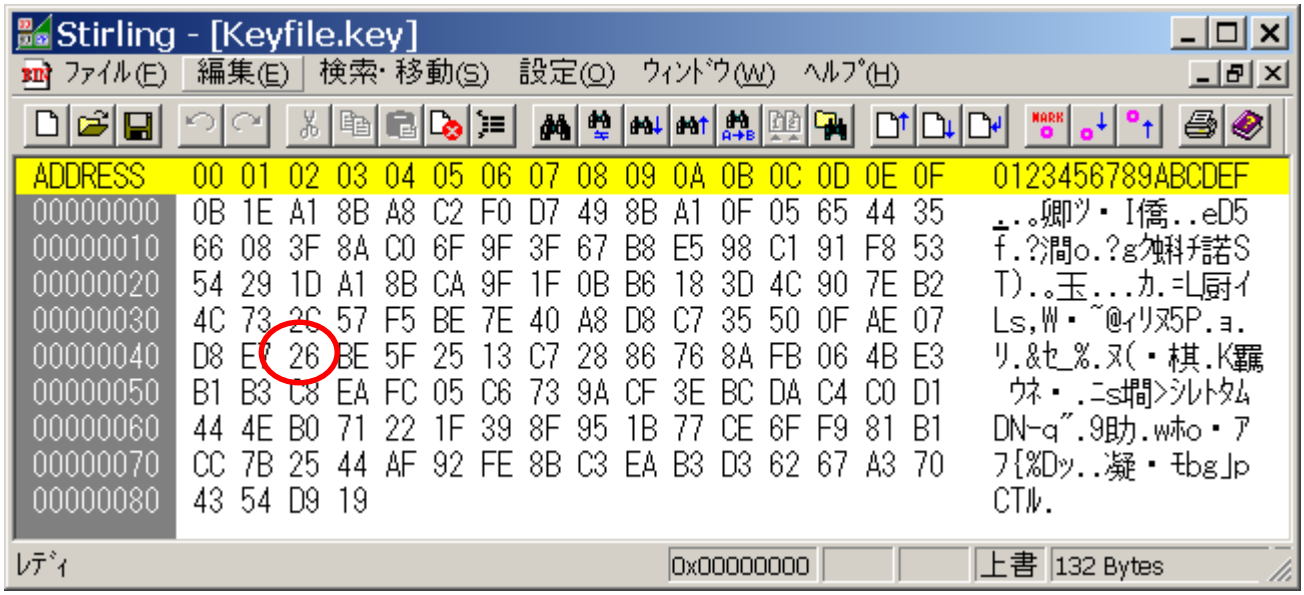
the password used for encryption is "Aaaa“, and

the encrypted file Keyfile.key has not been tampered with.

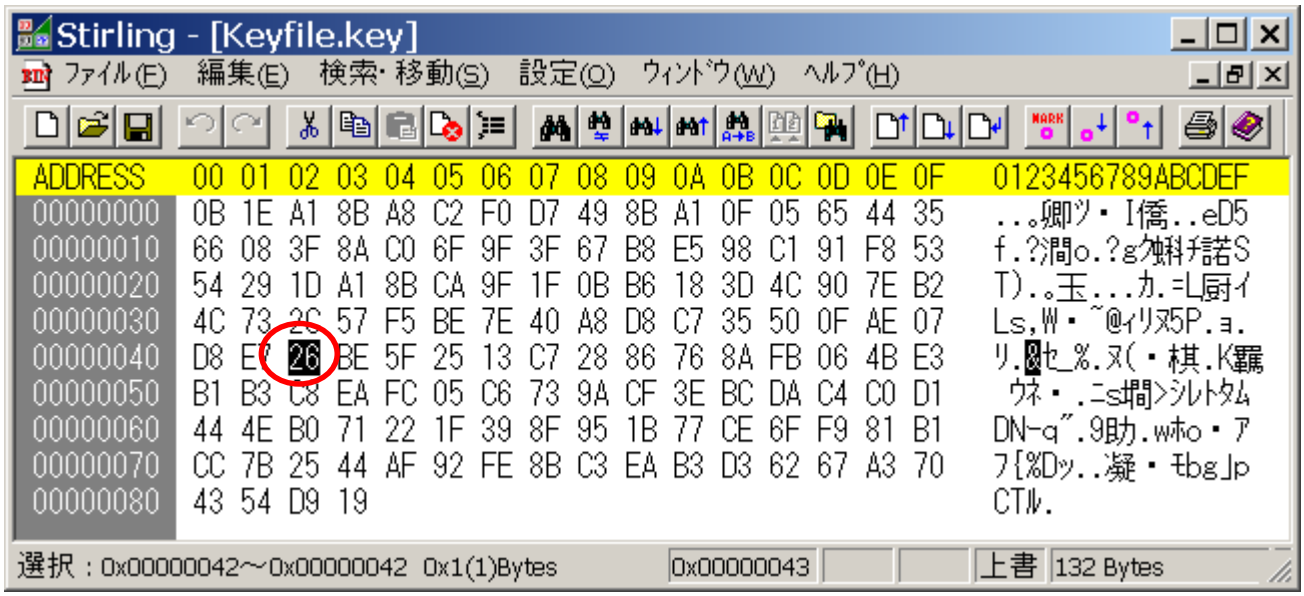
In the following cases, the value of the contents will not be displayed:

The password used for decryption is incorrect.

The encryption file Keyfile.key has been tampered with.

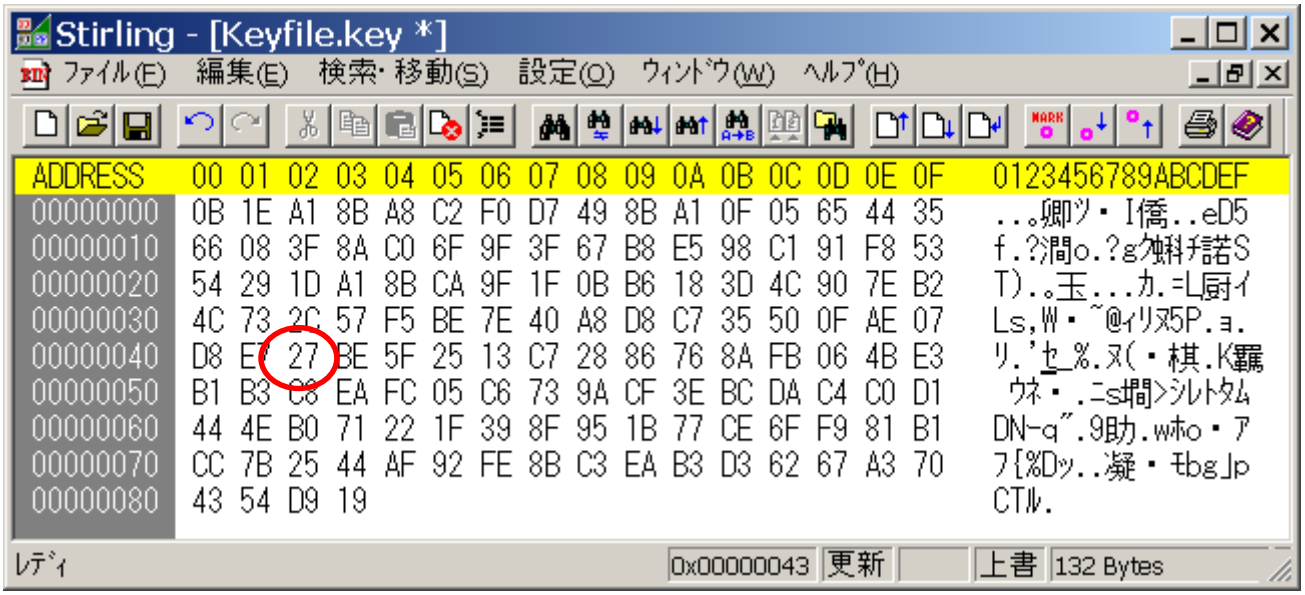


The contents of the file Keyfile.key used in this example are like this.



The value of the 66th byte (0x42) is

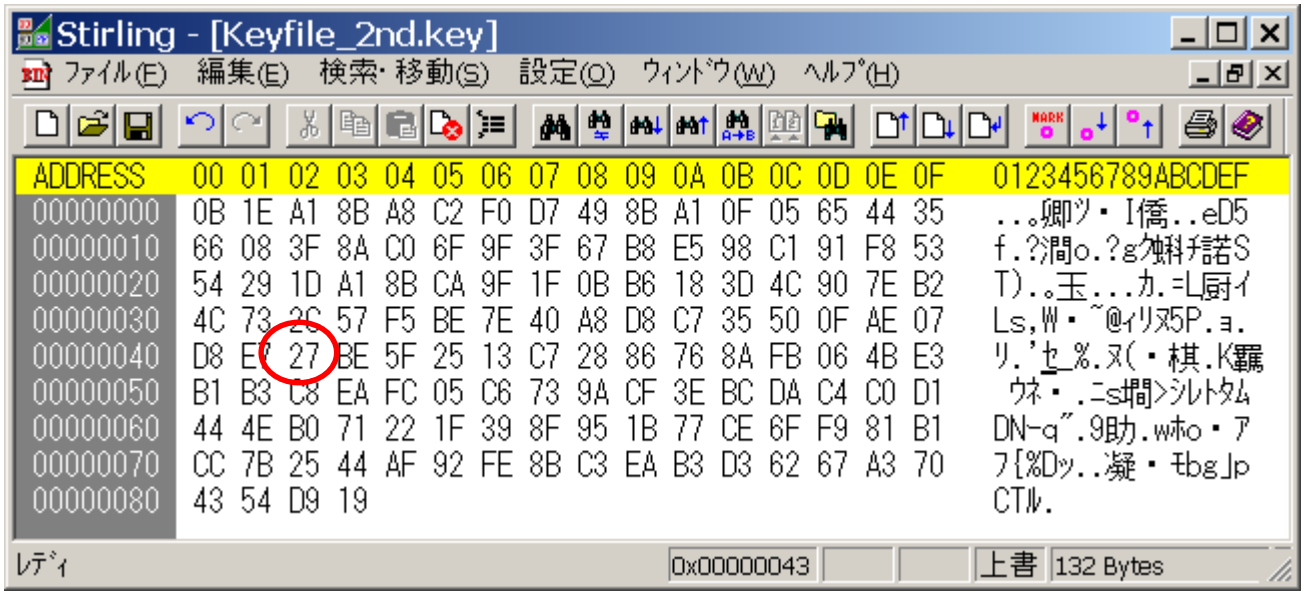
26.



26 → 27

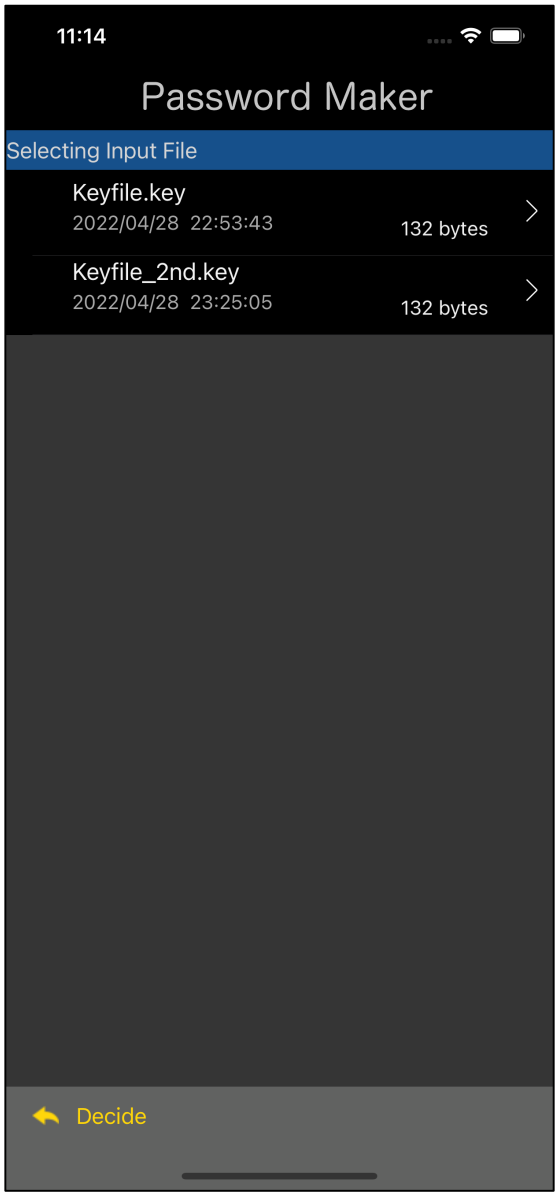
Let's change the value above. This means that it has been inverted by 1 bit.

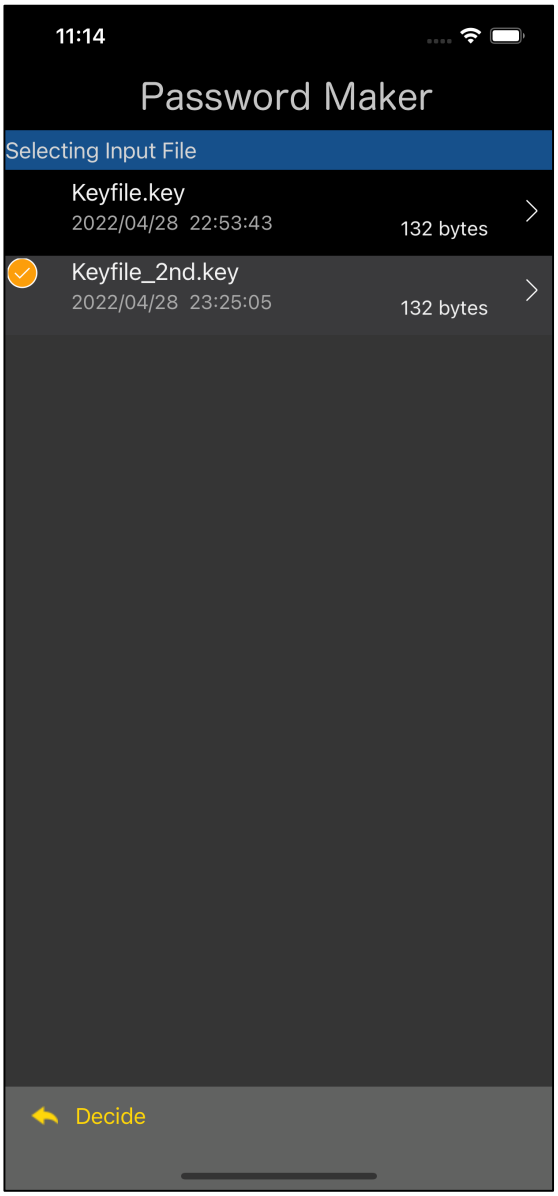
It means that it has been tampered with.



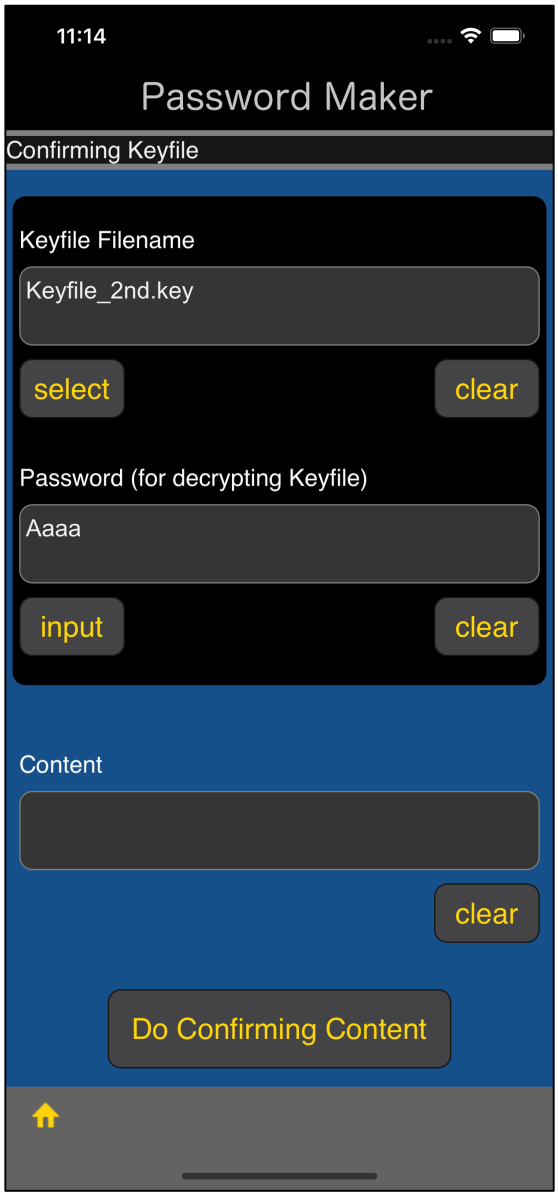
file name: Keyfile_2nd.key

Save it as this file name.



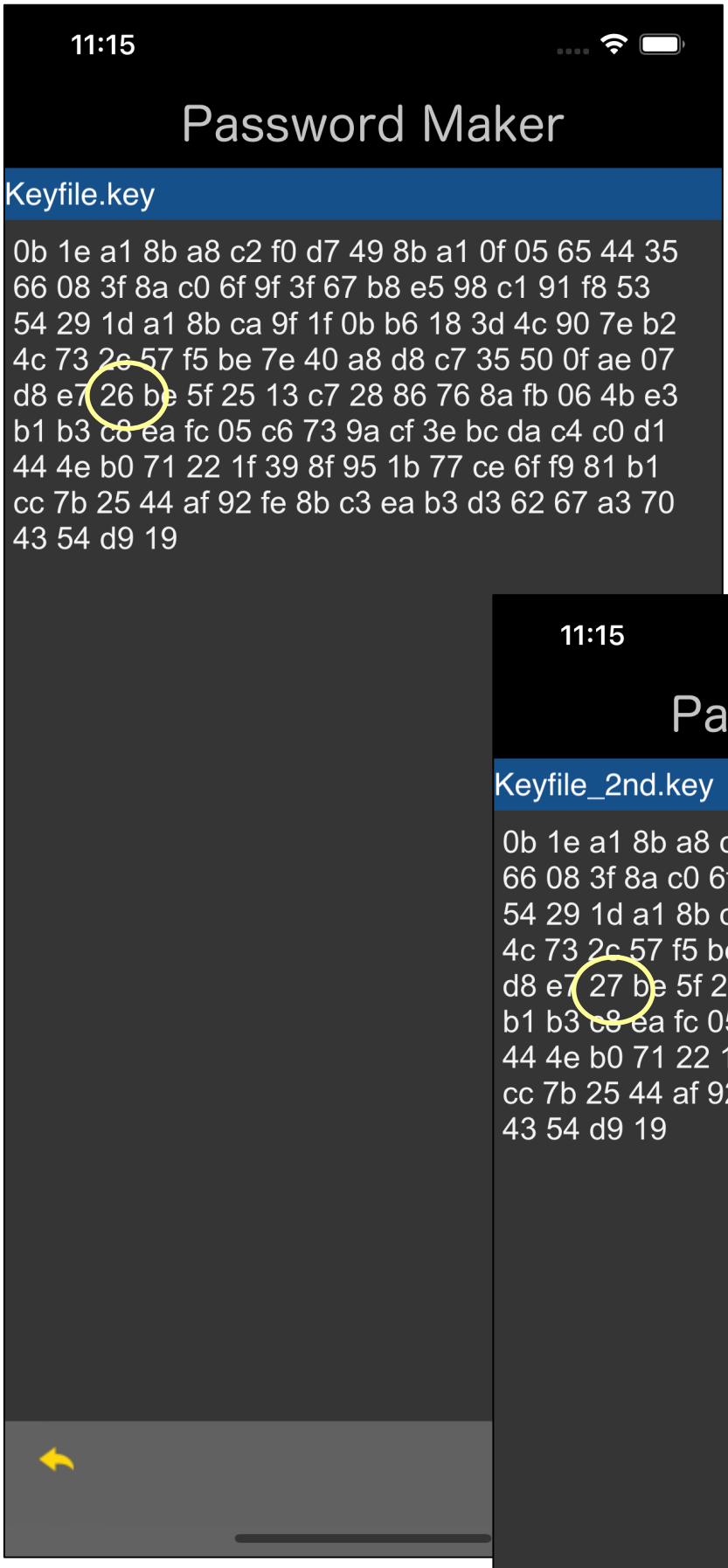


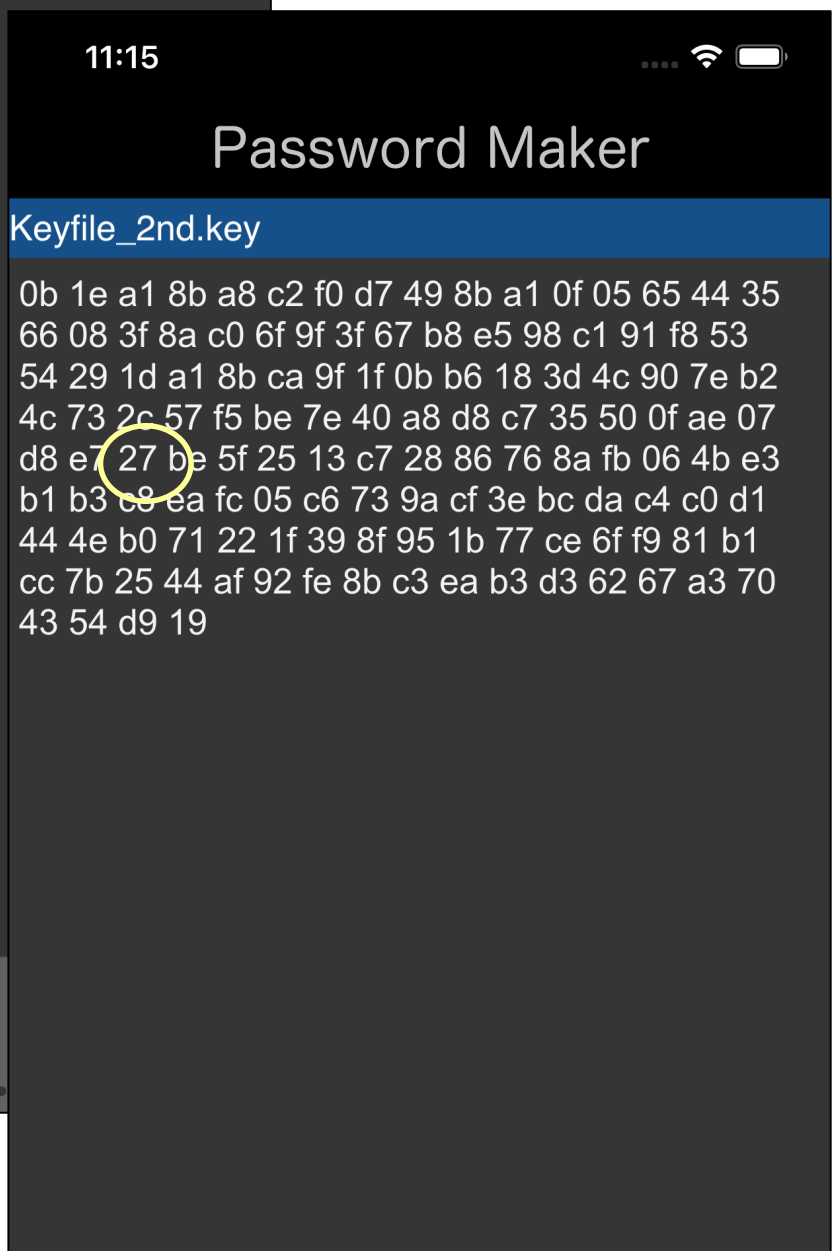
Send it back to iPhone and try decrypting Keyfile_2nd.key.



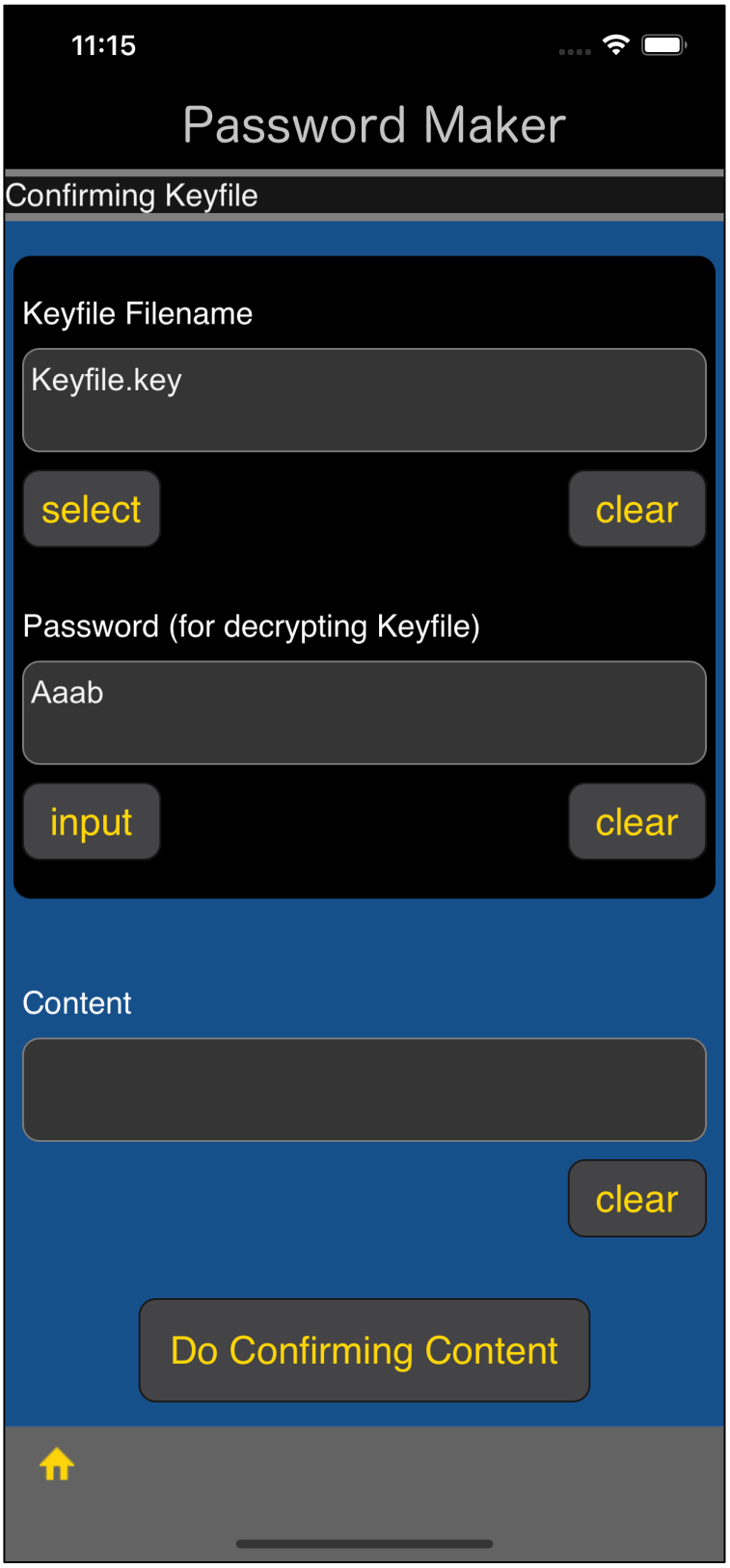
Even if you press the "Do Confirming Content" button, the value of the contents will not be displayed.

This is because it was detected that it had been tampered with (broken).


11:15
Password Maker
Keyfile.key
0b 1e a1 8b a8 c2 f0 d7 49 8b a1 0f 05 65 44 35
66 08 3f 8a c0 6f 9f 3f 67 b8 e5 98 c1 91 f8 53
54 29 1d a1 8b ca 9f 1f 0b b6 18 3d 4c 90 7e b2
4c 73 2e 57 f5 be 7e 40 a8 d8 c7 35 50 0f ae 07
d8 e7 26 be 5f 25 13 c7 28 86 76 8a fb 06 4b e3
b1 b3 c8 ea fc 05 c6 73 9a cf 3e bc da c4 c0 d1
44 4e b0 71 22 1f 39 8f 95 1b 77 ce 6f f9 81 b1
cc 7b 25 44 af 92 fe 8b c3 ea b3 d3 62 67 a3 70
43 54 d9 19


11:15
Password Maker
Keyfile_2nd.key
0b 1e a1 8b a8 c2 f0 d7 49 8b a1 0f 05 65 44 35
66 08 3f 8a c0 6f 9f 3f 67 b8 e5 98 c1 91 f8 53
54 29 1d a1 8b ca 9f 1f 0b b6 18 3d 4c 90 7e b2
4c 73 2e 57 f5 be 7e 40 a8 d8 c7 35 50 0f ae 07
d8 e7 27 be 5f 25 13 c7 28 86 76 8a fb 06 4b e3
b1 b3 c8 ea fc 05 c6 73 9a cf 3e bc da c4 c0 d1
44 4e b0 71 22 1f 39 8f 95 1b 77 ce 6f f9 81 b1
cc 7b 25 44 af 92 fe 8b c3 ea b3 d3 62 67 a3 70
43 54 d9 19

In this way, even if the contents of Keyfile.key are inverted by one bit, it can be detected.



The password is Aaab.

In this way, even if the decryption password is incorrect, it can be detected.

The value before encryption cannot be retrieved.

This key file can detect such a thing.

In that case, decryption will not be performed.

Password generation using the key file is not performed.

A password is generated using a strange key file.

Such a thing never happen.