1 Introduction

Message Authentication System In the scenario of processor-memory architecture, data integrity refers to maintaining the data on the transmission outside the chip and stored on off-chip memory untampered. If the data stored on off-chip memory , or some external data injected to the system S, we can say that the data integrity of S is attacked. If the system S can not examine the tampered data or injected data, the attack is assumed to be successful. According to our knowledge, there are two aspects of attacks on data integrity:

1. Introducing new content: The content of data in the system is modified, or data construct by the attacker is inserted to the system.

2. Replacing data with valid copy: The data D in transmission or on the memory is replace with the following two types of copy: A copy of D at an old time point or a copy of other data from different memory address.

The goal of integrity protection can be defined as the capability to examine whether the data to read by processor has been tampered. Cryptographic hash function and Message Authentication Code(MAC) scheme are common cryptographic primitives aiming to protect the data integrity of a system. These theoretical models for integrity protection are called Authentication Primitives(AP) in some research works.

Hash is a short message block computed by hash function with data block D as input. When processor writes a data block to a memory address, the data block D is sent to Hash function and a Hash value H1 is computed and stored on the chip. The hash value H is stored on chip following the order of data block on the memory. The message stored on the memory from chip is data block D alone. When the processor wants to read a data block D from memory, D is sent to hash function and an hash value H2 is computed. If H1 is identical to H2, then the data block D is assumed to be untampered and read by processor. Figure 3-a expresses functionality of hash function in integrity protection. If the integrity of a processor-memory system is protected by hash function, the possible attacks are:

1. Modify the content of a data block

2. Insert new data

3. Replace the content of a data block with a copy of data from other address

If using hash function to protect integrity, the three attacks in the above list can be defended if the hash function behave like an theoretical random function, which means for any input data block D, its hash value H is randomly assigned.

Message Authentication Code(MAC), called tag in some research works, is a short message block. A tag is generated by a MAC scheme by accepting a data block D as input and process D with a secret key K inside the scheme. The MAC schemes require only secrete key to process inputs are denoted as deterministic

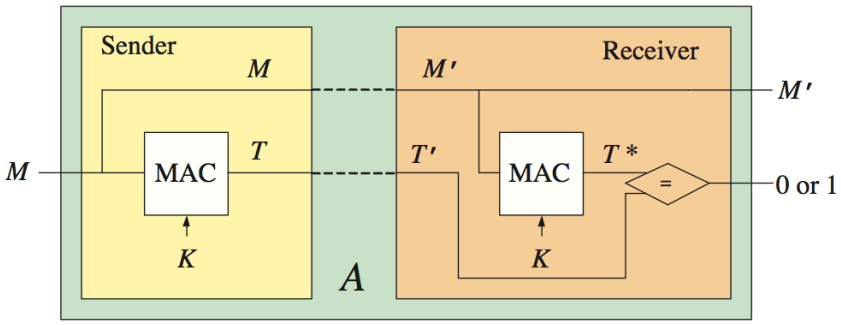

Figure 1: The Concept of Deterministic MAC Scheme

MAC schemes in some research works. Some MAC schemes require an additional input block, named nonce, in the generation of tag. Nonce is used to provide uniqueness for each input invoking the MAC scheme. MAC schemes with nonce as additional input are called state MAC schemes in some research works. When the processor writes a data block D to the memory, MAC scheme computes a tag and concatenate the tag withe the data forming a data-tag pair. The data-tag pair is sent to off-chip memory for storage. When the processor reads a data from memory, the data-pair is sent to the chip and to MAC scheme. Assume the data part is D and tag part is T1. MAC scheme compute a tag T2 using D T2 is compared with T1. If T1 is identical to T2, D is assumed to be untampered. Figure 3-b expresses the functionality of MAC scheme in integrity protection. If the integrity of a processor-memory system is protected by MAC scheme, the possible attacks are:

1. Modify the content of a data-tag pair
2. Insert new data-tag pair
3. Replace the content of a data-tag pair with a copy of data-tag pair from other address
4. Replace the content of a data-tag pair with a copy of this pair in an old time point

If a MAC scheme is capable to defend the 1st and 2nd attacks in the list, it should ensure that for any data block D, the tag is randomly assigned. If this scheme is capable to defend the 3rd and 4th attacks too, it should ensure for any two identical data blocks D1 and D2 that are written to memory on different time or to different addresses, their tags T1 and T2 should be randomly generated.

1.1 Security Analysis on Message Authentication Systems

If the resources, such as computation time, is unlimited for the attacker, and AP system can always be broken at some time. Research works on security analysis on AP system try quantify the evaluation result under the assumption that the attack has limitation on computation resources. According to our knowledge, there have been three categories of security evaluation mechanisms designed, namely computational model based on provable security theory, dolev-yao model based on formal methods theory and automatic security analysis framework combining the advantages from previous two models.

Symbolic Model based Security Analysis

Computational Sound Security Analysis based on Symbolic Model

Computational Model based Security Analysis

Code-based Game-Playing Techniques In cryptography, game-playing proof represents the techique that abstracts the interaction between the adversary and environment to a program named game. Computing the probability of adversary becomes the stepwise refinement of a sequences of games. Code-based game-playing proofs model the games with programing language, or code.