Bachelor's Thesis Computing Science

Security of the Sponge Construction with a Random Transformation

Robin Foekens
4666615

January 4, 2023

First supervisor/assessor:
Dr. Bart Mennink

Second supervisor:
Charlotte Lefevre

Second assessor:
Prof. Joan Daemen

Radboud University

Abstract

The sponge construction is often used for the design of cryptographic hash functions. Some interesting security properties of hash functions are collision, preimage and second preimage security.

Recently, a tight bound for preimage security has been proven in the case where this construction uses a permutation. Since tight bounds for collision and second preimage security had been proven already, this implies we have tight bounds for all of the above security properties for this construction.

In the case where the sponge uses a transformation instead of a permutation, the collision attack carries over and this proves that the bound for collision security is tight. However, tight first and second preimage bounds have not been proven yet. We will prove those bounds and show that they correspond to the original expectations for the sponge.

Contents

Chapter 1

Introduction

The sponge construction of Bertoni et al. [4] is often used for cryptographic hashing algorithms. It has a b-bit state, which consists of an inner part of c bits, that is called the capacity, and an outer part of r bits, that is called the rate. So $b = r + c$. The construction has two phases, the absorbing phase and the squeezing phase. In the absorbing phase, data is 'absorbed' into the construction in r-bit parts. During the squeezing phase, data is extracted from the construction in r-bit parts. After every step of these two phases, the state is updated with a function f. In the case of a hash function, the extracted data is then used as a digest.

The developers of PHOTON [5] proposed a generalization of this construction where a higher rate $r' \geq r$ is used during the squeezing phase. In that case, we have $b = r + c = r' + c'$. This is depicted in Figure 2.1. In this thesis, we will look at the generalized sponge construction, but for simplicity we will call it the "sponge".

When studying hash functions, we are generally interested in certain security properties. These are collision, first and second preimage resistance. For collision resistance, we consider an adversary that tries to find two different messages that give the same output. For preimage resistance, we consider an adversary that receives a certain digest and tries to find a message that gives that digest when inputted into the sponge. For second preimage resistance, we consider an adversary that receives a message and tries to find another message that gives the same digest as the received message when inputted into the sponge.

In the case of a sponge where we take a permutation for f, tight bounds for collision, first and second preimage resistance are proven [3, 1]. The bound for first preimage resistance has only been proven recently [6].

We will consider the sponge construction where f can be any transformation. The bound for collision resistance from the sponge with a permutation is still tight in this construction. However, tight bounds for first and second preimage resistance have not been proven yet. The best attacks for the sponge with a permutation do not work in this case, since these use the inverse of the permutation. If we use a transformation instead of a permutation, such an inverse does not need to exist.

When the sponge construction was first proposed, the designers expected the workload for a preimage attack to be 2^n and for a second preimage attack to be $\min\left\{\frac{2^c}{k}, 2^n\right\}$, where k is the number of r-bit message blocks that the input consists of [4].

In this thesis, we will solve this open problem and prove that the bounds of $\frac{q}{2^n}$ for the first and $\frac{q}{2^b} + \frac{q}{2^n} + \frac{q \cdot k}{2^c}$ for second preimage resistance of the sponge with a random function are tight up to constants, for any adversary that can make at most q queries. This is motivated by and closely follows the paper that proves tight preimage resistance of the sponge construction [6].

We will first give some notations and give a more detailed description of the sponge construction and the security model in Chapter 2. Then, we will look at the best known preimage and second preimage attack in Chapter 3. In Chapter 4 we will prove a tight bound for preimage resistance, and in Chapter 5 we will do the same for second preimage resistance. Finally, in Chapter 6 we will reflect on our findings.

Chapter 2

Preliminaries

2.1 Notation

For $b \in \mathbb{N}$, we use $\{0,1\}^b$ to denote the set of binary strings with length b. We define $\{0,1\}^*$ as $\cup_{b\in\mathbb{N}}\{0,1\}^b$, the set of binary strings with arbitrary length. For a b-bit string s and $0 \leq x \leq y \leq b-1$, $s[x:y]$ denotes the substring of s from the bits of position x to y. We denote $\text{inner}_x(s) = s[b-x:b-1]$ and $\text{outer}_x(s) = s[0:x-1]$.

For a finite set S, $x \overset{\$}{\leftarrow} S$ means that x is a uniformly random drawing from S.

The set $Funct(b)$ denotes the set of functions from $\{0,1\}^b \to \{0,1\}^b$. For a random function $f \overset{\$}{\leftarrow} Funct(b)$ and for $i \in \mathbb{N}$, f^0 means the identity function and f^i means i iterations of f.

2.2 Generalized Sponge Construction

We will be looking at a generalized sponge construction as seen in Figure 2.1 [5], based on the sponge construction [4], but where we can also use a larger rate in the squeezing phase. Let $b, c, r, c', r', n \in \mathbb{N}$ such that $b = c+r = c'+r'$ and $r' \geq r$. Let $f \in Funct(b)$ be a function on b bits. Let `pad` be an injective padding function that pads a message M into blocks of r bits such that the last block is non-zero. We will only look at sponge constructions with a fixed output of n bits and we take $l = \lceil n/r' \rceil$.

For any input message $M \in \{0,1\}^*$, we define the sponge construction with function $f \in Funct(b)$, denoted by $\mathcal{H}^f(M) : \{0,1\}^* \to \{0,1\}^n$, as follows.

- M is padded and split it into blocks or size r with `pad`, such that $M_1 \parallel \ldots \parallel M_k \leftarrow \texttt{pad}(M)$ for some k.

- Absorbing phase: The sponge construction has a b-bit state that we call S_i for $i = 0, \ldots, k$. A state $S_0 \leftarrow 0^b$ is initialized. We have $S_i \leftarrow f(S_{i-1} \oplus (M_i \| 0^c))$.

- Squeezing phase: We take S'_0 as the last state of the absorbing phase. For $i = 1, \ldots, l$ we have $S'_i \leftarrow f(S'_{i-1})$ and we extract the outer r' bits as $Z_i \leftarrow \text{outer}_{r'}(S'_{i-1})$.
- We take as digest $Z \leftarrow (Z_1 \| \ldots \| Z_l)[0 : n-1]$.

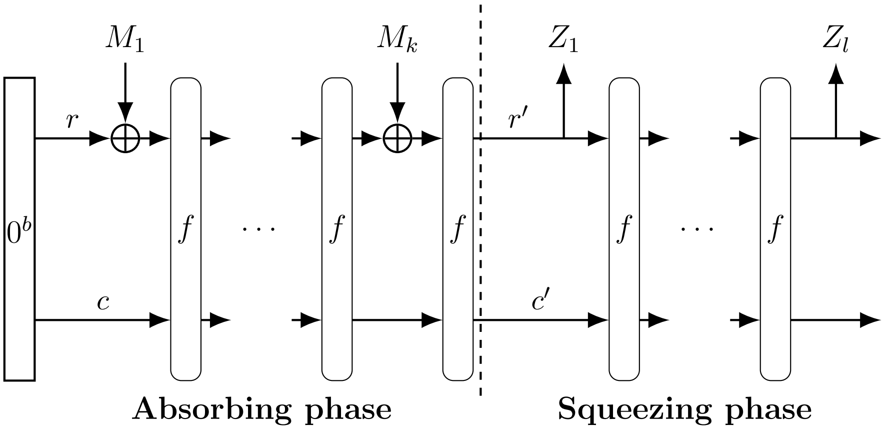

Figure 2.1: Generalized sponge construction as described in Section 2.1.

2.3 Security Model

Consider an adversary $\mathcal{A}$, which is a probabilistic algorithm. The adversary can make queries to a function $f \xleftarrow{\$} Funct(b)$. The number of queries is called q. The collection of queries that $\mathcal{A}$ has made is called $\mathcal{Q}$ and is an ordered list of tuples $(X, Y) \in \{0,1\}^b \times \{0,1\}^b$, where $f(X) = Y$. We can assume without loss of generality that the adversary never repeats queries. For $i \in \mathbb{N}$, we use $\mathcal{Q}_i$ to denote the set of the first i queries the adversary has made.
We use a graph representation to represent the query history $\mathcal{Q}$ of an adversary $\mathcal{A}$. The graph representation of $\mathcal{Q}$ contains nodes $\{0,1\}^b$, which represents all the possible internal states of the sponge. For each query $(X, Y) \in \mathcal{Q}$ for any $M \in \{0,1\}^r$, the edge $X \oplus (M \| 0^c) \xrightarrow{M} Y$ is added. In the case of a zero-block message, we write $X \rightarrow Y$. This occurs in the squeezing phase.

We look at two security properties for the sponge construction of Section 2.2 in the random function model [7].

Everywhere preimage resistance. Given any digest $Z \in \{0,1\}^n$ of length n bits, an adversary $\mathcal{A}$ wants to output a message M such that $\mathcal{H}^f(M) = Z$. We describe this by $M \leftarrow \mathcal{A}^f(Z)$. Note that we do not assume here that a preimage exists.
More formally, the everywhere preimage advantage of an adversary is:

$$\mathbf{Adv}_{\mathcal{H}}^{\mathrm{epre}}(\mathcal{A}) = \max_{Z \in \{0,1\}^n} \mathbf{Pr}(f \overset{\$}{\leftarrow} \mathit{Funct}(b), M \leftarrow \mathcal{A}^f(Z) : \mathcal{H}^f(M) = Z).$$

$\mathbf{Adv}_{\mathcal{H}}^{\mathrm{epre}}(q)$ is the maximum advantage over all adversaries making at most q queries.

Everywhere second preimage resistance. Given $k \in \mathbb{N}$ and any message $M \in \{0,1\}^*$ of arbitrary length such that $|\mathtt{pad}(M)| = r \cdot k$, an adversary $\mathcal{A}$ wants to find a message $M' \in \{0,1\}^*$ of arbitrary length such that $M \neq M'$ and $\mathcal{H}^f(M) = \mathcal{H}^f(M')$. We describe this by $M' \leftarrow \mathcal{A}^f(M)$.
More formally, the everywhere second preimage advantage of an adversary is:

$$\mathbf{Adv}_{\mathcal{H}}^{\mathrm{esec}(k)}(\mathcal{A}) =$$
$$\max_{\substack{M \in \{0,1\}^* \\ |\mathtt{pad}(M)| = rk}} \mathbf{Pr}(f \overset{\$}{\leftarrow} \mathit{Funct}(b), M' \leftarrow \mathcal{A}^f(M) : M' \neq M, \mathcal{H}^f(M) = \mathcal{H}^f(M')).$$

$\mathbf{Adv}_{\mathcal{H}}^{\mathrm{esec}(k)}(q)$ is the maximum advantage over all adversaries making at most q queries.

Chapter 3

Attacks

We will give the best known generic attack for preimages in Section 3.1 and for second preimages in Section 3.2. The attacks are based on [4].

3.1 Preimage attack

In this attack, we first assume that a preimage exists.
We make a case distinction depending on the values of n, c', b and l, with $l = \lceil \frac{n}{r'} \rceil$:

- Case $2^n \leq 2^{c'} \cdot (l-1) + 2^b$. The adversary picks a message M and queries $\mathcal{H}^f(M)$. Then $\mathcal{H}^f(M) = Z$ with probably 2^{-n}, since there are approximately 2^n possible results for this query. After $q \approx 2^n$ attempts, the adversary has found a preimage M with high probability.
- Case $2^{c'} \cdot (l-1) + 2^b < 2^n$. This attack consists of two sequential parts.
  - First, the adversary finds a state Y just before the first squeeze, which satisfies for $i = 0, \dots, l$:

    $$Z_i := \begin{cases} \text{outer}_{r'}(f^i(Y)) & \text{for } i = 0, \dots, l-1, \\ \text{outer}_{n-(l-1)r'}(f^i(Y)) & \text{for } i = l. \end{cases}$$

    This state is also shown in Figure 3.1.
    We know that the outer r' bits in Y need to be equal to Z_1, so we have $\text{outer}_{r'}(Y) = Z_1$. Picking these bits will mean that the equation for Z_1 will hold. Then the adversary needs to pick the other c' bits in such a way that the before mentioned equations for Y hold.
    The adversary can pick any $\text{inner}_{c'}(Y)$ and check whether the equations above hold. Checking this will take at most $l-1$ queries per attempt. This means that the adversary has found a state Y after $q \approx (l-1) \cdot \min\left\{2^{c'}, 2^{n-r'}\right\} = 2^{c'} \cdot (l-1)$ queries.

- Second, the adversary wants to get from the initial state 0^b to Y. The message that the adversary uses to do this can have any length other than 0, because of padding, since there are no restrictions on the size of a preimage.

  The adversary takes a message block $M_i \in \{0,1\}^r$ and a state $Y_j \in \{0,1\}^b$. This state is either the initial state 0^b, or a state that the adversary has queried to before. The adversary queries $f(Y_j \oplus (M_i \| 0^c))$ until a query gives Y.

  Since the adversary then has queries $0^b \xrightarrow{M_1} \dots \xrightarrow{M_i} Y$ in $\mathcal{Q}$ for some $i \in \mathbb{N}^+$, we know that $M \in \{0,1\}^*$ with $\texttt{pad}(M) = M_1 \| \dots \| M_i$ gives a valid preimage.

  The probability that the adversary will get (X, Y) for some $X \in \{0,1\}^b$ per query is $\frac{1}{2^b}$, since it tries to hit a b-bit state through a random function. This means that after $q \approx 2^b$ attempts, the adversary has found a chain to Y with high probability.

Combined, this case requires $q \approx 2^{c'} \cdot (l-1) + 2^b$ evaluations.

These two cases combined give a required amount of

$$q \approx \min\left\{2^n, 2^{c'} \cdot (l-1) + 2^b\right\}$$

evaluations.

However, if we do not assume that a preimage exists, the state Y from the second case of the attack does not necessarily have to exist. Since $\text{outer}_{r'}(Y) = Z_1$, there are only $2^{c'}$ possible attempts. However, the probability that one attempt succeeds is $2^{n-r'}$. We will look at this probability in more detail in Section 4.2. This means that the probability that the second case of the attack succeeds is only $\frac{2^{c'}}{2^{n-r'}} = 2^{b-n}$.

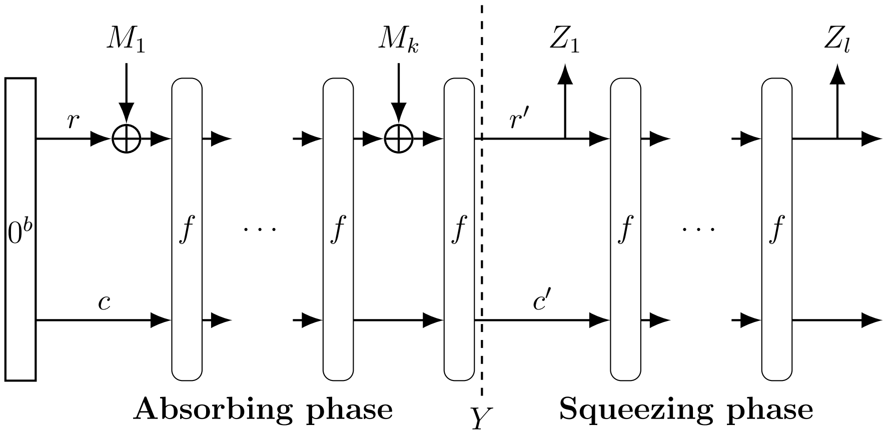

Figure 3.1: State Y in the preimage attack from Section 3.1.

3.2 Second preimage attack

We make a case distinction depending on the values of n, b, r and k, where k is the amount of message blocks of M:

- Case $2^n \leq \frac{2^b}{1+k2^r}$. The adversary picks a message $M' \neq M$ and queries $\mathcal{H}^f(M')$. Then $\mathcal{H}^f(M') = Z$ with probability of approximately 2^{-n}. After $q \approx 2^n$, the adversary has found a second preimage M' with high probability.

- Case $\frac{2^b}{1+k2^r} < 2^n$. The adversary first queries $\mathcal{H}^f(M) = Z$ where $Z \in \{0,1\}^n$. We take $\texttt{pad}(M) = M_1 \| \ldots \| M_k$ and define Y_i for $i = 0, \ldots, k$ as $Y_0 = 0^b$ and $Y_i = f(Y_{i-1} \oplus (M_i \| 0^c))$ for $i = 1, \ldots, k$. The adversary then ends up with Figure 3.2.

  The adversary takes a message block $m' \in \{0,1\}^r$ and a state $Y'' \in \{0,1\}^b$. This state is either the initial state 0^b, or a state that the adversary has queried to before. It then queries $f(Y'' \oplus (m' \| 0^c))$, until this gives a state $Y' \in \{0,1\}^b$ with $\text{inner}_c(Y') = \text{inner}_c(Y_i)$ for some $i \in \{0, \ldots, k-1\}$ or with $Y' = Y_k$. This means the adversary has found a path $0^b \xrightarrow{M'_0} \ldots \xrightarrow{M'_j} Y'$.

  If $\text{inner}_c(Y') = \text{inner}_c(Y_i)$ for some $i \in \{0, \ldots, k-1\}$, the adversary has found a preimage $\texttt{pad}(M') = M'_1 \| \ldots \| M'_j \| \text{upper}_r(Y') \oplus \text{upper}_r(Y_i)$ $\oplus M_{i+1} \| M_{i+2} \| \ldots \| M_k$.
  Otherwise, $Y' = Y_k$ and the adversary has found a preimage $\texttt{pad}(M') = M'_1 \| \ldots \| M'_j$.

Per query, the adversary has a $\frac{1}{2^c}$ probability to hit the inner part state Y_i with $i = 0, \ldots k-1$, so a $\frac{k}{2^c}$ probability to hit any of those cases. The adversary also has a $\frac{1}{2^b}$ probability to hit the state Y_k. Each query then has a success probability of approximately $\frac{k}{2^c} + \frac{1}{2^b}$.

After $q \approx \frac{1}{2^{-b}+k2^{-c}} = \frac{2^b}{1+k2^r}$ the adversary has found a message M' such that $\mathcal{H}^f(M') = \mathcal{H}^f(M)$ with high probability.

In total, this attack requires $q \approx \min\left\{2^n, \frac{2^b}{1+k2^r}\right\}$ queries. This gives $\mathbf{Adv}_{\mathcal{H}}^{\text{esec}(k)}(q) \approx 1$ for this q.

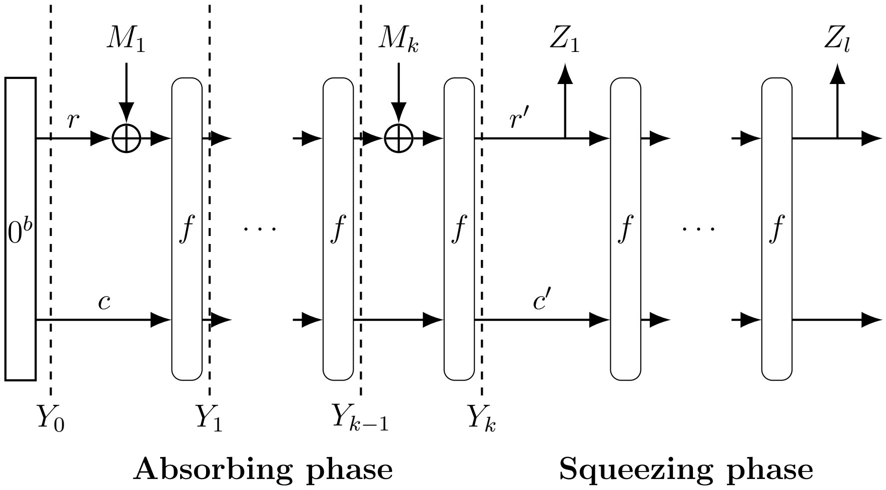

Figure 3.2: The state of the sponge construction after querying M.

Chapter 4

Preimage Resistance

Theorem 1. *Let $b, c, r, c', r', n, q \in \mathbb{N}$ with $b = c+r = c'+r'$ and let $l = \lceil \frac{n}{r'} \rceil$. The sponge construction $\mathcal{H}$ of Section 2.2 then satisfies the following bound:*

$$\mathbf{Adv}_{\mathcal{H}}^{epre}(q) \leq \frac{q}{2^n}.$$

In particular, the advantage of the adversary is independent of c.

The proof is given in the remainder of this section.

4.1 Setup

Let $Z \in \{0,1\}^n$ be any image, and $Z = Z_1 \| Z_2 \| \dots \| Z_l$ with $|Z_i| = r'$ for $i \in \{1, \dots, l-1\}$ and $|Z_l| = s \leq r'$. Let $\mathcal{A}$ be any preimage adversary as defined in Section 2.3. We use the graph representation for its query history $\mathcal{Q}$ as introduced in Section 2.3.
Let $\mathbf{Z}_i$ be defined as follows:

$$\mathbf{Z}_i := \begin{cases} \left\{ Y_i \in \{0,1\}^b \mid \text{outer}_{r'}(Y_i) = Z_i \right\}, & \text{for } i \in \{1, \dots, l-1\}, \\ \left\{ Y_i \in \{0,1\}^b \mid \text{outer}_{s}(Y_i) = Z_i \right\}, & \text{for } i = l. \end{cases} \tag{4.1}$$

The goal of $\mathcal{A}$ is to find a preimage of Z, which implies the event PRE($\mathcal{Q}$):

$$\begin{aligned} \text{PRE}(\mathcal{Q}) : \mathcal{Q} \text{ defines a path } 0^b \xrightarrow{M_1} \dots \xrightarrow{M_{k-1}} Y_0 \xrightarrow{M_k} Y_1 \longrightarrow \cdots \longrightarrow Y_l \\ \text{such that } Y_i \in \mathbf{Z}_i \text{ for } i = 1, \dots, l. \end{aligned} \tag{4.2}$$

In the case of the padding we introduced in Section 2.2, finding a preimage corresponds to PRE($\mathcal{Q}$) with the restriction that the last message block is non-zero. The preimage found by $\mathcal{A}$ is then the unique message M such that $\text{pad}(M) = M_1 \| \dots \| M_k$.

We are interested in the set $\mathcal{S}$ of starting points of good chains in the squeezing phase, which is shown as Y in Figure 3.1. This is defined as:

$$\mathcal{S} = \left\{ Y \mid f^{i-1}(Y) \in \mathbf{Z}_i \text{ for all } i \in \{1, \dots, l\} \right\} \subseteq \mathbf{Z}_1. \tag{4.3}$$

We define BAD($\mathcal{Q}$) as:

$$\text{BAD}(\mathcal{Q}) : \mathcal{Q} \text{ contains a query } (X, Y) \text{ for some } X \in \{0,1\}^b \text{ with } Y \in \mathcal{S}. \tag{4.4}$$

Then clearly PRE($\mathcal{Q}$) $\Rightarrow$ BAD($\mathcal{Q}$) and thus

$$\mathbf{Pr}(\text{PRE}(\mathcal{Q})) \leq \mathbf{Pr}(\text{BAD}(\mathcal{Q})). \tag{4.5}$$

4.2 Probability computation

We have

$$\begin{aligned}
&\mathbf{Pr}(\text{BAD}(\mathcal{Q})) = \mathbf{Pr}(\text{BAD}(\mathcal{Q}_q)) = \sum_s \mathbf{Pr}(\text{BAD}(\mathcal{Q}_q) \wedge |\mathcal{S}| = s) \\
&= \sum_{i=1}^{q} \sum_s \mathbf{Pr}(\text{BAD}(\mathcal{Q}_i) \wedge \neg\text{BAD}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s) \\
&= \sum_{i=1}^{q} \sum_s \mathbf{Pr}(\text{BAD}(\mathcal{Q}_i) \,|\, \neg\text{BAD}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s) \cdot \mathbf{Pr}(\neg\text{BAD}_{i-1} \wedge |\mathcal{S}| = s).
\end{aligned} \tag{4.6}$$

We first calculate $\mathbf{Pr}(\text{BAD}(\mathcal{Q}_i) \,|\, \neg\text{BAD}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s)$ for any $i, s \in \mathbb{N}$. We assume that $\neg$BAD($\mathcal{Q}_{i-1}$) and $|\mathcal{S}| = s$ and calculate the probability that BAD($\mathcal{Q}_i$). BAD($\mathcal{Q}_i$) means that there is some query $(X', Y') \in \mathcal{Q}_i$ with $X' \in \{0,1\}^b$ and $Y' \in \mathcal{S}$. Since we assumed that $\neg$BAD($\mathcal{Q}_{i-1}$), (X', Y') must be the i^{th} query.
Let the i^{th} query be (X_i, Y_i) with $X_i, Y_i \in \{0,1\}^b$. We then only have to calculate the probability that $Y_i \in \mathcal{S}$. Since Y_i is uniformly drawn from the set of possible states $\{0,1\}^b$, there are 2^b possible values for Y_i. Of these, only $|\mathcal{S}| = s$ give $Y_i \in \mathcal{S}$. This gives

$$\mathbf{Pr}(\text{BAD}(\mathcal{Q}_i) \,|\, \neg\text{BAD}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s) = \frac{s}{2^b}. \tag{4.7}$$

Then we look at $\mathbf{Pr}(\neg\text{BAD}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s)$. Since $\neg\text{BAD}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s \Rightarrow |\mathcal{S}| = s$, we have

$$\mathbf{Pr}(\neg\text{BAD}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s) \leq \mathbf{Pr}(|\mathcal{S}| = s). \tag{4.8}$$

Combined with (4.7), this gives:

$$\mathbf{Pr}(\mathrm{BAD}(\mathcal{Q})) \leq \sum_{i=1}^{q} \sum_{s} \mathbf{Pr}(|\mathcal{S}| = s) \frac{s}{2^b} = \sum_{i=1}^{q} \mathrm{E}(|\mathcal{S}|) \cdot \frac{1}{2^b} = \frac{q}{2^b} \cdot \mathrm{E}(|\mathcal{S}|). \tag{4.9}$$

What remains is evaluating $\mathrm{E}(|\mathcal{S}|)$. For this, we define for any $Y \in \{0,1\}^b$ Bernoulli variable I_Y as

$$I_Y = 1 \Longleftrightarrow Y \in \mathcal{S}.$$

Since we saw in Section 4.1 that $\mathcal{S} \subseteq \mathbf{Z}_1$, we have:

$$\mathrm{E}(|\mathcal{S}|) = \mathrm{E}\left(\sum_{Y \in \{0,1\}^b} I_Y \right) = \sum_{Y \in \mathbf{Z}_1} \mathrm{E}(I_Y) = \sum_{Y \in \mathbf{Z}_1} \mathbf{Pr}(Y \in \mathcal{S}). \tag{4.10}$$

We first notice that $|\mathbf{Z}_1| = 2^{c'}$, since this set contains all b-bit states with $\mathrm{outer}_{r'}(Y_1) = Z_1$, which leaves $2^{c'}$ options for the other c' bits.
We consider $Y \in \mathbf{Z}_1$. Then $Y \in \mathcal{S}$ if and only if $f^{i-1}(Y) \in \mathbf{Z}_i$ for $i = 2, \ldots, l$. Using (4.10) we have

$$\mathrm{E}(|\mathcal{S}|) = \sum_{Y \in \mathbf{Z}_1} \mathbf{Pr}(Y \in \mathcal{S}) = \sum_{Y \in \mathbf{Z}_1} \prod_{i=2,\ldots,l} \mathbf{Pr}(Y \in \mathbf{Z}_i). \tag{4.11}$$

For $i = 2, \ldots, l-1$, $\mathbf{Pr}(f^{i-1}(Y) \in \mathbf{Z}_i) = \frac{2^{c'}}{2^b}$, since $f^{i-1}(Y)$ is uniformly drawn from 2^b possible b-bit states, of which $2^{c'}$ give $\mathrm{outer}_{r'}(f^{i-1}(Y)) = Z_i$, and thus $f^{i-1}(Y) \in \mathbf{Z}_i$. Similarly, $\mathbf{Pr}(f^{l-1}(Y) \in \mathbf{Z}_l) = \frac{2^{b-s}}{2^b}$.
Combining these probabilities with (4.11), we have:

$$\begin{aligned} \mathrm{E}(|\mathcal{S}|) &= \sum_{Y \in \mathbf{Z}_1} \frac{(2^{c'})^{l-2} \cdot 2^{b-s}}{(2^b)^{l-1}} \\ &= 2^{c'} \frac{(2^{c'})^{l-2} \cdot 2^{b-s}}{(2^b)^{l-1}} \\ &= \frac{(2^{c'})^{l-1} \cdot 2^{b-s}}{(2^{c'+r'})^{l-1}} \\ &= \frac{2^{b-s}}{(2^{r'})^{l-1}} \\ &= \frac{2^b}{2^s \cdot 2^{r'(l-1)}} \\ &= \frac{2^b}{2^n}. \end{aligned} \tag{4.12}$$

Combining (4.12) and (4.9), we have:

$$\mathbf{Pr}(\mathrm{BAD}(\mathcal{Q})) \leq \frac{q}{2^b} \cdot \mathrm{E}(|\mathcal{S}|) = \frac{q}{2^b} \cdot \frac{2^b}{2^n} = \frac{q}{2^n}.$$

From (4.5) then $\mathrm{PRE}(\mathcal{Q}) \leq \frac{q}{2^n}$.

Chapter 5

Second Preimage Resistance

Theorem 2. *Let $b, c, r, c', r', n, q, k \in \mathbb{N}$ with $b = c + r = c' + r'$ and $l = \left\lceil \frac{n}{r'} \right\rceil$. The sponge construction $\mathcal{H}$ of Section 2.2 then satisfies the following bound:*

$$\mathbf{Adv}_{\mathcal{H}}^{esec(k)}(q) \leq \frac{q}{2^b} + \frac{q}{2^n} + \frac{q \cdot k}{2^c}.$$

The proof is given in the remainder of this section.

5.1 Setup

Let $\mathcal{A}$ be any second preimage adversary as defined in Section 2.3 and let M be the first preimage. We take $\texttt{pad}(M) = M_1 \| M_2 \| \dots \| M_k$ with $|M_i| = r$ for $i \in \{1, \dots, k\}$. Let $Z \in \{0,1\}^b$ be equal to $\mathcal{H}^f(M)$. We take $Z = Z_1 \| Z_2 \| \dots \| Z_l$ with $|Z_i| = r'$ for $i \in \{1, \dots, l-1\}$ and $|Z_l| = s \leq r'$.
We define the states $Y_i \in \{0,1\}^b$ for $i = 0, \dots, k$ as $Y_0 = 0^b$ and Y_i for $i = 1, \dots, k$ as $Y_i = f(Y_{i-1} \oplus (M_i \| 0^c))$. In other words, Y_i is the state of the sponge construction after absorbing M_i. This is also seen in Figure 3.2.
We use the graph representation again to represent the adversary's query history $\mathcal{Q}$ as introduced in Section 2.3.
For simplicity, we provide $\mathcal{A}$ at the beginning of the game all the queries $Y_{i-1} \xrightarrow{M_i} Y_i$ for $i = 1, \dots, k$. These are the queries that are made in the absorbing phase of $\mathcal{H}^f(M)$. While the adversary knows these queries and will thus not make them again, they are not put in the query history $\mathcal{Q}$.
We again define $\mathbf{Z}_i$ as (4.1). Then, the goal of $\mathcal{A}$ is to find a second preimage of Z, which implies the event SECPRE($\mathcal{Q}$):

SECPRE($\mathcal{Q}$) : $\mathcal{Q}$ defines a path

$$0^b \xrightarrow{M'_1} \dots \xrightarrow{M'_{k'-1}} Y_{k'-1} \xrightarrow{M'_{k'}} Y_1^{sq} \longrightarrow \dots \longrightarrow Y_l^{sq} \text{ such that}$$

$Y_i^{sq} \in \mathbf{Z}_i$ for $i = 1, \dots, l$ and $M'_1 \| \dots \| M'_{k'} \neq \texttt{pad}(M)$.

5.2 Logic

We define the set $\mathcal{S}$ of starting points of good chains in the squeezing phase as (4.3). In particular $Y_k \in \mathcal{S}$.

We define INNERY($\mathcal{Q}$) as:

$$\text{INNERY}(\mathcal{Q}) : \mathcal{Q} \text{ contains a query } (X, Y) \text{ for some } X \in \{0,1\}^b \\ \text{with } \text{inner}_c(Y) = \text{inner}_c(Y_i) \text{ for } i \in \{0, \dots, k-1\}.$$

We also define HITS($\mathcal{Q}$) as:

$$\text{HITS}(\mathcal{Q}) : \mathcal{Q} \text{ contains a query } (X, Y) \text{ for some } X \in \{0,1\}^b \\ \text{with } Y \in \mathcal{S}.$$

For SECPRE($\mathcal{Q}$) to happen, the adversary must make a query (X, Y) with $X \in \{0,1\}^b$ and $Y \in \mathcal{S}$, such that we get a different full message than $\texttt{pad}(M)$. There are two different way in which it can do this. First, it can get to $Y_k \in \mathcal{S}$ via a state Y' with the same inner part as one of the states Y_i with $i \in \{0, \dots, k-1\}$. This way, it finds a path containing $(Y' \oplus (\text{outer}_r(Y')\|0^c) \oplus (M_{i+1}\|0^c), Y_{i+1})$ and $(Y_{i+1} \oplus (M_{i+1}\|0^c), Y_{i+2}), \dots, (Y_{k-1} \oplus (M_k\|0^c), Y_k)$. Second, it can find some other path to one of the states $Y \in \mathcal{S}$, which includes $Y_k \in \mathcal{S}$. This first event corresponds to INNERY($\mathcal{Q}$) and the second to HITS($\mathcal{Q}$). Since we assumed that the adversary makes no queries that it already knows, and we already gave it the queries $(Y_{i-1} \oplus (M_i\|0^c), Y_i)$ for $i \in \{0, \dots, k\}$, we know that the full message it finds will be different from $\texttt{pad}(M)$.

Thus SECPRE($\mathcal{Q}$) $\Rightarrow$ HITS($\mathcal{Q}$) $\vee$ INNERY($\mathcal{Q}$), so

$$\mathbf{Pr}(\text{SECPRE}(\mathcal{Q})) \leq \mathbf{Pr}(\text{HITS}(\mathcal{Q})) + \mathbf{Pr}(\text{INNERY}(\mathcal{Q})). \tag{5.1}$$

5.3 Probability computation

We first look at $\mathbf{Pr}(\text{HITS}(\mathcal{Q}))$. Similar to (4.6), we get

$$\mathbf{Pr}(\text{HITS}(\mathcal{Q})) = \sum_{i=1}^{q} \sum_{s} \mathbf{Pr}(\text{HITS}(\mathcal{Q}_i) \mid \neg\text{HITS}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s) \\ \cdot \mathbf{Pr}(\neg\text{HITS}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s). \tag{5.2}$$

We first evaluate $\mathbf{Pr}(\text{HITS}(\mathcal{Q}_i) \mid \neg\text{HITS}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s)$ for any $i, s \in \mathbb{N}$. This is the probability that there is query (X, Y) in $\mathcal{Q}_i$ for some $X \in \{0,1\}^b$ with $X \neq Y_{k-1} \oplus (M_k\|0^c)$ and $Y \in \mathcal{S}$, given that $\neg\text{HITS}(\mathcal{Q}_{i-1})$ and $|\mathcal{S}| = s$. Since $\neg\text{HITS}(\mathcal{Q}_{i-1})$ holds, the query that triggers HITS($\mathcal{Q}_i$) must be the i^{th}

query.
Let the i^{th} query be (X, Y) with $X, Y \in \{0,1\}^b$. Since we assumed that the adversary already knows $(Y_{k-1} \oplus (M_k \| 0^c), Y_k)$ and we assumed that the adversary only makes queries that it has never made before in Section 2.3, we know that $X \neq Y_{k-1} \oplus (M_k \| 0^c)$. This probability is then equal to the probability that we calculated in (4.7), so we get

$$\mathbf{Pr}(\mathrm{HITS}(\mathcal{Q}_i) \mid \neg\mathrm{HITS}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s) = \frac{s}{2^b}. \tag{5.3}$$

We then evaluate $\mathbf{Pr}(\neg\mathrm{HITS}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s)$. Similar to (4.8), we get

$$\mathbf{Pr}(\neg\mathrm{HITS}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s) \leq \mathbf{Pr}(|\mathcal{S}| = s). \tag{5.4}$$

Combining (5.2) with (5.3) and (5.4) gives

$$\begin{aligned}\mathbf{Pr}(\mathrm{HITS}(\mathcal{Q})) &\leq \sum_{i=1}^{q} \sum_{s} \frac{s}{2^b} \cdot \mathbf{Pr}(|\mathcal{S}| = s) \\ &= \sum_{i=1}^{q} \mathrm{E}(|\mathcal{S}|) \cdot \frac{1}{2^b} = \frac{q}{2^b} \cdot \mathrm{E}(|\mathcal{S}|). \end{aligned} \tag{5.5}$$

Now we evaluate $\mathrm{E}(|\mathcal{S}|)$. We know that $Y_k \in \mathcal{S}$, so (4.10) gives

$$\begin{aligned}\mathrm{E}(|\mathcal{S}|) = \sum_{Y \in \mathbf{Z}_1} \mathbf{Pr}(Y \in \mathcal{S}) &= \mathbf{Pr}(Y_k \in \mathcal{S}) + \sum_{Y \in \mathbf{Z}_1 \setminus Y_k} \mathbf{Pr}(Y \in \mathcal{S}) \\ &= 1 + \sum_{Y \in \mathbf{Z}_1 \setminus Y_k} \mathbf{Pr}(Y \in \mathcal{S}). \end{aligned} \tag{5.6}$$

We have $|\mathbf{Z}_1 \setminus Y_k| = 2^{c'} - 1 \leq 2^{c'}$. If we combine this with (4.12), we get

$$\begin{aligned}\mathrm{E}(|\mathcal{S}|) = 1 + \sum_{Y \in \mathbf{Z}_1 \setminus Y'} \mathbf{Pr}(Y \in \mathcal{S}) &= 1 + \sum_{Y \in \mathbf{Z}_1 \setminus Y'} \frac{(2^{c'})^{l-2} \cdot 2^{b-s}}{(2^b)^{l-1}} \\ \leq 1 + 2^{c'} \frac{(2^{c'})^{l-2} \cdot 2^{b-s}}{(2^b)^{l-1}} &= 1 + \frac{2^b}{2^n}. \end{aligned} \tag{5.7}$$

Combining (5.5) and (5.7) gives

$$\begin{aligned}\mathbf{Pr}(\mathrm{HITS}(\mathcal{Q})) &\leq \frac{q}{2^b} \cdot \left(1 + \frac{2^b}{2^n}\right) \\ &= \frac{q}{2^b} + \frac{q}{2^n}. \end{aligned} \tag{5.8}$$

We then look at $\mathbf{Pr}(\mathrm{INNERY}(\mathcal{Q}))$. Then

$$\begin{aligned}\mathbf{Pr}(\mathrm{INNERY}(\mathcal{Q})) &= \mathbf{Pr}(\mathrm{INNERY}(\mathcal{Q}_q)) \\ &\leq \sum_{i=1}^{q} \mathbf{Pr}(\mathrm{INNERY}(\mathcal{Q}_i) \mid \neg\mathrm{INNERY}(\mathcal{Q}_{i-1})). \end{aligned} \tag{5.9}$$

$\mathbf{Pr}(\text{INNERY}(\mathcal{Q}_i) \mid \neg\text{INNERY}(\mathcal{Q}_{i-1}))$ is the probability that there is a query (X, Y) in $\mathcal{Q}_i$ for some $X \in \{0,1\}^b$ with $X \neq (Y_{i-1} \oplus (M_i \| 0^c))$ for some $i \in \{1, \ldots, k-1\}$ and $Y = \text{inner}_c(Y_i)$ for some $i \in \{0, \ldots, k-1\}$, given that $\neg\text{INNERY}(\mathcal{Q}_{i-1})$. Since $\neg\text{INNERY}(\mathcal{Q}_{i-1})$ this query must be the i^{th} query. Since we assumed that the adversary does not repeat any queries and that it already knows $(Y_{i-1} \oplus (M_i \| 0^c), Y_i)$ for all $i \in \{1, \ldots, k-1\}$, we know that $X \neq (Y_{i-1} \oplus (M_i \| 0^c))$ for all $i \in \{1, \ldots, k-1\}$.

The Y from the i^{th} query will be uniformly drawn from the set of possible states. There are 2^b possible b-bit states, so Y is uniformly drawn from a set of 2^b elements.

We then want to know for how many $Y \in \{0,1\}^b$, $\text{inner}_c(Y) = \text{inner}_c(Y_i)$ for some $i \in \{0, \ldots, k-1\}$ holds. For a certain $i \in \{0, \ldots, k-1\}$, there are 2^r elements Y with $\text{inner}_c(Y) = \text{inner}_c(Y_i)$. There are clearly at most k options for i, so this means that there are at most $k \cdot 2^r$ elements Y with $\text{inner}_c(Y) = \text{inner}_c(Y_i)$ for some $i \in \{0, \ldots, k-1\}$. This gives

$$\mathbf{Pr}(\text{INNERY}(\mathcal{Q}_i) \mid \neg\text{INNERY}(\mathcal{Q}_{i-1})) \leq \frac{k \cdot 2^r}{2^b} = \frac{k}{2^c}. \tag{5.10}$$

Combining (5.9) with (5.10), we get

$$\begin{aligned} \mathbf{Pr}(\text{INNERY}(\mathcal{Q})) &\leq \sum_{i=1}^{q} \mathbf{Pr}(\text{INNERY}(\mathcal{Q}_i) \mid \neg\text{INNERY}(\mathcal{Q}_{i-1})) \\ &\leq \sum_{i=1}^{q} \frac{k}{2^c} = \frac{q \cdot k}{2^c}. \end{aligned} \tag{5.11}$$

Lastly, combining (5.1) with (5.8) and (5.11) gives

$$\mathbf{Pr}(\text{SECPRE}(\mathcal{Q})) \leq \frac{q}{2^b} + \frac{q}{2^n} + \frac{q \cdot k}{2^c}.$$

Chapter 6

Conclusion

We will now look into our bounds from Chapter 4 and 5. First, by comparing them with the best existing attacks, as described in Chapter 3, which are based upon the descriptions of the designers of the sponge. Second, by comparing them with the expectations for the sponge [4].

First of all we look at everywhere preimage resistance. We obtained the following bound in Theorem 1:

$$\mathbf{Adv}_{\mathcal{H}}^{\mathrm{epre}}(q) \leq \frac{q}{2^n}.$$

The best existing attack as described in Section 3.1 requires $q \approx \min\left\{2^n, 2^{c'} \cdot (l-1) + 2^b\right\}$ queries when we assume a preimage exists. However, when we consider everywhere preimage, we do not assume a preimage exists. In that case, the second part of the attack only succeeds with probability 2^{b-n}. The first part of the attack does still work, which requires $q \approx 2^n$ queries. This means this bound is tight. It also corresponds with the expectation of the designers of the sponge.

Then we look at everywhere second preimage resistance. We obtained the following bound in Theorem 2:

$$\mathbf{Adv}_{\mathcal{H}}^{\mathrm{esec}(k)}(q) \leq \frac{q}{2^b} + \frac{q}{2^n} + \frac{q \cdot k}{2^c}.$$

The best existing attack as described in Section 3.2 requires $q \approx \min\left\{2^n, \frac{2^b}{1+k2^r}\right\}$ queries in Section 3.2. However, the factor $\frac{q}{2^b}$ in the bound of Theorem 2 is negligible, since $b = r + c$ and thus $b > c$. Since this factor is negligible, the bound is tight and corresponds with the expectations.

Another interesting security property that we did not manage to look into, is domain-oriented preimage resistance [2]. In short, for domain-oriented preimage resistance, we take some $M \in \{0,1\}^*$ and consider an adversary who receives the digest $Z = \mathcal{H}^f(M)$. The adversary also receives the length $|M|$, but not M itself, and wants to find some $M' \in \{0,1\}^*$ with $\mathcal{H}^f(M') = Z$.

The second case of the preimage attack that we found in Chapter 3 will then succeed after the required amount of queries, which means that there exists an attack that requires $q \approx \min\left\{2^n, 2^{c'} \cdot (l-1) + 2^b\right\}$ queries.

If we want to prove a bound on the advantage of an adversary with at most q queries, we can follow the proof of Chapter 4 to a degree. A difference is, that since we assume that a preimage exists, $\mathrm{E}(|\mathcal{S}|)$ will be as in (5.7). However, we run into problems when we want to calculate $\mathbf{Pr}(\mathrm{BAD}(\mathcal{Q}_i) \mid \neg\mathrm{BAD}(\mathcal{Q}_{i-1}) \wedge |\mathcal{S}| = s)$ for some $i, s \in \mathbb{N}$, since this probability depends on the query history $\mathcal{Q}_{i-1}$. This is the case because the adversary is allowed to guess M and return that value. This means that the adversary can try to find M by querying all messages of length $|M|$. Specifically, if the adversary already guessed the first $\frac{|\mathtt{pad}(M)|}{r} - 1$ message blocks in $\mathcal{Q}_{i-1}$, the probability that the adversary guesses M in the i^{th} query will be large.
Note that this problem did not occur for second preimage resistance, since the adversary was not allowed to output M in that case.

Bibliography

[1] Elena Andreeva, Bart Mennink, and Bart Preneel. Security reductions of the second round SHA-3 candidates. In Mike Burmester, Gene Tsudik, Spyros S. Magliveras, and Ivana Ilic, editors, *Information Security - 13th International Conference, ISC 2010, Boca Raton, FL, USA, October 25-28, 2010, Revised Selected Papers*, volume 6531 of *Lecture Notes in Computer Science*, pages 39–53. Springer, 2010.

[2] Elena Andreeva and Martijn Stam. The symbiosis between collision and preimage resistance. In Liqun Chen, editor, *Cryptography and Coding - 13th IMA International Conference, IMACC 2011, Oxford, UK, December 12-15, 2011. Proceedings*, volume 7089 of *Lecture Notes in Computer Science*, pages 152–171. Springer, 2011.

[3] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. On the indifferentiability of the sponge construction. In Nigel P. Smart, editor, *Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Istanbul, Turkey, April 13-17, 2008. Proceedings*, volume 4965 of *Lecture Notes in Computer Science*, pages 181–197. Springer, 2008.

[4] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Sponge functions. Ecrypt Hash Workshop 2007, May 2007.

[5] Jian Guo, Thomas Peyrin, and Axel Poschmann. The PHOTON family of lightweight hash functions. In Phillip Rogaway, editor, *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings*, volume 6841 of *Lecture Notes in Computer Science*, pages 222–239. Springer, 2011.

[6] Charlotte Lefevre and Bart Mennink. Tight preimage resistance of the sponge construction. *IACR Cryptol. ePrint Arch.*, page 734, 2022.

[7] Phillip Rogaway and Thomas Shrimpton. Cryptographic hash-function basics: Definitions, implications and separations for preimage resistance, second-preimage resistance, and collision resistance. *IACR Cryptol. ePrint Arch.*, page 35, 2004.