10

Authentication Codes

10.1 Introduction

We have spent a considerable amount of time studying cryptosystems, which are used to obtain secrecy. An authentication code provides a method of ensuring the *integrity* of a message, i.e., that the message has not been tampered with and that it originated with the presumed transmitter. Our goal is to achieve this authentication capability even in the presence of an active opponent, Oscar, who can observe messages in the channel and introduce messages of his own choosing into the channel. This goal is accomplished in the "private-key" setting whereby Alice and Bob share a secret key, K, before any message is transmitted.

In this chapter, we study codes that provide authentication but no secrecy. In such a code, a key is used to compute an authentication tag which will enable Bob to check the authenticity of the message he receives. Another application of an authentication code is verify that data in a large file has not been tampered with. An authentication tag would be stored with the data; the key used to generate and verify the authenticator would be stored separately, in a "secure" area.

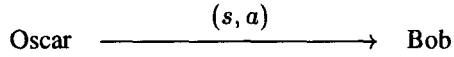
We should also point out that, in many respects, an authentication code is similar to a signature scheme or to a message authentication code (MAC). The main differences are as follows: The security of an authentication code is unconditional, whereas signature schemes and MACs are studied from the point of view of computational security. Also, when an authentication code (or a MAC) is used, a message can be verified only by the intended receiver. In comparison, anyone can verify a signature using a public verification algorithm.

We now give a formal definition of the terminology we use in the study of authentication codes.

DEFINITION 10.1 *An* ***authentication code*** *is a four-tuple* $(\mathcal{S}, \mathcal{A}, \mathcal{K}, \mathcal{E})$, *where the following conditions are satisfied:*

1. $\mathcal{S}$ *is a finite set of possible* ***source states***

FIGURE 10.1
Impersonation by Oscar



2. *$\mathcal{A}$ is a finite set of possible* ***authentication tags***
3. *$\mathcal{K}$, the* ***keyspace****, is a finite set of possible* ***keys***
4. *For each $K \in \mathcal{K}$, there is an* ***authentication rule*** *$e_K : \mathcal{S} \to \mathcal{A}$.*

The ***message set*** *is defined to be $\mathcal{M} = \mathcal{S} \times \mathcal{A}$.*

REMARK Note that a source state is analogous to a plaintext. A message consists of a plaintext with an appended authentication tag; it could be more precisely referred to as a *signed message*. Also, an authentication rule need not be an injective function. ▮

In order to transmit a (signed) message, Alice and Bob follow the following protocol. First, they jointly choose a random key $K \in \mathcal{K}$. This is done in secret, as in a private-key cryptosystem. At a later time, suppose that Alice wants to communicate a source state $s \in \mathcal{S}$ to Bob over an insecure channel. Alice computes $a = e_K(s)$ and sends the message (s, a) to Bob. When Bob receives (s, a), he computes $a' = e_K(s)$. If $a' = a$, then he accepts the message as authentic; otherwise, he rejects it.

We will study two different types of attacks that Oscar might carry out. In both of these attacks, Oscar is an "intruder-in-the-middle." These attacks described are as follows:

Impersonation

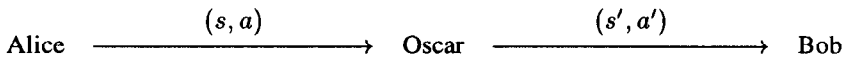
Oscar introduces a message (s, a) into the channel, hoping to have it accepted as authentic by Bob. This is depicted in Figure 10.1.

Substitution

Oscar observes a message (s, a) in the channel, and then changes it to (s', a'), where $s' \neq s$, again hoping to have it accepted as authentic by Bob. Hence, he is hoping to mislead Bob as to the source state. This is depicted in Figure 10.2.

Associated with each of these attacks is a *deception probability*, which represents the probability that Oscar will successfully deceive Bob, if he (Oscar) follows an optimal strategy. These probabilities are denoted by Pd_0 (impersonation) and Pd_1 (substitution). In order to compute Pd_0 and Pd_1, we need to specify probability distributions on $\mathcal{S}$ and $\mathcal{K}$. These will be denoted by $p_{\mathcal{S}}$ and $p_{\mathcal{K}}$, respectively.

FIGURE 10.2
Substitution by Oscar



We assume that the authentication code and these two probability distributions are known to Oscar. The only information that Alice and Bob possess that is not known to Oscar is the value of the key, K. This is analogous to the way that we studied the unconditional security of private-key cryptosystems.

10.2 Computing Deception Probabilities

In this section, we look at the computation of deception probabilities. We begin with a small example of an authentication code.

Example 10.1
Suppose

$$\mathcal{S} = \mathcal{A} = \mathbb{Z}_3$$

and

$$\mathcal{K} = \mathbb{Z}_3 \times \mathbb{Z}_3.$$

For each $(i, j) \in \mathcal{K}$ and each $s \in \mathcal{S}$, define

$$e_{ij}(s) = is + j \bmod 3.$$

It will be useful to study the *authentication matrix*, which tabulates all the values $e_{ij}(s)$. For each key $K \in \mathcal{K}$ and for each $s \in \mathcal{S}$, place the authentication tag $e_K(s)$ in row K and column s of a $|\mathcal{K}| \times |\mathcal{S}|$ matrix M. The array M is presented in Figure 10.3.

Suppose that the key is chosen at random, i.e., $p_{\mathcal{K}}(K) = 1/9$ for each $K \in \mathcal{K}$. We do not specify the probability distribution $p_{\mathcal{S}}$ since it turns out to be immaterial in this example.

Let's first consider an impersonation attack. Oscar will pick a source state s, and attempt to guess the "correct" authentication tag. Denote by K_0 the actual key being used (which is unknown to Oscar). Oscar will succeed in deceiving Bob if he guesses the tag $a_0 = e_{K_0}(s)$. However, for any $s \in \mathcal{S}$ and $a \in \mathcal{A}$, it is easy to verify that there are exactly three (out of nine) authentication rules $K \in \mathcal{K}$ such

FIGURE 10.3
An authentication matrix

key	0	1	2
(0,0)	0	0	0
(0,1)	1	1	1
(0,2)	2	2	2
(1,0)	0	1	2
(1,1)	1	2	0
(1,2)	2	0	1
(2,0)	0	2	1
(2,1)	1	0	2
(2,2)	2	1	0

that $e_K(s) = a$. (In other words, each symbol occurs three times in each column of the authentication matrix.) Hence, it follows that $Pd_0 = 1/3$.

Substitution is a bit more complicated to analyze. As a specific case, suppose Oscar observes the message $(0,0)$ in the channel. This does give Oscar some information about the key: he now knows that

$$K_0 \in \{(0,0),(1,0),(2,0)\}.$$

Now suppose Oscar replaces the message $(0,0)$ with the message $(1,1)$. Then, he will succeed in his deception if and only if $K_0 = (1,1)$. The probability that K_0 is the key is $1/3$, since the key is known to be in the set $\{(0,0),(1,0),(2,0)\}$.

A similar analysis can be done for any substitution that Oscar might make. In general, if Oscar observes the message (s,a), and replaces it with any message (s',a') where $s' \neq s$, then he deceives Bob with probability $1/3$. We can see this as follows. Observation of (s,a) restricts the key to one of three possibilities. Then, for each choice of (s',a'), there is one key (out of the three possible keys) under which a' is the authentication tag for s'. □

Let's now discuss how to compute the deception probabilities in general. First, we consider Pd_0. As above, let K_0 denote the key chosen by Alice and Bob. For $s \in \mathcal{S}$ and $a \in \mathcal{A}$, define $payoff(s,a)$ to be the probability that Bob will accept the message (s,a) as being authentic. It is not difficult to see that

$$\begin{aligned} payoff(s,a) &= prob(a = e_{K_0}(s)) \\ &= \sum_{\{K \in \mathcal{K}: e_K(s) = a\}} p_{\mathcal{K}}(K). \end{aligned}$$

That is, $payoff(s, a)$ is computing by selecting the rows of the authentication matrix that have entry a in column s, and summing the probabilities of the corresponding keys.

In order to maximize his chance of success, Oscar will choose (s, a) such that $payoff(s, a)$ is a maximum. Hence,

$$Pd_0 = \max\{payoff(s, a) : s \in \mathcal{S}, a \in \mathcal{A}\}. \tag{10.1}$$

Note that Pd_0 does not depend on the probability distribution $p_{\mathcal{S}}$.

Pd_1 is more difficult to compute, and it may depend on the probability distribution $p_{\mathcal{S}}$. Let's first consider the following problem: Suppose Oscar observes the message (s, a) in the channel. Oscar will substitute some (s', a') for (s, a), where $s' \neq s$. Hence, for $s, s' \in \mathcal{S}$, $s \neq s'$, and $a, a' \in \mathcal{A}$, we define $payoff(s', a'; s, a)$ to be the probability that a substitution of (s, a) with (s', a') will succeed in deceiving Bob. Then we can compute the following:

$$\begin{aligned} payoff(s', a'; s, a) &= prob(a' = e_{K_0}(s') | a = e_{K_0}(s)) \\ &= \frac{prob(a' = e_{K_0}(s') \wedge a = e_{K_0}(s))}{prob(a = e_{K_0}(s))} \\ &= \frac{\displaystyle\sum_{\{K \in \mathcal{K} : e_K(s) = a, e_K(s') = a'\}} p_{\mathcal{K}}(K)}{\displaystyle\sum_{\{K \in \mathcal{K} : e_K(s) = a\}} p_{\mathcal{K}}(K)} \\ &= \frac{\displaystyle\sum_{\{K \in \mathcal{K} : e_K(s) = a, e_K(s') = a'\}} p_{\mathcal{K}}(K)}{payoff(s, a)}. \end{aligned}$$

The numerator of this fraction is found by selecting the rows of the authentication matrix that have the value a in column s and the value a' in column s', and summing the probabilities of the corresponding keys.

Since Oscar wants to maximize his chance of deceiving Bob, he will compute

$$p_{s,a} = \max\{payoff(s', a'; s, a) : s' \in \mathcal{S}, s \neq s', a' \in \mathcal{A}\}.$$

The quantity $p_{s,a}$ denotes the probability that Oscar can deceive Bob with a substitution, given that (s, a) is the message observed in the channel.

Now, how do we compute the deception probability Pd_1? Evidently, we have to compute a weighted average of the quantities $p_{s,a}$ with respect to the probabilities $p_{\mathcal{M}}(s, a)$ of observing messages (s, a) in the channel. That is, we calculate Pd_1 to be

$$Pd_1 = \sum_{(s,a) \in \mathcal{M}} p_{\mathcal{M}}(s, a) p_{s,a}. \tag{10.2}$$

FIGURE 10.4
An authentication matrix

key	1	2	3	4
1	1	1	1	2
2	2	2	1	2
3	1	2	2	1

The probability distribution $p_{\mathcal{M}}$ is as follows:

$$\begin{aligned} p_{\mathcal{M}}(s,a) &= p_{\mathcal{S}}(s) \times p_{\mathcal{K}}(a|s) \\ &= p_{\mathcal{S}}(s) \times \sum_{\{K \in \mathcal{K}: e_K(s)=a\}} p_{\mathcal{K}}(K) \\ &= p_{\mathcal{S}}(s) \times payoff(s,a). \end{aligned}$$

In Example 10.1,

$$payoff(s,a) = 1/3$$

for all s, a, so $Pd_0 = 1/3$. Also, it can be checked that

$$payoff(s',a';s,a) = 1/3$$

for all s, s', a, a', $s \neq s'$. Hence, $Pd_1 = 1/3$ for any probability distribution $p_{\mathcal{S}}$. (In general, though, Pd_1 will depend on $p_{\mathcal{S}}$.)

Let's look at the computation of Pd_0 and Pd_1 for a less "regular" example.

Example 10.2
Consider the authentication matrix of Figure 10.4. Suppose the probability distributions on $\mathcal{S}$ and $\mathcal{K}$ are

$$p_{\mathcal{S}}(i) = 1/4,$$

$1 \leq i \leq 4$; and

$$p_{\mathcal{K}}(1) = 1/2, p_{\mathcal{K}}(2) = p_{\mathcal{K}}(3) = 1/4.$$

The values $payoff(s,a)$ are as follows:

$$payoff(1,1) = 3/4$$
$$payoff(1,2) = 1/4$$

$$payoff(2,1) = 1/2$$
$$payoff(2,2) = 1/2$$
$$payoff(3,1) = 3/4$$
$$payoff(3,2) = 1/4$$
$$payoff(4,1) = 1/4$$
$$payoff(4,2) = 3/4.$$

Hence, $Pd_0 = 3/4$. Oscar's optimal impersonation strategy is to place any of the messages $(1,1)$, $(3,1)$ or $(4,2)$ into the channel.

Now we turn to the computation of Pd_1. First, we present the various values $payoff(s', a'; s, a)$ in the form of a matrix. The entry in row (s, a) and column (s', a') is the value $payoff(s', a'; s, a)$.

	$(1,1)$	$(1,2)$	$(2,1)$	$(2,2)$	$(3,1)$	$(3,2)$	$(4,1)$	$(4,2)$
$(1,1)$			2/3	1/3	2/3	1/3	1/3	2/3
$(1,2)$			0	1	1	0	1	0
$(2,1)$	1	0			0	1	0	1
$(2,2)$	1/2	1/2			1/2	1/2	1/2	1/2
$(3,1)$	2/3	1/3	2/3	1/3			0	1
$(3,2)$	1	0	0	1			1	0
$(4,1)$	1	0	0	1	0	1		
$(4,2)$	2/3	1/3	2/3	1/3	1	0		

Thus we have $p_{1,1} = 2/3$, $p_{2,2} = 1/2$, and $p_{s,a} = 1$ for all other s, a. It is then a simple matter to evaluate $Pd_1 = 7/8$. An optimal substitution strategy for Oscar is as follows:

$$(1,1) \to (2,1)$$
$$(1,2) \to (2,2)$$
$$(2,1) \to (1,1)$$
$$(2,2) \to (1,1)$$
$$(3,1) \to (4,2)$$
$$(3,2) \to (1,1)$$
$$(4,1) \to (1,1)$$
$$(4,2) \to (3,1).$$

This strategy indeed yields $Pd_1 = 7/8$. □

The computation of Pd_1 in Example 10.2 is straightforward but lengthy. We can in fact simplify the computation of Pd_1 by observing that we divide by the quantity $payoff(s,a)$ in the computation of $p_{s,a}$, and then later multiply by $payoff(s,a)$ in the computation of Pd_1. Of course, these two operations cancel each other out. Suppose we define

$$q_{s,a} = \max \left\{ \sum_{\{K \in \mathcal{K}: e_K(s)=a, e_K(s')=a'\}} p_{\mathcal{K}}(K) : s' \in \mathcal{S}, s' \neq s, a' \in \mathcal{A} \right\}$$

for all s, a. Then we have the following more concise formula for Pd_1:

$$Pd_1 = \sum_{(s,a) \in \mathcal{M}} p_{\mathcal{S}}(s) q_{s,a}. \tag{10.3}$$

10.3 Combinatorial Bounds

We have seen that the security of an authentication code is measured by the deception probabilities. Hence, we want to construct codes so that these probabilities are as small as possible. But other considerations are also important. Let's consider the various objectives that we might strive for in an authentication code:

1. The deception probabilities Pd_0 and Pd_1 must be small enough to obtain the desired level of security.
2. The number of source states must be large enough so that we can communicate the desired information by appending an authentication tag to one source state.
3. The size of the key space should be minimized, since the value of the key must be communicated over a secure channel. (Note that the key must be changed every time a message is communicated, as is done with the **One-time Pad**.)

In this section, we determine lower bounds on the deception probabilities, which will be computed in terms of other parameters of the code. Recall that we have defined an authentication code to consist of a four-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{K}, \mathcal{E})$. Throughout this section, we will denote $|\mathcal{A}| = \ell$.

Suppose we fix a source state $s \in \mathcal{S}$. Then we can compute:

$$\sum_{a \in \mathcal{A}} payoff(s,a) = \sum_{a \in \mathcal{A}} \sum_{\{K \in \mathcal{K}: e_K(s)=a\}} p_{\mathcal{K}}(K)$$

$$= \sum_{K \in \mathcal{K}} p_{\mathcal{K}}(K)$$

$$= 1.$$

Hence, for every $s \in \mathcal{S}$, there exists an authentication tag $a(s)$ such that

$$payoff(s, a(s)) \geq \frac{1}{\ell}.$$

The following theorem follows easily.

THEOREM 10.1
Suppose $(\mathcal{S}, \mathcal{A}, \mathcal{K}, \mathcal{E})$ is an authentication code. Then $Pd_0 \geq 1/\ell$, where $\ell = |\mathcal{A}|$. Further, $Pd_0 = 1/\ell$ if and only if

$$\sum_{\{K \in \mathcal{K}: e_K(s) = a\}} p(K) = \frac{1}{\ell} \tag{10.4}$$

for every $s \in \mathcal{S}$, $a \in \mathcal{A}$.

Now, we turn our attention to substitution. Suppose we fix s, a and s', where $s' \neq s$. Then we have the following:

$$\begin{aligned}\sum_{a' \in \mathcal{A}} payoff(s', a'; s, a) &= \sum_{a' \in \mathcal{A}} \frac{\sum_{\{K \in \mathcal{K}: e_K(s) = a, e_K(s') = a'\}} p_{\mathcal{K}}(K)}{\sum_{\{K \in \mathcal{K}: e_K(s) = a\}} p_{\mathcal{K}}(K)} \\ &= \frac{\sum_{\{K \in \mathcal{K}: e_K(s) = a\}} p_{\mathcal{K}}(K)}{\sum_{\{K \in \mathcal{K}: e_K(s) = a\}} p_{\mathcal{K}}(K)} \\ &= 1.\end{aligned}$$

So, there exists an authentication tag $a'(s', s, a)$ such that

$$payoff(s', a'(s', s, a); s, a) \geq \frac{1}{\ell}.$$

The next theorem follows as a consequence.

THEOREM 10.2
Suppose $(\mathcal{S}, \mathcal{A}, \mathcal{K}, \mathcal{E})$ *is an authentication code. Then* $Pd_1 \geq 1/\ell$, *where* $\ell = |\mathcal{A}|$. *Further,* $Pd_1 = 1/\ell$ *if and only if*

$$\frac{\sum_{\{K \in \mathcal{K}: e_K(s)=a, e_K(s')=a'\}} p_{\mathcal{K}}(K)}{\sum_{\{K \in \mathcal{K}: e_K(s)=a\}} p_{\mathcal{K}}(K)} = \frac{1}{\ell} \tag{10.5}$$

for every $s, s' \in \mathcal{S}, s' \neq s$, $a, a' \in \mathcal{A}$.

PROOF We have

$$\begin{aligned} Pd_1 &= \sum_{(s,a) \in \mathcal{M}} p_{\mathcal{M}}(s,a) p_{s,a} \\ &\geq \sum_{(s,a) \in \mathcal{M}} \frac{p_{\mathcal{M}}(s,a)}{\ell} \\ &= \frac{1}{\ell}. \end{aligned}$$

Further, equality occurs if and only if $p_{s,a} = 1/\ell$ for every (s,a). But this is in turn equivalent to the condition that *payoff*$(s', a'; s, a) = 1/\ell$ for every (s,a). ∎

Combining Theorems 10.1 and 10.2, we get the following:

THEOREM 10.3
Suppose $(\mathcal{S}, \mathcal{A}, \mathcal{K}, \mathcal{E})$ *is an authentication code, where* $\ell = |\mathcal{A}|$. *Then* $Pd_0 = Pd_1 = 1/\ell$ *if and only if*

$$\sum_{\{K \in \mathcal{K}: e_K(s)=a, e_K(s')=a'\}} p_{\mathcal{K}}(K) = \frac{1}{\ell^2} \tag{10.6}$$

for every $s, s' \in \mathcal{S}, s' \neq s$, $a, a' \in \mathcal{A}$.

PROOF Equations (10.4) and (10.5) imply Equation (10.6). Conversely, Equation (10.6) implies Equations (10.4) and (10.5). ∎

If the keys are equiprobable, then we obtain the following corollary:

FIGURE 10.5
An OA(3, 3, 1)

$$\begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 2 & 2 & 2 \\ 0 & 1 & 2 \\ 1 & 2 & 0 \\ 2 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 0 & 2 \\ 2 & 1 & 0 \end{pmatrix}$$

COROLLARY 10.4
Suppose $(\mathcal{S}, \mathcal{A}, \mathcal{K}, \mathcal{E})$ is an authentication code where $\ell = |\mathcal{A}|$, and keys are chosen equiprobably. Then $Pd_0 = Pd_1 = 1/\ell$ if and only if

$$|\{K \in \mathcal{K} : e_K(s) = a, e_K(s') = a'\}| = \frac{|\mathcal{K}|}{\ell^2}, \tag{10.7}$$

for every $s, s' \in \mathcal{S}, s' \neq s$, $a, a' \in \mathcal{A}$.

10.3.1 Orthogonal Arrays

In this section, we look at the connections between authentication codes and certain combinatorial structures called orthogonal arrays. First, we give a definition.

DEFINITION 10.2 *An **orthogonal array** $\mathrm{OA}(n, k, \lambda)$ is a $\lambda n^2 \times k$ array of n symbols, such that in any two columns of the array every one of the possible n^2 pairs of symbols occurs in exactly λ rows.*

Orthogonal arrays are well-studied structures in combinatorial design theory, and are equivalent to other structures such as transversal designs, mutually orthogonal Latin squares and nets.

In Figure 10.5, we present an orthogonal array $\mathrm{OA}(3, 3, 1)$ which is obtained from the authentication matrix of Figure 10.3. Any orthogonal array $\mathrm{OA}(n, k, \lambda)$ can be used to construct an authentication code with $Pd_0 = Pd_1 = 1/n$, as stated in the following theorem.

THEOREM 10.5
Suppose there is an orthogonal array $\mathrm{OA}(n, k, \lambda)$. Then there is an authentication code $(\mathcal{S}, \mathcal{A}, \mathcal{K}, \mathcal{E})$, where $|\mathcal{S}| = k$, $|\mathcal{A}| = n$, $|\mathcal{K}| = \lambda n^2$ and $Pd_0 = Pd_1 = 1/n$.

PROOF Use each row of the orthogonal array as an authentication rule with equal probability $1/(\lambda n^2)$. The correspondences are as follows:

orthogonal array	authentication code
row	authentication rule
column	source state
symbol	authentication tag

Since Equation (10.7) is satisfied, we can apply Corollary 10.4, obtaining a code with the stated properties. ∎

10.3.2 Constructions and Bounds for OAs

Suppose that we construct an authentication code from an $\mathrm{OA}(n, k, \lambda)$. The parameter n determines the number of authenticators (i.e., the security of the code), while the parameter k determines the number of source states the code can accommodate. The parameter λ relates only to the number of keys, which is λn^2. Of course, the case $\lambda = 1$ is most desirable, but we will see that it is sometimes necessary to use orthogonal arrays with higher values of λ.

Suppose we want to construct an authentication code with a specified source set $\mathcal{S}$, and a specified security level ϵ (i.e., so that $Pd_0 \leq \epsilon$ and $Pd_1 \leq \epsilon$). An appropriate orthogonal array will satisfy the following conditions:

1. $n \geq 1/\epsilon$
2. $k \geq |\mathcal{S}|$ (observe that we can always delete one or more columns from an orthogonal array and the resulting array is still an orthogonal array, so we do not require $k = |\mathcal{S}|$)
3. λ is minimized, subject to the two previous conditions being satisfied.

Let's first consider orthogonal arrays with $\lambda = 1$. For a given value of n, we are interested in maximizing the number of columns. Here is a necessary condition for existence:

THEOREM 10.6
Suppose there exists an $\mathrm{OA}(n, k, 1)$. *Then* $k \leq n + 1$.

PROOF Let A be an $\mathrm{OA}(n, k, 1)$ on symbol set $X = \{0, 1, \ldots, n-1\}$. Suppose π is a permutation of X, and we permute the symbols in any column of A according to the permutation π. The result is again an $\mathrm{OA}(n, k, 1)$. Hence, by applying a succession of permutations of this type, we can assume without loss of generality that the first row of A is $(00\ldots0)$.

We next show that each symbol must occur exactly n times in each column of A. Choose two columns, say c and c', and let x be any symbol. Then for each symbol x', there is a unique row of A in which x occurs in column c and x' occurs

in column c'. Letting x' vary over X, we see that x occurs exactly n times in column c.

Now, since the first row is $(00\ldots0)$, we have exhausted all occurrences of ordered pairs $(0,0)$. Hence, no other row contains more than one occurrence of 0. Now, let us count the number of rows containing at least one 0: the total is $1 + k(n-1)$. But this total cannot exceed the total number of rows in A, which is n^2. Hence, $1 + k(n-1) \leq n^2$, so $k \leq n+1$, as desired. ∎

We now present a construction for orthogonal arrays with $\lambda = 1$ in which $k = n$. This is, in fact, the construction that was used to obtain the orthogonal array presented in Figure 10.5.

THEOREM 10.7

Suppose p is prime. Then there exists an orthogonal array $\mathrm{OA}(p, p, 1)$.

PROOF The array will be a $p^2 \times p$ array, where the rows are indexed by $\mathbb{Z}_p \times \mathbb{Z}_p$ and the columns are indexed by $\mathbb{Z}_p$. The entry in row (i,j) and column x is defined to be $ix + j \bmod p$.

Suppose we choose two columns, x, y, $x \neq y$, and two symbols a, b. We want to find a (unique) row (i,j) such that a occurs in column x and b occurs in column y of row (i,j). Hence, we want to solve the two equations

$$a = ix + j$$
$$b = iy + j$$

for the unknowns i and j (where all arithmetic is done in the field $\mathbb{Z}_p$). But this system has the unique solution

$$i = (a-b)(x-y)^{-1} \bmod p$$
$$j = a - ix \bmod p.$$

Hence, we have an orthogonal array. ∎

We remark that any $\mathrm{OA}(n, n, 1)$ can be extended by one column to form an $\mathrm{OA}(n, n+1, 1)$ (see the Exercises).. Hence, using Theorem 10.7, we can obtain an infinite class of OA's that meet the bound of Theorem 10.6 with equality.

Theorem 10.6 tells us that $\lambda > 1$ if $k > n+1$. We will prove a more general result that places a lower bound on λ as a function of n and k. First, however, we derive an important inequality that we will use in the proof.

LEMMA 10.8
Suppose $b_1, \ldots, b_m$ are real numbers. Then

$$m\sum_{i=1}^{m} b_i^2 \geq \left(\sum_{i=1}^{m} b_i\right)^2.$$

PROOF Apply Jensen's Inequality (Theorem 2.5) with $f(x) = -x^2$ and $a_i = 1/m$, $1 \leq i \leq m$. The function f is continuous and concave, so we obtain

$$-\sum_{i=1}^{m} \frac{b_i^2}{m} \leq -\left(\sum_{i=1}^{m} \frac{b_i}{m}\right)^2,$$

which simplifies to give the desired result. ∎

THEOREM 10.9
Suppose there exists an $\mathrm{OA}(n, k, \lambda)$. *Then*

$$\lambda \geq \frac{k(n-1)+1}{n^2}.$$

PROOF Let A be an $\mathrm{OA}(n, k, \lambda)$ on symbol set $X = \{0, 1, \ldots, n-1\}$, where, without loss of generality, the first row of A is $(00\ldots0)$ (as in Theorem 10.6).

Let us denote the set of rows of A by $\mathcal{R}$, let r_1 denote the first row, and let $\mathcal{R}_1 = \mathcal{R}\backslash\{r_1\}$. For any row r of A, denote by x_r the number of occurrences of 0 in row r. It is easy to count the total number of occurrences of 0 in $\mathcal{R}_1$. Since each symbol must occur exactly λn times in each column of A, we have that

$$\sum_{r\in\mathcal{R}_1} x_r = k(\lambda n - 1).$$

Now, the number of times the ordered pair $(0, 0)$ occurs in rows in $\mathcal{R}_1$ is

$$\begin{aligned}\sum_{r\in\mathcal{R}_1} x_r(x_r - 1) &= \sum_{r\in\mathcal{R}_2} x_r^2 - \sum_{r\in\mathcal{R}_2} x_r \\ &= \sum_{r\in\mathcal{R}_2} x_r^2 - k(\lambda n - 1).\end{aligned}$$

Applying Lemma 10.8, we obtain

$$\sum_{r\in\mathcal{R}_1} x_r^2 \geq \frac{(k(\lambda n-1))^2}{\lambda n^2 - 1},$$

and hence

$$\sum_{r\in\mathcal{R}_1} x_r(x_r - 1) \geq \frac{(k(\lambda n-1))^2}{\lambda n^2 - 1} - k(\lambda n - 1).$$

On the other hand, in any given pair of columns, the ordered pair $(0,0)$ occurs in exactly λ rows. Since there are $k(k-1)$ ordered pairs of columns, it follows that the exact number of occurrences of the ordered pair $(0,0)$ in rows in $\mathcal{R}_1$ is $(\lambda-1)k(k-1)$. We therefore have

$$(\lambda-1)k(k-1) \geq \frac{(k(\lambda n-1))^2}{\lambda n^2-1} - k(\lambda n-1),$$

and hence

$$\big((\lambda-1)k(k-1)+k(\lambda n-1)\big)\,(\lambda n^2-1) \geq (k(\lambda n-1))^2.$$

If we divide out a factor of k, we get

$$(\lambda k-k-\lambda+\lambda n)(\lambda n^2-1) \geq k(\lambda n-1)^2.$$

Expanding, we have

$$\lambda^2 k n^2 - \lambda k n^2 - \lambda^2 n^2 + \lambda^2 n^3 - \lambda k + k + \lambda - \lambda n \geq \lambda^2 k n^2 - 2\lambda k n + k.$$

This simplifies to give

$$-\lambda^2 n^2 + \lambda^2 n^3 \geq \lambda k n^2 + \lambda k - \lambda + \lambda n - 2\lambda k n,$$

or

$$\lambda^2(n^3-n^2) \geq \lambda(k(n-1)^2+n-1).$$

Finally, taking out a factor of $\lambda(n-1)$, we obtain

$$\lambda n^2 \geq k(n-1)+1,$$

which is the desired bound. ∎

Our next result establishes the existence of an infinite class of orthogonal arrays that meet the above bound with equality.

THEOREM 10.10

Suppose p is prime and $d \geq 2$ is an integer. Then there is an orthogonal array $\mathrm{OA}(p,(p^d-1)/(p-1),p^{d-2})$.

PROOF Denote by $(\mathbb{Z}_p)^d$ the vector space of all d–tuples over $\mathbb{Z}_p$. We will construct A, an $\mathrm{OA}(p,(p^d-1)/(p-1),p^{d-2})$ in which the rows and columns are indexed by certain vectors in $(\mathbb{Z}_p)^d$. The entries of A will be elements of $\mathbb{Z}_p$. The set of rows is defined to be $\mathcal{R}=(\mathbb{Z}_p)^d$; the set of columns is

$$\mathcal{C} = \{(c_1,\ldots,c_d) \in (\mathbb{Z}_p)^d : \exists j, 0 \leq j \leq d-1, c_1 = \ldots = c_j = 0, c_{j+1} = 1\}.$$

$\mathcal{R}$ consists of all vectors in $(\mathbb{Z}_p)^d$, so $|\mathcal{R}| = p^d$. $\mathcal{C}$ consists of all non-zero vectors that have the first non-zero coordinate equal to 1. Observe that

$$|\mathcal{C}| = \frac{p^d - 1}{p - 1},$$

and that no two vectors in $\mathcal{C}$ are scalar multiples of each other.

Now, for each $\overline{r} \in \mathcal{R}$ and each $\overline{c} \in \mathcal{C}$, define

$$A(\overline{r}, \overline{c}) = \overline{r} \cdot \overline{c},$$

where $\cdot$ denotes the inner product of two vectors (reduced modulo p).

We prove that A is the desired orthogonal array. Let $\overline{b}, \overline{c} \in \mathcal{C}$ be two distinct columns, and let $x, y \in \mathbb{Z}_p$. We will count the number of rows $\overline{r}$ such that $A(\overline{r}, \overline{b}) = x$ and $A(\overline{r}, \overline{c}) = y$. Denote $\overline{r} = (r_1, r_2, \ldots, r_d)$, $\overline{b} = (b_1, b_2, \ldots, b_d)$ and $\overline{c} = (c_1, c_2, \ldots, c_d)$. The two equations $\overline{r} \cdot \overline{b} = x$, $\overline{r} \cdot \overline{c} = y$ can be written as two linear equations in $\mathbb{Z}_p$:

$$b_1 r_1 + \ldots + b_d r_d = x$$
$$c_1 r_1 + \ldots + c_d r_d = y.$$

This is a system of two linear equations in the d unknowns $r_1, \ldots r_d$. Since $\overline{b}$ and $\overline{c}$ are not scalar multiples, the two equations are linearly independent. Hence, this system has a solution space of dimension $d - 2$. That is, the number of solutions (i.e., the number of rows in which x occurs in column $\overline{b}$ and y occurs in column $\overline{c}$) is p^{d-2}, as desired. ∎

Let's carry out a small example of this construction.

Example 10.3

Suppose we take $p = 2$, $d = 3$. Then we will construct an OA(2, 7, 2). We have

$$\mathcal{R} = \{000, 001, 010, 011, 100, 101, 110, 111\}$$

and

$$\mathcal{C} = \{001, 010, 011, 100, 101, 110, 111\}.$$

The orthogonal array in Figure 10.6 results. □

10.3.3 Characterizations of Authentication Codes

To this point, we have studied authentication codes obtained from orthogonal arrays. Then we looked at necessary existence conditions and constructions for orthogonal arrays. One might wonder whether there are better alternatives to the

FIGURE 10.6
An OA(2, 7, 2)

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

orthogonal array approach. However, two characterization theorems tell us that this is not the case if we restrict our attention to authentication codes in which the deception probabilities are as small as possible.

We first prove the following partial converse to Theorem 10.5:

THEOREM 10.11
Suppose $(\mathcal{S}, \mathcal{A}, \mathcal{K}, \mathcal{E})$ *is an authentication code where* $|\mathcal{A}| = n$ *and* $Pd_0 = Pd_1 = 1/n$. *Then* $|\mathcal{K}| \geq n^2$. *Further,* $|\mathcal{K}| = n^2$ *if and only if there is an orthogonal array* OA$(n, k, 1)$ *where* $|\mathcal{S}| = k$, *and* $p_{\mathcal{K}}(K) = 1/n^2$ *for every key* $K \in \mathcal{K}$.

PROOF Fix two (arbitrary) source states s and s', $s \neq s'$, and consider Equation (10.6). For each ordered pair (a, a') of authentication tags, define

$$\mathcal{K}_{a,a'} = \{K \in \mathcal{K} : e_K(s) = a, e_K(s') = a'\}.$$

Then $|\mathcal{K}_{a,a'}| > 0$ for every pair (a, a'). Also, the n^2 sets $\mathcal{K}_{a,a'}$ are disjoint. Hence, $|\mathcal{K}| \geq n^2$.

Now, suppose that $|\mathcal{K}| = n^2$. Then $|\mathcal{K}_{a,a'}| = 1$ for every pair (a, a'), and Equation (10.6) tells us that $p_{\mathcal{K}}(K) = 1/n^2$ for every key $K \in \mathcal{K}$.

It remains to show that the authentication matrix forms an orthogonal array OA$(n, k, 1)$. Consider the columns indexed by the source states s and s'. Since $|\mathcal{K}_{a,a'}| = 1$ for every (a, a'), we have every ordered pair occurring exactly once in these two columns. Since, s and s' are arbitrary, we see that every ordered pair occurs exactly once in any two columns. ∎

The following characterization is more difficult; we state it without proof.

THEOREM 10.12
Suppose $(\mathcal{S}, \mathcal{A}, \mathcal{K}, \mathcal{E})$ *is an authentication code where* $|\mathcal{A}| = n$ *and* $Pd_0 = Pd_1 = 1/n$. *Then* $|\mathcal{K}| \geq k(n-1) + 1$. *Further,* $|\mathcal{K}| = k(n-1) + 1$ *if and only if there*

is an orthogonal array $\mathrm{OA}(n, k, \lambda)$, *where* $|\mathcal{S}| = k$, $\lambda = (k(n-1)+1)/n^2$, *and* $p_{\mathcal{K}}(K) = 1/(k(n-1)+1)$ *for every key* $K \in \mathcal{K}$.

REMARK Notice that Theorem 10.10 provides an infinite class of orthogonal arrays that meet the bound of Theorem 10.12 with equality. ▮

10.4 Entropy Bounds

In this section, we use entropy techniques to obtain bounds on the deception probabilities. The first of these is a bound on Pd_0.

THEOREM 10.13
Suppose that $(\mathcal{S}, \mathcal{A}, \mathcal{K}, \mathcal{E})$ *is an authentication code. Then*

$$\log Pd_0 \geq H(\mathbf{K}|\mathbf{M}) - H(\mathbf{K}).$$

PROOF From Equation (10.1), we have

$$Pd_0 = \max\{payoff(s,a) : s \in \mathcal{S}, a \in \mathcal{A}\}.$$

Since the maximum of the values $payoff(s,a)$ is greater than their weighted average, we obtain

$$Pd_0 \geq \sum_{s \in \mathcal{S}, a \in \mathcal{A}} p_{\mathcal{M}}(s,a) payoff(s,a).$$

Hence, by Jensen's inequality (Theorem 2.5), we have

$$\log Pd_0 \geq \log \sum_{s \in \mathcal{S}, a \in \mathcal{A}} p_{\mathcal{M}}(s,a) payoff(s,a)$$

$$\geq \sum_{s \in \mathcal{S}, a \in \mathcal{A}} p_{\mathcal{M}}(s,a) \log payoff(s,a).$$

Recalling from Section 10.2 that

$$p_{\mathcal{M}}(s,a) = p_{\mathcal{S}}(s) \times payoff(s,a),$$

we see that

$$\log Pd_0 \geq \sum_{s \in \mathcal{S}, a \in \mathcal{A}} p_{\mathcal{S}}(s) payoff(s,a) \log payoff(s,a).$$

Now, we observe that $payoff(s, a) = p_{\mathcal{A}}(a|s)$ (i.e., the probability that a is the authenticator, given that s is the source state). Hence,

$$\begin{aligned}\log Pd_0 &\geq \sum_{s\in\mathcal{S}, a\in\mathcal{A}} p_{\mathcal{S}}(s)p_{\mathcal{A}}(a|s)\log p_{\mathcal{A}}(a|s)\\ &= -H(\mathbf{A}|\mathbf{S}),\end{aligned}$$

by the definition of conditional entropy. We complete the proof by showing that $-H(\mathbf{A}|\mathbf{S}) = H(\mathbf{K}|\mathbf{M}) - H(\mathbf{K})$. This follows from basic entropy identities. On one hand, we have

$$H(\mathbf{K}, \mathbf{A}, \mathbf{S}) = H(\mathbf{K}|\mathbf{A}, \mathbf{S}) + H(\mathbf{A}|\mathbf{S}) + H(\mathbf{S}).$$

On the other hand, we compute

$$\begin{aligned}H(\mathbf{K}, \mathbf{A}, \mathbf{S}) &= H(\mathbf{A}|\mathbf{K}, \mathbf{S}) + H(\mathbf{K}, \mathbf{S})\\ &= H(\mathbf{K}) + H(\mathbf{S}),\end{aligned}$$

where we use the facts that $H(\mathbf{A}|\mathbf{K}, \mathbf{S}) = 0$ since the key and source state uniquely determine the authenticator, and $H(\mathbf{K}, \mathbf{S}) = H(\mathbf{K}) + H(\mathbf{S})$ since the source and key are independent events.

Equating the two expressions for $H(\mathbf{K}, \mathbf{A}, \mathbf{S})$, we obtain

$$-H(\mathbf{A}|\mathbf{S}) = H(\mathbf{K}|\mathbf{A}, \mathbf{S}) - H(\mathbf{K}).$$

But a message $m = (s, a)$ is defined to consist of a source state and an authenticator (i.e., $\mathcal{M} = \mathcal{S} \times \mathcal{A}$). Hence, $H(\mathbf{K}|\mathbf{A}, \mathbf{S}) = H(\mathbf{K}|\mathbf{M})$ and the proof is complete. ∎

There is a similar bound for Pd_1 which we will not prove here. It is as follows:

THEOREM 10.14
Suppose that $(\mathcal{S}, \mathcal{A}, \mathcal{K}, \mathcal{E})$ is an authentication code. Then

$$\log Pd_1 \geq H(\mathbf{K}|\mathbf{M}^2) - H(\mathbf{K}|\mathbf{M}).$$

We need to define what we mean by the random variable $\mathbf{M}^2$. Suppose we authenticate two distinct source states using the same key K. In this way, we obtain an ordered pair of messages $(m_1, m_2) \in \mathcal{M} \times \mathcal{M}$. In order to define a probability distribution on $\mathcal{M} \times \mathcal{M}$, it is necessary to define a probability distribution on $\mathcal{S} \times \mathcal{S}$, with the stipulation that $p_{\mathcal{S}\times\mathcal{S}}(s, s) = 0$ for every $s \in \mathcal{S}$ (that is, we do not allow source states to be repeated). The probability distributions on $\mathcal{K}$ and $\mathcal{S} \times \mathcal{S}$ will induce a probability distribution on $\mathcal{M} \times \mathcal{M}$, in the same way that the probability distributions on $\mathcal{K}$ and $\mathcal{S}$ induce a probability distribution on $\mathcal{M}$.

As an illustration of the two bounds, we consider our basic orthogonal array construction and show that the bounds of Theorems 10.13 and 10.14 are both met with equality. First, it is clear that

$$H(\mathbf{K}) = \log \lambda n^2,$$

since each of the λn^2 authentication rules are chosen with equal probability. Let's next turn to the computation of $H(\mathbf{K}|\mathbf{M})$. If any message $m = (s, a)$ is observed, this restricts the possible keys to a subset of size λn. Each of these λn keys is equally likely. Hence, $H(\mathbf{K}|m) = \log \lambda n$, for any message m. Then, we get the following:

$$\begin{aligned} H(\mathbf{K}|\mathbf{M}) &= \sum_{m \in \mathcal{M}} p_{\mathcal{M}}(m) H(\mathbf{K}|m) \\ &= \sum_{m \in \mathcal{M}} p_{\mathcal{M}}(m) \log \lambda n \\ &= \log \lambda n. \end{aligned}$$

Thus we have

$$H(\mathbf{K}|\mathbf{M}) - H(\mathbf{K}) = \log \lambda n - \log \lambda n^2 = -\log n = \log P d_0,$$

so the bound is met with equality.

If we observe two messages which have been produced using the same key (and different source states), then the number of possible keys is reduced to λ. Using similar reasoning as above, we have that $H(\mathbf{K}|\mathbf{M}^2) = \log \lambda$. Then

$$H(\mathbf{K}|\mathbf{M}^2) - H(\mathbf{K}|\mathbf{M}) = \log \lambda - \log \lambda n = -\log n = \log P d_1,$$

so this bound is also met with equality.

10.5 Notes and References

Authentication codes were invented in 1974 by Gilbert, MacWilliams, and Sloane [GMS74]. Much of the theory of authentication codes was developed by Simmons, who proved many fundamental results in the area. Two useful survey articles by Simmons are [SI92] and [SI88]. Another good survey is Massey [MA86].

The connections between orthogonal arrays and authentication codes has been addressed by several researchers. The treatment here is based on three papers by Stinson [ST88], [ST90] and [ST92]. Orthogonal arrays have been studied for over 45 years by researchers in statistics and in combinatorial design theory. For example, the bound in Theorem 10.9 was first proved by Plackett and Berman in 1945 in [PB45]. Many interesting results on orthogonal arrays can be found in

various textbooks on combinatorial design theory such as Beth, Jungnickel, and Lenz [BJL85].

Finally, the use of entropy techniques in the study of authentication codes was introduced by Simmons. The bound of Theorem 10.13 was first proved in Simmons [SI85]; a proof of Theorem 10.14 can be found in Walker [WA90].

Exercises

10.1 Compute Pd_0 and Pd_1 for the following authentication code, represented in matrix form:

key	1	2	3	4
1	1	1	2	3
2	1	2	3	1
3	2	1	3	1
4	2	3	1	2
5	3	2	1	3
6	3	3	2	1

The probability distributions on $\mathcal{S}$ and $\mathcal{K}$ are as follows:

$$p_{\mathcal{S}}(1) = p_{\mathcal{S}}(4) = 1/6, p_{\mathcal{S}}(2) = p_{\mathcal{S}}(3) = 1/3$$

$$p_{\mathcal{K}}(1) = p_{\mathcal{K}}(6) = 1/4, p_{\mathcal{K}}(2) = p_{\mathcal{K}}(3) = p_{\mathcal{K}}(4) = p_{\mathcal{K}}(5) = 1/8.$$

What are the optimal impersonation and substitution strategies?

10.2 We have seen a construction for an orthogonal array $\mathrm{OA}(p, p, 1)$ when p is prime. Prove that this $\mathrm{OA}(p, p, 1)$ can always be extended by one extra column to form an $\mathrm{OA}(p, p+1, 1)$. Illustrate your construction in the case $p = 5$.

10.3 Suppose A is an $\mathrm{OA}(n_1, k, \lambda_1)$ on symbol set $\{1, \ldots, n_1\}$ and suppose B is an $\mathrm{OA}(n_2, k, \lambda_2)$ on symbol set $\{1, \ldots, n_2\}$. We construct C, an $\mathrm{OA}(n_1 n_2, k, \lambda_1 \lambda_2)$ on symbol set $\{1, \ldots, n_1\} \times \{1, \ldots, n_2\}$, as follows: for each row $r_1 = (x_1, \ldots, x_k)$ of A and for each row $s_1 = (y_1, \ldots, y_k)$ of B, define a row

$$t_1 = ((x_1, y_1), \ldots, (x_k, y_k))$$

of C. Prove that C is indeed an $\mathrm{OA}(n_1 n_2, k, \lambda_1 \lambda_2)$.

10.4 Construct an orthogonal array $\mathrm{OA}(3, 13, 3)$.

10.5 Write a computer program to compute $H(\mathbf{K})$, $H(\mathbf{K}|\mathbf{M})$ and $H(\mathbf{K}|\mathbf{M}^2)$ for the authentication code from Exercise 10.1. The probability distribution on sequences of two sources is as follows:

$$p_{\mathcal{S}^2}(1,2) = p_{\mathcal{S}^2}(1,3) = p_{\mathcal{S}^2}(1,4) = 1/18$$

$$p_{\mathcal{S}^2}(2,1) = p_{\mathcal{S}^2}(2,3) = p_{\mathcal{S}^2}(2,4) = 1/9$$

$$p_{\mathcal{S}^2}(3,1) = p_{\mathcal{S}^2}(3,2) = p_{\mathcal{S}^2}(3,4) = 1/9$$

$$p_{\mathcal{S}^2}(4,1) = p_{\mathcal{S}^2}(4,2) = p_{\mathcal{S}^2}(4,3) = 1/18$$

Compare the entropy bounds for Pd_0 and Pd_1 with the actual values you computed in Exercise 10.1.

HINT To compute $p_{\mathcal{K}}(k|m)$, use Bayes' formula

$$p_{\mathcal{K}}(k|m) = \frac{p_{\mathcal{M}}(m|k)p_{\mathcal{K}}(k)}{p_{\mathcal{M}}(m)}.$$

We already know how to calculate $p_{\mathcal{M}}(m)$. To compute $p_{\mathcal{M}}(m|k)$, write $m = (s, a)$, and then observe that $p_{\mathcal{M}}(m|k) = p_{\mathcal{S}}(s)$ if $e_k(s) = a$, and $p_{\mathcal{M}}(m|k) = 0$ otherwise.

To compute $p_{\mathcal{K}}(k|m_1, m_2)$, use Bayes' formula

$$p_{\mathcal{K}}(k|m_1, m_2) = \frac{p_{\mathcal{M}^2}(m_1, m_2|k)p_{\mathcal{K}}(k)}{p_{\mathcal{M}^2}(m_1, m_2)}.$$

$p_{\mathcal{M}^2}(m_1, m_2)$ can be calculated as follows: write $m_1 = (s_1, a_1)$ and $m_2 = (s_2, a_2)$. Then

$$p_{\mathcal{M}^2}(m_1, m_2) = p_{\mathcal{S}^2}(s_1, s_2) \times \sum_{\{K \in \mathcal{K}: e_k(s_1)=a_1, e_k(s_2)=a_2\}} p_{\mathcal{K}}(K).$$

(Note the similarity with the computation of $p(m)$.) To compute $p_{\mathcal{M}^2}(m_1, m_2|k)$, observe that $p_{\mathcal{M}^2}(m_1, m_2|k) = p_{\mathcal{S}^2}(s_1, s_2)$ if $e_k(s_1) = a_1$ and $e_k(s_2) = a_2$, and $p_{\mathcal{M}^2}(m_1, m_2|k) = 0$, otherwise.