There is A and B . When we press buttons for sign and encrypt to B, what’s actually going on. Hashes, session keys are included.

When you press buttons to sign and encrypt a message from entity A to entity B, a series of cryptographic operations take place to ensure the confidentiality, integrity, and authenticity of the message. Here’s a simplified explanation of what typically happens, incorporating hashes and session keys:

1. **Hashing**: First, a hash of the message is generated using a cryptographic hash function. This creates a fixed-size digest (hash) of the message, which is unique to the contents of that message.
2. **Signing (Digital Signature)**: Entity A uses its private key to encrypt the hash, creating a digital signature. This signature can be used by entity B to verify that the message has not been altered and that it was indeed sent by entity A.
3. **Generating a Session Key**: A session key is generated for this particular communication session. This is a symmetric key, which means the same key is used for both encryption and decryption. The session key is generated in such a way that both A and B can derive it, often through a key exchange algorithm like Diffie-Hellman.
4. **Encrypting the Message**: The original message is encrypted using the session key with a symmetric encryption algorithm (like AES). This means that only someone with the session key can decrypt and read the message.
5. **Encrypting the Session Key**: The session key itself is then encrypted with entity B's public key. This ensures that only entity B can decrypt the session key using its private key.
6. **Transmission**: The encrypted message, the encrypted session key, and the digital signature are sent to entity B.
7. **Decryption by Entity B**: Upon receiving the transmission, entity B first uses its private key to decrypt the session key. Then it uses the session key to decrypt the message.
8. **Verification**: Entity B uses entity A's public key to decrypt the digital signature to retrieve the hash. Entity B then hashes the decrypted message using the same hash function as entity A and compares this hash to the one retrieved from the digital signature. If they match, it confirms the message's integrity and authenticity.

This process combines the efficiency of symmetric encryption (for the message body) with the security of asymmetric encryption (for the digital signature and the session key exchange). It ensures that the message is kept confidential, can only be read by the intended recipient (entity B), and that it has not been tampered with in transit. It also confirms that the message was indeed sent by entity A (non-repudiation).