Secure SMS exchange.   
RC6 + Blind RSA signature + DH key exchange

**Students : Rami Amasha + Jasmeen Mura.**

**This paper describes the protocol of our project, and describes the procedures for the specific steps .**

**The protocol :**

* Generate private key for both sides via DH key exchange+ BLIND RSA for digital signature.( RSA for digital signature is used to make sure that the data has not been corrupted and blind signature for authenticate the sender's identity).
* Form the shared key generate round keys according to algorithm RC6 key generation step.
* Each message is encrypted\ decrypted via RC6 algorithm, using blind RSA signature for authenticating the signer identity and RSA signatures for data integrity.

**DH key exchange with blind RSA:**

1) Key Generation: Alice and Bob generate their private DH exponents and RSA key pairs.

2) RSA Signature: Alice and Bob sign their DH public keys with their RSA private keys.

3) Exchange of Signed Public Keys: Alice and Bob send their DH public keys and their RSA signatures to each other.

4) RSA Signature Verification: Alice and Bob verify the received signed public keys using the sender's RSA public key.

5) Alice asks Bob to sign the shared secret key as described earlier, if verified then its indeed Bob. (Same for Bob)

**The full system :**

* **Message sending (Sender = Alice):**

1) generate random blinding factor.

2) Alice computes the blinded message by hashing the blinding factor concatenated with the actual message: blinded\_message = SHA256(blinding\_factor + message).

3) Blind the message using the server's public key: compute blind\_message = blinded\_message^e mod N, where e is the public exponent of the server's RSA key and N is the modulus.

4) Alice encrypts the blinded message using her RC6 encryption key.

5) Alice sends the encrypted and blinded message to the server.

* **Message Storage and Delivery: (server)**

1) The server receives the encrypted and blinded message from Alice and stores it securely without knowing Alice's identity or the message content.

2) The server generates a signature on the encrypted message using its private key: signed\_blind\_message = blind\_message^d mod N.

1. The server stores the signature along with the encrypted message.

* **Message Retrieval:**

1. Recipient (Bob):
2. Bob retrieves the encrypted and blinded message from the server along with the server's signature from the server.

* **Message Verification and Unblinding:**

1)Bob decrypts the encrypted message using his RC6 encryption key.

2)Bob verifies the signature of the message using the server's public key.

verified\_blind\_message = signed\_blind\_message^e mod N.

3)Bob cannot directly access the original message due to the blinding factor.

4) If needed, Bob can communicate with Alice through a secure channel provided by the server to gather additional information or clarify details.

**Note: the full implementation needs a server to be implemented so we couldn’t combine the digital signature with the DH and rc6 algorithm, we implemented it separately. Once a server is implemented, the system can easily be completed by calling the blind signature function.**