ISS Lab Project Description

-by Sarvagya Gaur, Rishi Goel

Algorithm – Attribute Based Encryption

Objective – To perform encryption and decryption using the ABE algorithm and to analyze the same.

First, we need to understand what Public key encryption is – it's a type of cryptography that involves a pair of keys known as public key and private key. Each public key is published and the corresponding private key is kept secret. Data that is encrypted with the public key can be decrypted only with the corresponding private key. For example – User1 wants to send message "Hello World" to User2, it will use its own Private key and User2's Public key to encrypt the message:

1. Add Id and 'salt' to the encrypted message. Salt represents random characters inserted to make encryption harder to crack.
2. Message is sent to User2, who will tally its own Private key with salt of message to decode it.

But in the case of ABE, we instead use attributes of User2 to generate the salt. Let's take an example:

Let User2's Attributes be:

A. Belongs to India
B. Is 25 years old
C. Is male

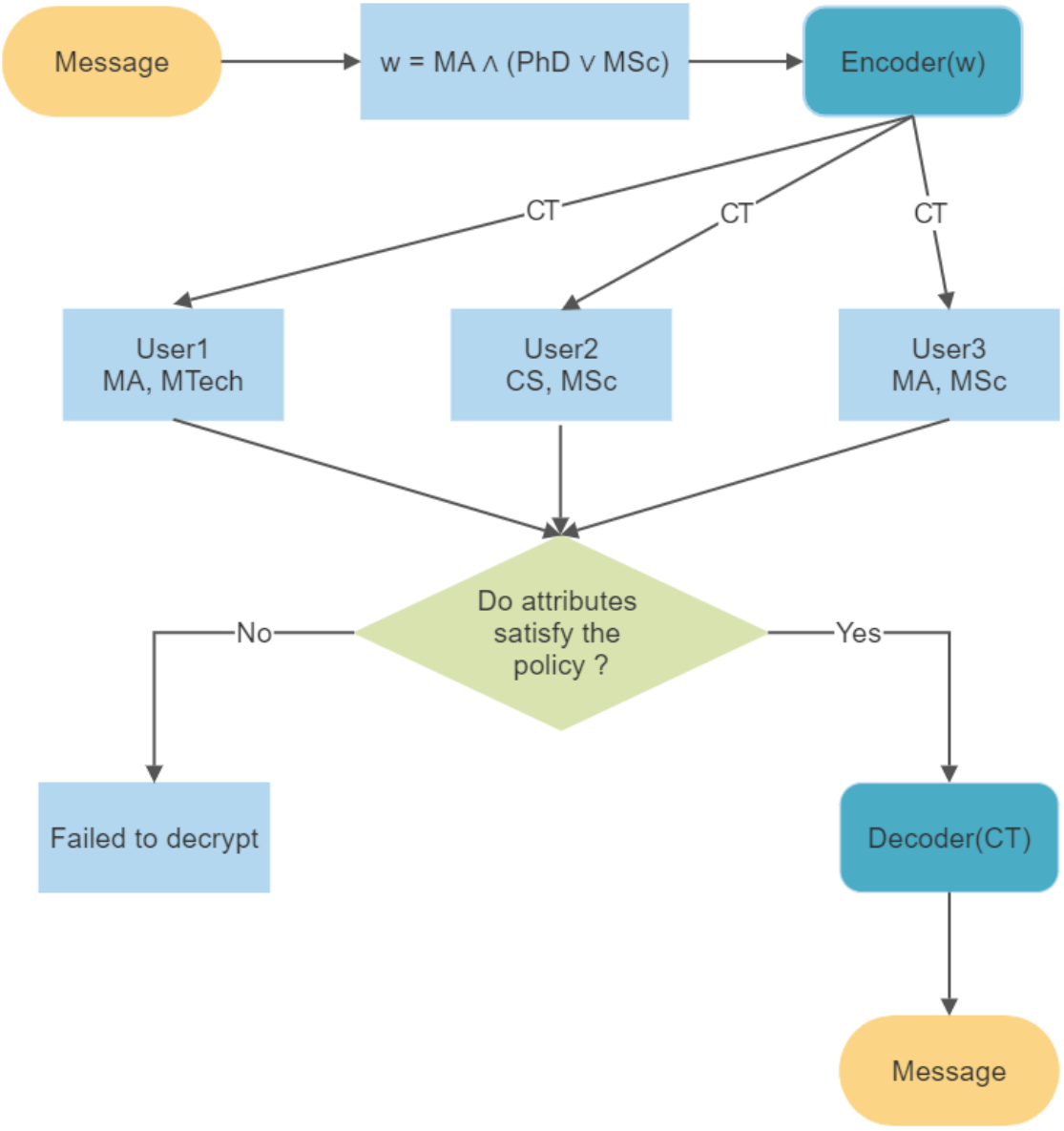
D. Date of Birth is 14/02/1997

Using these attributes, we can create an access policy. Suppose our access policy is given as:

{ A ∧ B } V { D }

This means that a user will only decrypt those messages whose salt satisfies the above condition. This is an application of cipher text policy. Another policy followed by ABE is the ‘key policy’, in which the attributes are used to describe encrypted data and policies are built into user’s keys.

Flow Diagram:



Implementation:

The formula used in Attribute based encryption will be one based on functional encryption algorithms:

- (pk, msk) <-- Setup: creates public key pk and master secret key msk
- sk <-- Keygen(msk, f): uses msk to generate new secret key sk for function f
- c <-- Enc(pk, x): uses pk to encrypt a message x
- y <-- Dec(sk, c): uses sk to calculate y = f(x), where x is the value c encrypts

Mathematically, ABE is based on a bilinear map, which is a function that maps pairs of elements from two groups to a third group while preserving certain algebraic properties. Let G and G' be two cyclic groups of prime order p, and let e: G x G -> G' be a bilinear map. ABE schemes use the bilinear map to construct a public key and a secret key for each user.

To encrypt a message M under a policy P, the ABE scheme first generates a random symmetric key k, and then computes the ciphertext C as follows:

C = (M * g^k) * h^s

where g and h are randomly chosen elements of G, and s is a secret vector that encodes the policy P. The secret vector s is chosen such that for any user with attributes A, the corresponding

secret key can decrypt the ciphertext if and only if A satisfies the policy P.

To decrypt the ciphertext C, a user with attributes A first computes the secret key sk_A as follows:

sk_A = prod_{i in A intersection s} d_i

where d_i is a randomly chosen element of G. The user can then recover the symmetric key k as follows:

k = e(C, sk_A)

Finally, the user can use k to decrypt the message M.

In summary, ABE uses a bilinear map to construct a flexible access control mechanism for encrypted data, where the access policy is associated with attributes rather than with specific users or groups. The encryption and decryption algorithms use the bilinear map to compute a ciphertext and a secret key that depend on the access policy and the user's attributes, respectively.

Analysis:

Attribute-based encryption (ABE) provides protection against several cybersecurity attacks, including:

1. *Data breaches:* ABE allows access control of encrypted data by associating access policies with attributes, rather than with specific users or groups. This makes it difficult for an attacker who gains unauthorized access to the data to decrypt it, as they would need to satisfy the access policy associated with the ciphertext.

2. *Insider attacks:* ABE enables fine-grained access control of data, which allows organizations to restrict access to sensitive data based on attributes such as job role or clearance level. This helps to prevent insider attacks where an authorized user with access to the data misuses it for personal gain or malicious purposes.
3. *Man-in-the-middle (MITM) attacks:* ABE uses public key cryptography to encrypt and decrypt data, which makes it resistant to MITM attacks where an attacker intercepts and modifies the data in transit. ABE also provides authenticity and integrity protection, which ensures that the encrypted data is not modified or tampered with during transmission.
4. *Phishing attacks:* ABE can be used to encrypt sensitive information such as login credentials or personal data, which can help to protect against phishing attacks. Even if an attacker obtains the encrypted data through phishing, they will not be able to decrypt it without satisfying the access policy associated with the ciphertext.

ABE is vulnerable to Chosen Plaintext Attacks (CPA) if the adversary has access to the encryption oracle and can choose arbitrary plaintext messages to be encrypted. To mitigate the risk of CPA, ABE schemes can be designed to include additional security features such as randomized ciphertexts, non-deterministic key generation, and message authentication codes (MACs). These features make it more difficult for an adversary to learn information about the encryption scheme and derive the secret key.

CP-ABE, the variant of ABE used in this project, gives protection against Chosen Ciphertext Attacks (CCA), since the ciphertext is associated with a policy that specifies the attributes required to

decrypt the message, rather than the set of attributes possessed by the user. In CP-ABE, the decryption algorithm checks whether the attributes possessed by the user satisfy the policy associated with the ciphertext, rather than directly using the user's secret key. This prevents an attacker from modifying the ciphertext in a way that would allow them to obtain the plaintext without satisfying the policy.

Overall, ABE is a powerful encryption mechanism that provides protection against various cybersecurity attacks by enabling fine-grained access control of data and using public key cryptography to ensure confidentiality, authenticity, and integrity of the encrypted data.

References:

- https://www.youtube.com/watch?v=95q9kgSoTJ8
- https://www.youtube.com/watch?v=ZogQMKzoQdw&t=69s
- https://www.ibm.com/docs/en/ztpf/1.1.0.15?topic=concepts-public-key-cryptography#s7pkey__figccs01
- https://www.cs.utexas.edu/~bwaters/publications/papers/cp-abe.pdf
- https://github.com/zeutro/openabe