QMAC from Graph States

Minh Thuy Truc Pham[1]
[1] *BTQ Technologies*

CONTENTS

I. GRAPH STATES

II. QUANTUM MESSAGE AUTHENTICATION CODE FROM GRAPH STATES

A. Message Authentication Code

Message Authentication Code (MAC), also referred to as "tag", is a piece of information sent along with a message from a specified sender to the receiver to ensure the **integrity** and **authenticity** of that message.

Formally, message authentication involves two parties, Alice and Bob, who share a secret key k related to a specific function f_k. When Alice wants to send a message m to Bob, she will compute the unique tag $t = f_k(m)$ associated with m and send the pair $(m, t = f_k(m))$ to Bob. When Bob receives the message-tag pair (m', t), he will recompute the authentication tag $f_k(m') = t'$ and check whether $t' = t$. If it matches, Alice's message m is authenticated. The security requirement of the protocol is that adversaries who do not know about the secret key k cannot create valid tags for messages they have never seen before.

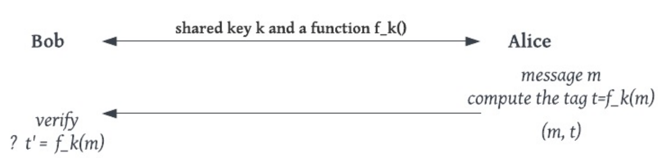

Figure 1: Message Authentication Code

B. HMAC

A Hash-based Message Authentication Code (HMAC) is a type of Message Authentication Code (MAC) that employs a hash function to combine the message being authenticated with a secret key. This results in a unique hash value, the message tag, which can only be replicated if both the message and the key are known.

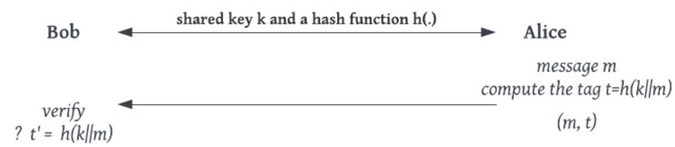

Figure 2: Hash-based Message Authentication Code

It is important to note that in a message authentication key and tag are different to the key and signature in digital signature schemes due to the needed security properties.

- Integrity. The receiver is confident that the message has not been modified.
- Authenticity. The receiver is confident that the message originates from the sender.
- Non-repudiation property is a property that ensures that a party (sender or receiver) cannot deny having sent or received a message.
- Distinction between a digital signature and message authentication schemes:

Table I: Comparison between the HMAC and Digital signature

	HMAC	Digital signature
Integrity	✓	✓
Authentication	✓	✓
Non-repudiation	✗	✓
Key	Symmetric	Asymmetric

C. Quantum Message Authentication Code (QMAC)

1. Graph structure

Let $\mathcal{X}_{k+n}$ be the set of all graphs G with $k+n$ vertices that satisfy:

1. The graph is separated into two parts: one side with k vertices V_M and the other with n vertices $V_{G'}$.
2. Each vertex m_i in V_M represents a bit of the message m, and no vertices are connected.
3. The subgraph G' has edges exists with probability $1/2$.
4. Let $V_i \subset V_{G'}$ be the set of vertices in the subgraph G' that have edges with m_i. The union of all V_i covers $V_{G'}$.

$$\bigcup_{i \in [k]} V_i = V'_G \tag{2.1}$$

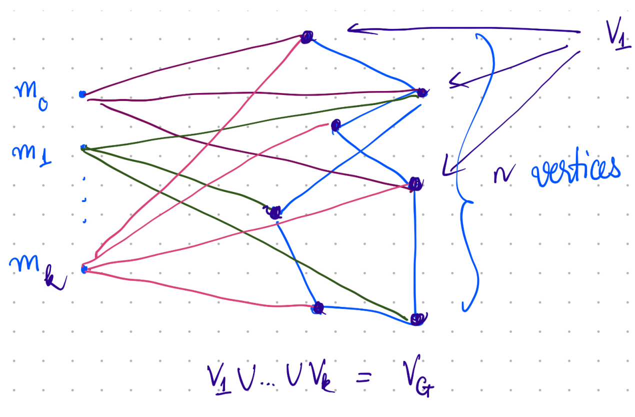

Figure 3: Graph structure of a graph $G \in \mathcal{X}_{k+n}$.

2. QMAC from graph states

The quantum message authentication code (QMAC) scheme for a message m of k bits can be described as follows:

- **KeyGen**: Alice samples uniformly at random a graph state $G' = (V, E) \in \mathcal{X}_{k+n}$ from the set $\mathcal{X}_{k+n}$ and sends it to Bob. Note that both parties know the whole graph structure and the shared secret key is $(G', \rho(G'))$.
- **Authentication**: Alice wants to authenticate a message m of k bits.

  If $m_i = 0$, Alice measures the corresponding qubit in the Z basis; otherwise, she measures it in the Y basis. The resulting graph is a graph G of n vertices. Alice now sends the graph state $\rho(G)$ as the tag of this message.

  $$(m, t = \rho(G)) \tag{2.2}$$

  Note that qubits measured in Z are eliminated while Y measurements perform an edge complement on the subgraph induced by the random subset of vertices V_i that qubit m_i is connected to.
- **Verification**: Receiving the message-tag pair from Alice, Bob now performs similar deletions and complementing of the graph structure according to the message on his secret key G' to obtain a graph G. With the knowledge of all stabilisers of the graph G, Bob then measures the "tag" state $\rho(G)$. If this yields the $+1$ outcome, Alice's message is authenticated.

3. Security Analysis

The QMAC from the graph states construction does not rely on any hardness assumptions. If an eavesdropper Eve $\mathcal{E}$ sees the message-tag pair $(m, \rho(G'))$, $\mathcal{E}$ cannot extract any information about the underlying graph G', and hence, cannot extract any information about the secret key G. Therefore, the protocol provides information-theoretic security.

It is important to note that multiple messages can be reduced to the same subgraph G'. However, the above QMAC protocol is still secure since QMAC does not require the non-repudiation property, only integrity and authentication.

Given the two message-tag pairs $(m, \rho(G'))$ and $(m', \rho(G''))$, is it possible for an adversary to extract information about the secret key G? Since the adversary knows G' and G'' can be obtained from G.

REFERENCES

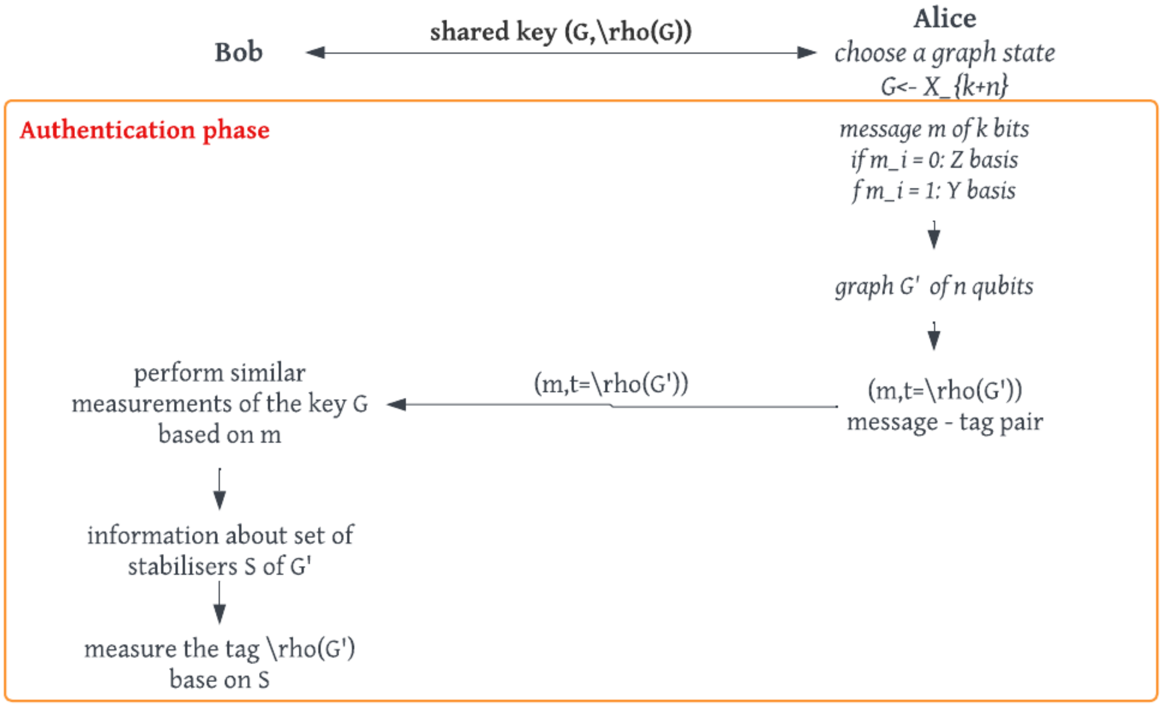

Figure 4: Quantum Message Authentication Code from graph states