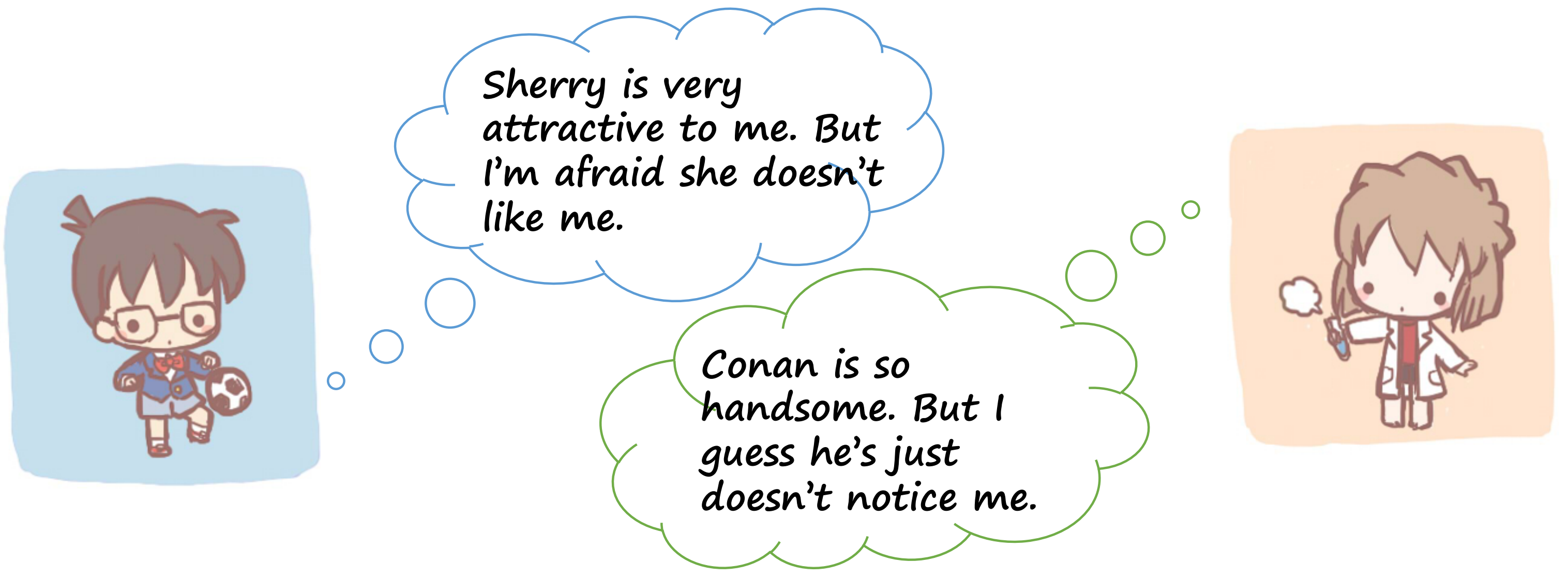
Proposal for Secure and Private Dating

Xiaohan Fu

University of California, San Diego

INTRODUCTION

People in the same group (e.g., students in the same class or employees in a same office), may develop a romantic interest in one another. However, this may cause embarrassment, perhaps even leading to an end of a friendship or difficulty in collaborating. For this reason, people are often shy about speaking out their interest in another person. However, at the same time, not being able to confess an interest would also be unfortunate when two person are coincidentally mutually interested. To avoid this, we aim to solve the problem at its source, which is to prevent the potential for embarrassment while still registering the interest, maintaining the security and privacy of the source party.


Sherry is very attractive to me. But I'm afraid she doesn't like me.
Conan is so handsome. But I guess he's just doesn't notice me.

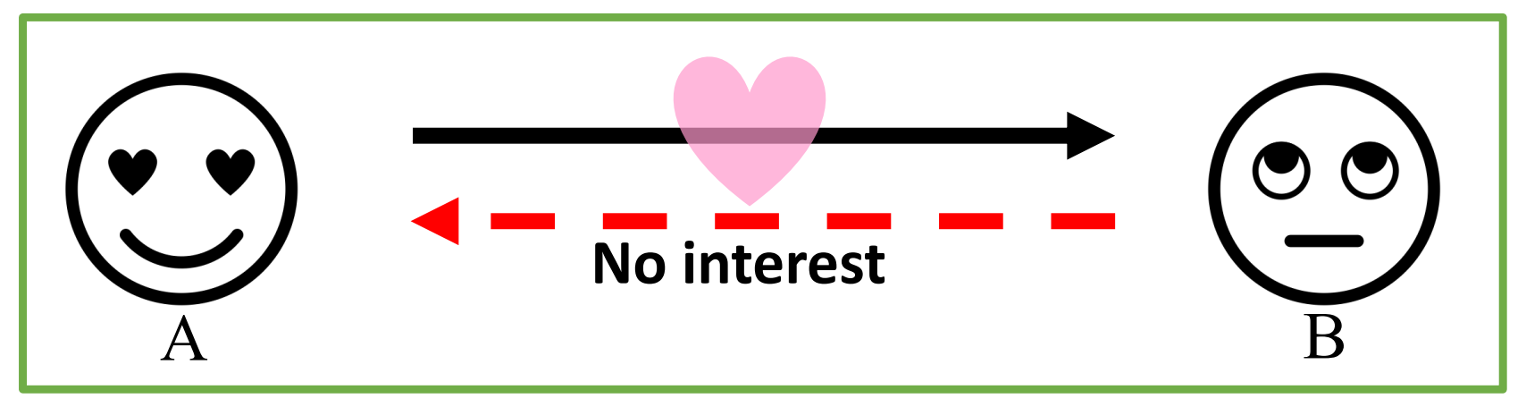
OBJECTIVE

Our goal is to develop a system composed of users as members of relatively small groups, allowing them to securely and privately express interest in another in the group without the risk of embarrassment.

To clarify our objective,

- an embarrassment case is defined as

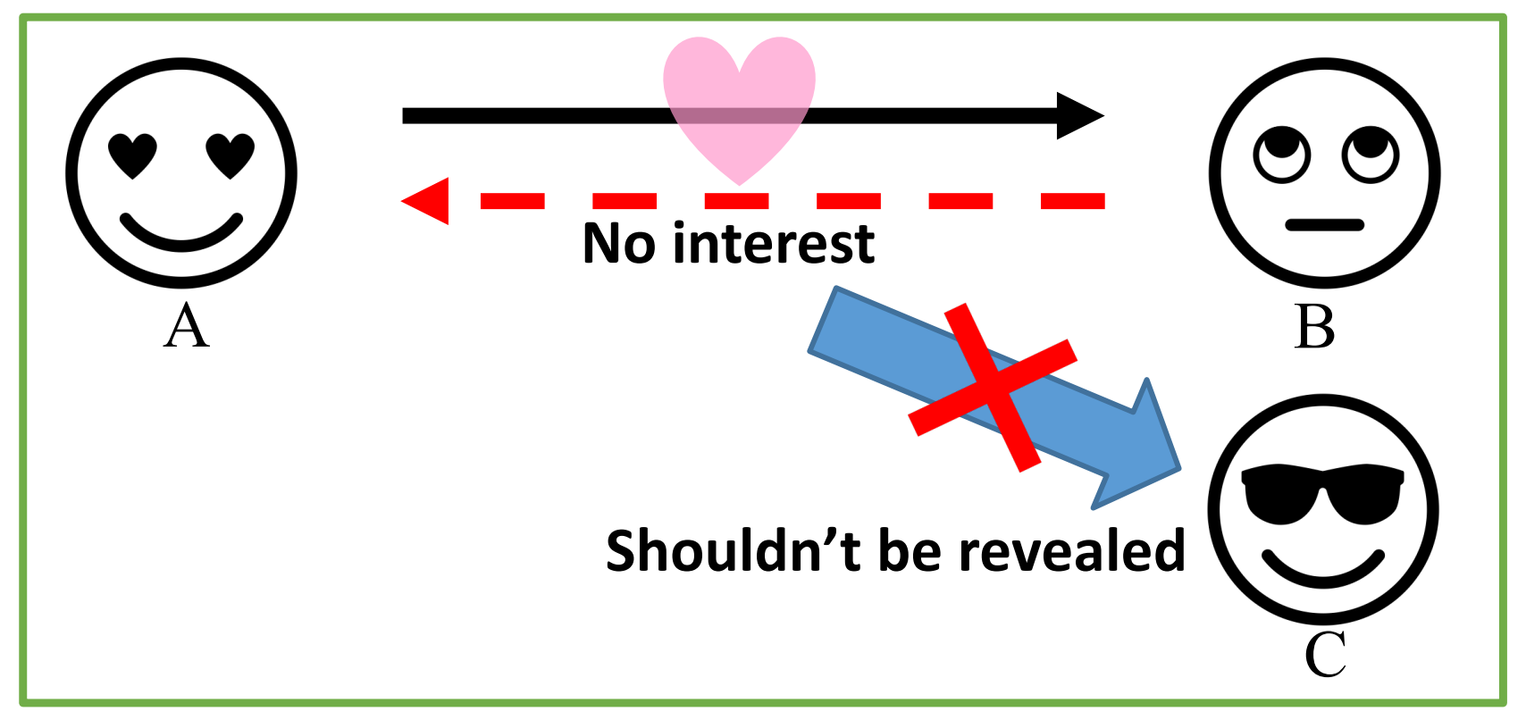
"The secret of person A being attracted to person B is revealed to B while B is not interested in A."



For example, in the above situation. Person B should not know that A loves him/her, otherwise it is an embarrassment case.

- privacy is defined as

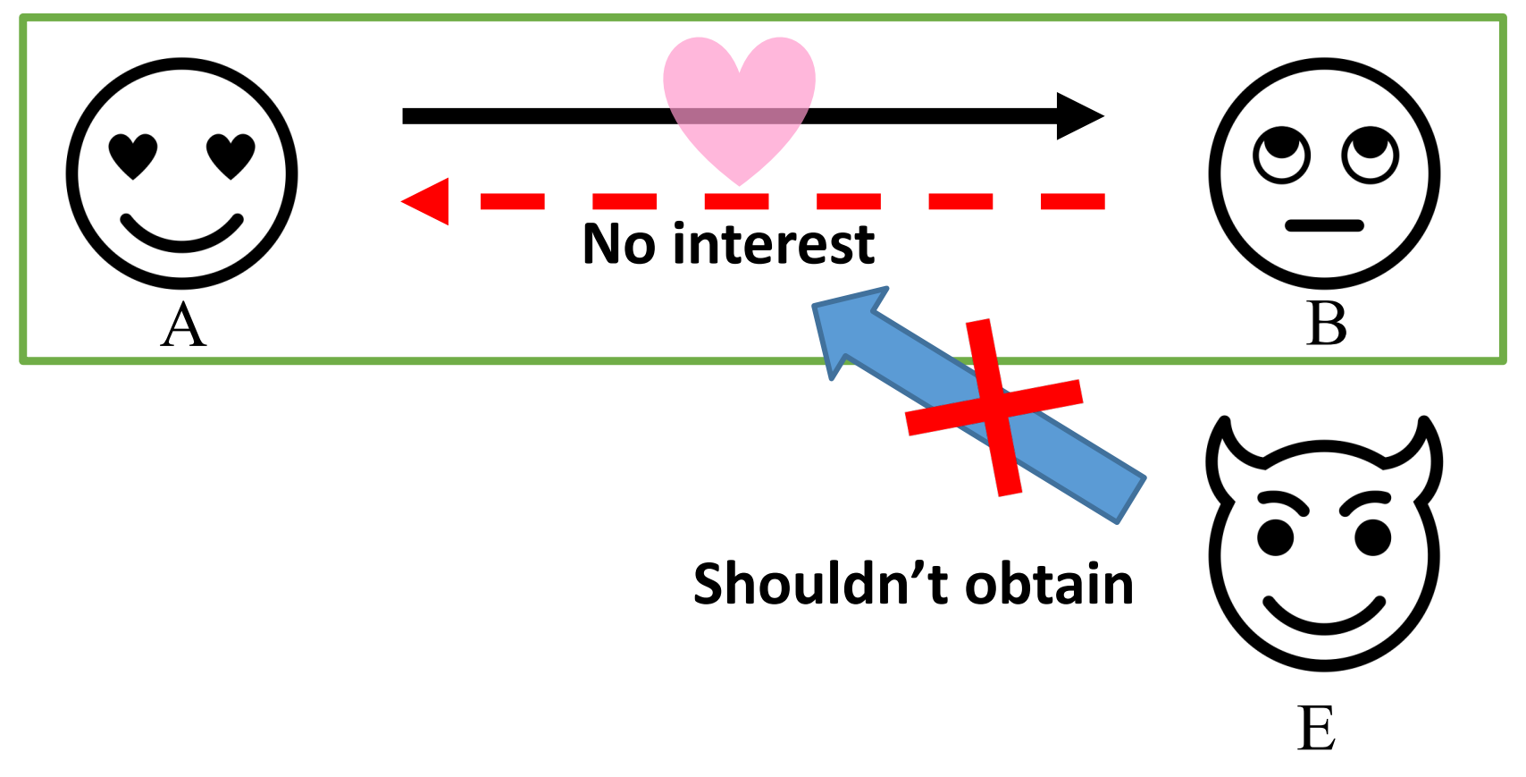
"The secret of person A being attracted to another arbitrary person, let's say B, must not be not revealed to any other users in the system at any time."



For example, in the above case, A, B and C are three users in the system, and C should not be able to discover the expressed interest of A in B. Otherwise, privacy is compromised.

- security is defined as

"The secret of person A expressing an interest in another arbitrary person, let's say B, can not be obtained by any other entities outside this system, including the service provider (E)."



In the above example, the external adversary E (e.g. an evil hacker, the service provider) should not be able to obtain any secret within this system, otherwise security is compromised.

- relatively small group is defined as

"a group of no more than 500 members"

A typical example of such a group is a year of students in one department, e.g., Class 2019 of CSE Department. The group size limit is chosen as 500 following the design of WeChat, the most popular chatting software in China.

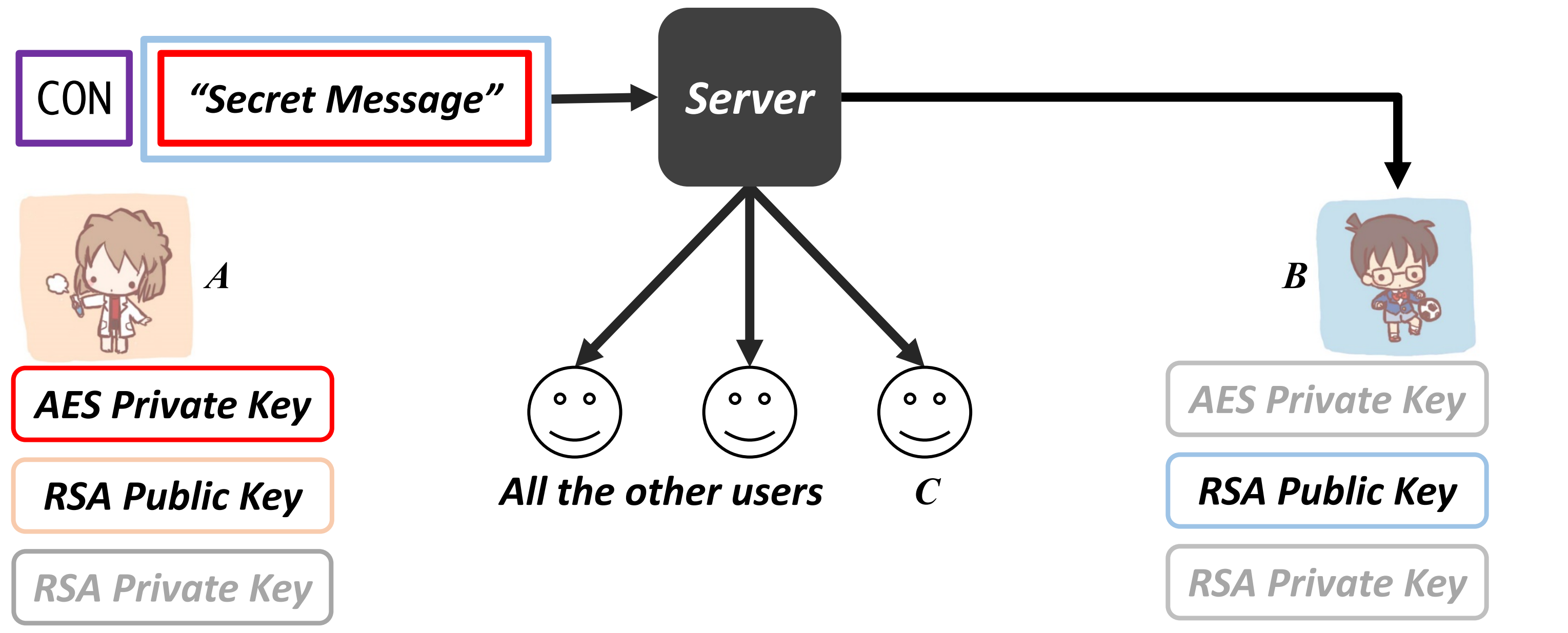
PROPOSED SOLUTION

Our solution makes use of asymmetric and symmetric encryption. Each user is initialized with an AES key, and a pair of RSA key, with the public key public. Users will pass three types of message: confession, response, or acknowledgement, all encrypted, to the server, which acts as a router and always (and only) broadcasts incoming messages to the other users.

To help understand, we use an example where user A and B are mutually interested in the following illustration.

- Confession

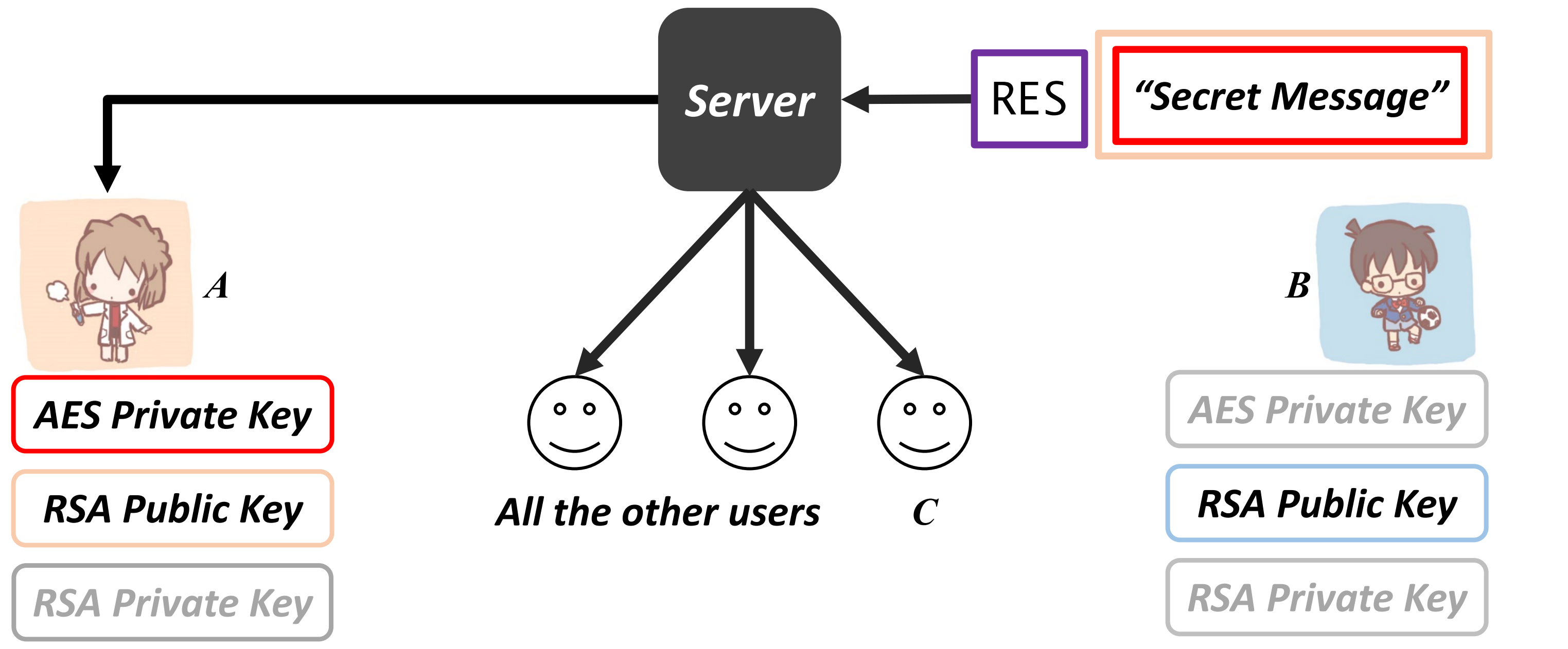
Confession message is sent when a user initiates a confession. It is a random secret encrypted with the user's own symmetric key then again encrypted by the target user's asymmetric public key plus a header showing it's a confession message.



All clients will try to decrypt an incoming confession message with their own RSA Private Keys. Only the target user, in this example B, is able to decrypt this confession message.

- Response

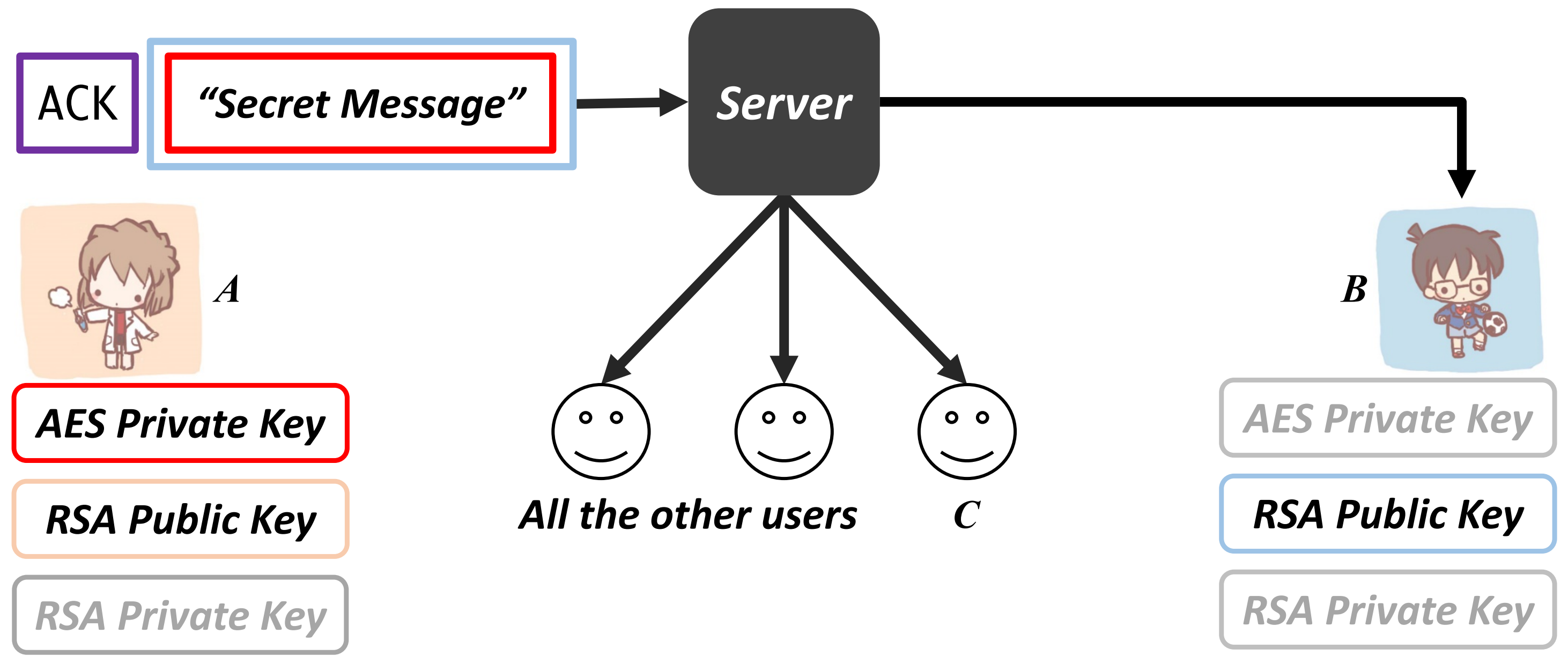
A response message is sent only after a user successfully decrypts an incoming confession message. In the above example, user B is the one going to send a response message subsequently. The decrypted package is going to be encrypted by user B's target's asymmetric public key, which forms the response message.



Similarly, all clients will try to decrypt an incoming response message with their own RSA Private Keys. The user who can successfully decrypt it, will continue to decrypt the message further with his/her own AES key. Only the target user of B, in this example A, is able to decrypt this response message. In addition, she will find she can decrypt it further with her own AES key.

- Acknowledgement

An acknowledgement message is sent only when a user is able to decrypt a response message AND is able to further decrypt it with the AES key. This would indicate a match between A and B. But at this moment, only the receiver of the response message knows this (in our example, user A). The acknowledgement message is sent to inform the other side, like a three-way handshake. Just as what we did for the confession message, a header showing this is an acknowledgement message added ahead of a secret encrypted by the target user's RSA public key.



All clients will try to decrypt an incoming acknowledgement message with their own RSA Private Keys. Only the target user, in this example B, is able to decrypt this acknowledgement message. At this moment, both users now know they are mutually interested!

However, it might also be possible that user B is interested in C but not A. Things would differ after users receive the response message from B. In this case, user C instead of user A would be able to decrypt. However, she will not be able to further decrypt it with her AES key since that is encrypted by A. The requirement to send an ACK message is not satisfied and the transaction ends at this stage.

As for actual implementation, messages are all temporarily stored on server for each group. Once a user logs into the system, its client will pull all unread messages from server and take actions corresponding to the message type. The server will clear those messages read by all users in a group. In addition, in order to avoid privacy leakage from network flow, every user's client will send random types of dummy messages e.g. encrypted with random key to the server in random period.

LIMITATIONS AND PERSPECTIVES

A fundamental limitation of this proposed solution recently pointed out is this protocol has not yet been proved to be secure in any formal model. Since this problem appears to be novel, though with certain similarity to existing problems such as cryptography dating and public computation, a new formal security model should be established for it. This is our next focus.

Attentive readers may have noticed some intrinsic limitations on this protocol as well:

- The server, as a "router," is a performance bottleneck and a single failure point.
- Privacy is compromised when group size is very small, e.g. with only 3 users: if A is attracted to B but B is attracted to C, this information is revealed when C receives a response message from B.
- Revocation of interest is not supported at this moment and it is uncertain how to support it.
- Traffic may be unacceptably intensive with large group sizes.

After the development of a formal security model for this problem, a more comprehensive protocol with proper crypto methods to solve the above limitations would be our next target. Suggestions are more than welcome.

ACKNOWLEDGEMENTS

This project is the joint effort of Xiaohan Fu, Yihao Liu, Yunguo Cai, Yichen Hu and Sifan Jiang, supervised by Manuel Charlemagne from Shanghai Jiao Tong University – University of Michigan Joint Institute. Thanks also to our sponsor, Dr. Yunlong Guo from Shanghai Jiao Tong University – University of Michigan Joint Institute.

Special thanks to Mihir, Daniele and Deian from UCSD for pointing out a few more limitations.