Chapter 05

Mingjia Huo

Problem 5.1. Formal definition of **second preimage resistance**:
The collision-finding experiment Hash-sec$_{\mathcal{A},\Pi}(n)$:

1. A key s is generated by running Gen(1^n). Then uniformly choose $x \in \{0,1\}^*$. (If Π is a fixed-length hash function for inputs of length $l'(n)$, then require $x \in \{0,1\}^{l'(n)}$.)
2. Then adversary $\mathcal{A}$ is given s, x. Then $\mathcal{A}$ outputs $x' \in \{0,1\}^*$.(If Π is a fixed-length hash function for inputs of length $l'(n)$, then require $x \in \{0,1\}^{l'(n)}$.)
3. The output of the experiment is defined to be 1 is and only if $x \neq x'$ and $H^s(x') = H^s(x)$. In such a case we say that $\mathcal{A}$ has found a collision.

Definition 1. *A hash function $\Pi = (Gen, H)$ is second preimage resistance if for all* `PPT` *adversary $\mathcal{A}$, there is a negl(n) such that*

$$\Pr[\textit{Hash-sec}_{\mathcal{A},\Pi}(n) = 1] \leq \textit{negl}(n).$$

Formal definition of **preimage resistance**:
The collision-finding experiment Hash-prei$_{\mathcal{A},\Pi}(n)$:

1. A key s is generated by running Gen(1^n). Then uniformly choose $x \in \{0,1\}^*$. (If Π is a fixed-length hash function for inputs of length $l'(n)$, then require $x \in \{0,1\}^{l'(n)}$.) Compute $y = H^s(x)$.
2. Then adversary $\mathcal{A}$ is given s, y. Then $\mathcal{A}$ outputs $x' \in \{0,1\}^*$.(If Π is a fixed-length hash function for inputs of length $l'(n)$, then require $x \in \{0,1\}^{l'(n)}$.)
3. The output of the experiment is defined to be 1 is and only if $y = H^s(x')$. In such a case we say that $\mathcal{A}$ has found a collision.

Definition 2. *A hash function $\Pi = (Gen, H)$ is preimage resistance if for all* `PPT` *adversaries $\mathcal{A}$, there is a negl(n) such that*

$$\Pr[\textit{Hash-prei}_{\mathcal{A},\Pi}(n) = 1] \leq \textit{negl}(n).$$

Proof of collision resistant to second preimage resistant: Proof by contradiction, assume $\Pi = (\text{Gen}, H)$ is not second preimage resistant but is collision resistant, so there's an `PPT` adversary $\mathcal{A}'$, such that

$$\Pr[\text{Hash-sec}_{\mathcal{A}',\Pi}(n) = 1] > \text{negl}(n).$$

Construct an experiment for `PPT` adversary $\mathcal{A}$:

1. A key s is generated by running Gen(1^n).
2. Then adversary $\mathcal{A}$ is given s. Then $\mathcal{A}$ uniformly chooses $x \in \{0,1\}^*$ and gives A' s and x.
3. When $\mathcal{A}'$ outputs $x' \in \{0,1\}^*$, $\mathcal{A}$ outputs x, x'.
4. The output is 1 if and only if $x \neq x', H^s(x) = H^s(x')$. (If Π is a fixed-length hash function for inputs of length $l'(n)$, then require $x, x' \in \{0,1\}^{l'(n)}$.)

So $\mathcal{A}$ outputs $1 \Leftrightarrow x \neq x'$ and $H^s(x) = H^s(x') \Leftrightarrow \mathcal{A}'$ outputs 1.
Thus $\Pr[\text{Hash-coll}_{\mathcal{A},\Pi}(n) = 1] = \Pr[\text{Hash-prei}_{\mathcal{A}',\Pi}(n) = 1] > \text{negl}(n)$, a contradiction.

Proof of second preimage resistant to preimage resistant: Proof by contradiction, assume $\Pi = (\mathrm{Gen}, H)$ is not preimage resistant but is second preimage resistant, so there's an PPT adversary $\mathcal{A}'$, such that

$$\Pr[\text{Hash-prei}_{\mathcal{A}',\Pi}(n) = 1] > \mathrm{negl}(n).$$

Construct an experiment for PPT adversary $\mathcal{A}$:

1. A key s is generated by running $\mathrm{Gen}(1^n)$. Then uniformly choose $x \in \{0,1\}^*$.
2. Then adversary $\mathcal{A}$ is given s, x. Compute $y = H^s(x)$. Then, gives A' s and y.
3. When $\mathcal{A}'$ outputs $x' \in \{0,1\}^*$, $\mathcal{A}$ outputs x'.
4. The output is 1 if and only if $x \neq x', H^s(x) = H^s(x')$. (If Π is a fixed-length hash function for inputs of length $l'(n)$, then require $x, x' \in \{0,1\}^{l'(n)}$.)

So $\mathcal{A}'$ outputs 1 $\Leftrightarrow H^s(x) = H^s(x')$, and $\mathcal{A}$ outputs 1 $\Leftrightarrow x \neq x', H^s(x) = H^s(x')$.

Denote set $W = \{x \mid y = H^s(x)$ has only one x hashed to y.$\}$, since $y \in \{0,1\}^{l(n)}$, we have $|W| \leq 2^{l(n)}$. Consider the probability that $x \neq x'$ when $H^s(x) = H^s(x')$, here we assume $2^{l(n)-l'(n)} \leq \mathrm{negl}(n)$, where $l(n) = len(y), l'(n) = len(x)$:

$$\begin{aligned}
&\Pr[x \neq x' \wedge H^s(x) = H^s(x')] \\
=&\Pr[x \neq x' \wedge H^s(x) = H^s(x') \wedge x \notin W] \\
=&\Pr[H^s(x) = H^s(x')] - \Pr[H^s(x) = H^s(x') \wedge x \in W] - \Pr[x = x' \wedge H^s(x) = H^s(x') \wedge x \notin W] \\
=&\Pr[H^s(x) = H^s(x')] - 0 - \Pr[x = x' \wedge H^s(x) = H^s(x') \mid x \notin W] \times \Pr[x \notin W] \\
\geq&\Pr[H^s(x) = H^s(x')] - \frac{1}{2}\Pr[H^s(x) = H^s(x')] \times (1 - 2^{l(n)-l'(n)}) \\
=&\frac{1}{2}\Pr[H^s(x) = H^s(x')] - \mathrm{negl}(n)
\end{aligned}$$

Since $\Pr[H^s(x) = H^s(x')] > \mathrm{negl}(n)$, we have $\Pr[\text{Hash-sec}_{\mathcal{A},\Pi}(n) = 1] = \Pr[x \neq x' \wedge H^s(x) = H^s(x')] > \mathrm{negl}(n)$, a contradiction.

Refute: If we remove the condition that $2^{l(n)-l'(n)} \leq \mathrm{negl}(n)$, then second preimage resistant can not imply preimage resistant. A construction is as followed: assume there is a second preimage resistant $H' : \{0,1\}^{n+1} \to \{0,1\}^n$, then

$$H(x) = \begin{cases} 0x_3 \cdots x_{n+1}, & x_1 = x_2 = 0 \\ 1H'(x_1 \cdots x_{n+1}), & otherwise \end{cases}$$

$H(x)$ is second preimage resistant. But there is probability $\frac{1}{4}$ that given y, $\mathcal{A}$ can invert x. So it not implies preimage resistant.

Problem 5.2. (a). Assume $\Pi_1 = (\mathrm{Gen}_1, H_1)$ is collision resistant. For arbitrary $\mathcal{A}$ of $\Pi = (\mathrm{Gen}, H)$, construct an experiment for PPT adversary $\mathcal{A}_1$ of $\Pi_1 = (\mathrm{Gen}_1, H_1)$ based on it:

1. A key s_1 is generated by running $\mathrm{Gen}(1^n)$.
2. Then adversary $\mathcal{A}_1$ is given s_1. Run $\mathrm{Gen}_2(1^n)$ and get s_2.
3. $\mathcal{A}_1$ gives s_1, s_2 to $\mathcal{A}$, then $\mathcal{A}$ outputs x, x'.
4. $\mathcal{A}_1$ outputs x, x'.
5. The output is 1 if and only if $H_1^{s_1}(x) = H_1^{s_1}(x')$.

Here, $\mathcal{A}$ succeeds $\Leftrightarrow H_1^{s_1}(x) \| H_2^{s_2}(x) = H_1^{s_1}(x') \| H_2^{s_2}(x') \Rightarrow H_1^{s_1}(x) = H_1^{s_1}(x') \Leftrightarrow \mathcal{A}_1$ succeeds. Thus,

$$\Pr[\text{Hash-coll}_{\mathcal{A},\Pi}(n) = 1] \leq \Pr[\text{Hash-coll}_{\mathcal{A}_1,\Pi_1}(n) = 1] \leq \mathrm{negl}(n).$$

So $\Pi = (\mathrm{Gen}, H)$ is collision resistant.

(b). It holds for second preimage resistant.

Assume $\Pi_1 = (\mathrm{Gen}_1, H_1)$ is second preimage resistant. Assume all the theme are fixed-length with input length $l'(n)$. For arbitrary $\mathcal{A}$ of $\Pi(\mathrm{Gen}, H)$, construct an experiment for PPT adversary $\mathcal{A}_1$ of $\Pi_1(\mathrm{Gen}_1, H_1)$ based on it:

1. A key s_1 is generated by running $\text{Gen}(1^n)$, uniformly choose $x \in \{0,1\}^{l'(n)}$.
2. Then adversary $\mathcal{A}_1$ is given s_1, x. Run $\text{Gen}_2(1^n)$ and get s_2.
3. $\mathcal{A}_1$ gives s_1, s_2, x to $\mathcal{A}$, then $\mathcal{A}$ outputs $x' \in \{0,1\}^{l'(n)}$.
4. $\mathcal{A}_1$ outputs x'.
5. The output is 1 if and only if $x \neq x', H_1^{s_1}(x) = H_1^{s_1}(x')$.

Here, $\mathcal{A}$ succeeds $\Leftrightarrow H_1^{s_1}(x) \| H_2^{s_2}(x) = H_1^{s_1}(x') \| H_2^{s_2}(x') \Rightarrow H_1^{s_1}(x) = H_1^{s_1}(x') \Leftrightarrow \mathcal{A}_1$ succeeds. Thus,

$$\Pr[\text{Hash-sec}_{\mathcal{A},\Pi}(n) = 1] \leq \Pr[\text{Hash-sec}_{\mathcal{A}_1,\Pi_1}(n) = 1] \leq \text{negl}(n).$$

So $\Pi = (\text{Gen}, H)$ is second preimage resistant.

It doesn't hold for preimage resistant.
Construct $\Pi_2 = (\text{Gen}_2, H_2)$ as followed:

1. Gen_2: do nothing.
2. H_2: on input $x = \{0,1\}^{n+1}$, output first n bits of x as $H_2(x)$.

Thus, in the experiment of preimage resistant: on input y, we first define the last n bits as y_2, then simply add a uniformly bit $b \in \{0,1\}$ after y_2, define as x. Then with probability $\frac{1}{2}$, the adversary succeeds.

Thus, although Π_1 is preimage resistant, it doesn't work for Π.

Problem 5.3. Yes.

For arbitrary $\mathcal{A}$ of $\Pi = (\text{Gen}, \hat{H})$, construct an experiment for `PPT` adversary $\mathcal{A}'$ of $\Pi' = (\text{Gen}, H)$ based on it:

1. A key s is generated by running $\text{Gen}(1^n)$.
2. Then adversary $\mathcal{A}'$ is given s. $\mathcal{A}'$ gives s to $\mathcal{A}$, then $\mathcal{A}$ outputs $x, x' \in \{0,1\}^*$.
3. $\mathcal{A}'$ checks: if $H^s(x) \neq H^s(x')$, let $x = H^s(x), x' = H^s(x')$. Then $\mathcal{A}'$ output x, x'.
4. The output is 1 if and only if $x \neq x', H^s(x) = H^s(x')$.

In the experiment, if $\mathcal{A}$ succeeds, then $x \neq x'$ and $H^s(H^s(x)) = H^s(H^s(x'))$. If $H^s(x) = H^s(x')$, then x, x' succeeds for $\mathcal{A}$; otherwise we have $H^s(x) \neq H^s(x')$ and $H^s(H^s(x)) = H^s(H^s(x'))$, so $H^s(x), H^s(x')$ succeeds for $\mathcal{A}$.

Thus,

$$\Pr[\text{Hash-coll}_{\mathcal{A},\Pi}(n) = 1] = \Pr[\text{Hash-coll}_{\mathcal{A}',\Pi'}(n) = 1] \leq \text{negl}(n)$$

The inequality holds because $\Pi' = (\text{Gen}, H)$ is collision resistant. So $\Pi = (\text{Gen}, \hat{H})$ is collision resistant.

Problem 5.6. Before answering the questions, prove claim: if hash function $h : \{0,1\}^{2n} \to \{0,1\}^n$ is collision resistant, then construct H as followed:

$$\forall x \in \{0,1\}^{2n}, s, H^s(x) = h^s(x) \oplus c_s.$$

Then H is collision resistant. (Here $c_s \in \{0,1\}^n$.)

Proof. For arbitrary $\mathcal{A}$ of $\Pi = (\text{Gen}, H)$, construct an experiment for `PPT` adversary $\mathcal{A}'$ of $\Pi' = (\text{Gen}, h)$ based on it:

1. A key s is generated by running $\text{Gen}(1^n)$.
2. Then adversary $\mathcal{A}'$ is given s. Then $\mathcal{A}'$ gives s to $\mathcal{A}$,
3. When $\mathcal{A}$ outputs $x, x' \in \{0,1\}^{2n}$, $\mathcal{A}'$ outputs x, x'.
4. The output is 1 if and only if $x \neq x', H^s(x) = H^s(x')$.

Here,

$$\begin{aligned}
\text{Hash-coll}_{\mathcal{A}',\Pi'} = 1 &\Leftrightarrow H^s(x) = H^s(x') \\
&\Leftrightarrow h^s(x) \oplus h^s(0^n \| c_s) = h^s(x') \oplus h^s(0^n \| c_s) \\
&\Leftrightarrow h^s(x) = h^s(x') \\
&\Leftrightarrow \text{Hash-coll}_{\mathcal{A},\Pi} = 1
\end{aligned}$$

Thus, H is collision resistant. □

(a).No.

Fixed $x_0 \in \{0,1\}^n$ Given hash function h, construct H as followed:

$$\forall x \in \{0,1\}^{2n}, s, H^s(x) = h^s(x) \oplus h^s(0^n \| x_0).$$

Thus,

$$\forall s, H^s(0^n \| x_0) = 0^n.$$

For the claim we prove above, H is collision resistant.

Use H as the hash block in Merkle-Damgård transform and define as H_m, we have

$$H_m(x_0 \| x_0) = H(0^n \| x_0) = 0^n = H_m(x_0),$$

a collision.

So it's not collision resistant.

(b).Yes.

Proof by contradiction: If there are $x \neq x'$, such that

$$H^s(x) = z_B \| L = z'_{B'} \| L' = H^s(x'),$$

we have $L = L'$. Assume $x = x_1 \cdots x_B, x' = x'_1 \cdots x'_B$.

Let $I_i = z_{i-1} \| x_i$ denote the ith input to h^s, and set $I_{B+1} = z_B$. Define I'_i analogously with respect to x'.

Let N be the largest index of $\{1, 2 \cdots, B\}$, such that $I_N \neq I'_N$. Since $x \neq x'$, there exists such N.

By the maximization of N, we have $I_{N+1} = I'_{N+1}$, that is $z_N = z'_N$. However, $I_N \neq I'_N$. So we find a collision in h^s.

But h^s is collision resistant, a contradiction. So it's collision resistant.

(c).Yes.

Proof by contradiction: If there are $x \neq x'$, such that

$$H^s(x) = z_B \| L = z'_{B'} \| L' = H^s(x'),$$

we have $L = L'$. Assume $x = x_1 \cdots x_B, x' = x'_1 \cdots x'_B$.

Let $I_i = z_{i-1} \| x_i$ denote the ith input to h^s, and set $I_{B+1} = z_B$. Define I'_i analogously with respect to x'.

Let N be the largest index of $\{1, 2 \cdots, B\}$, such that $I_N \neq I'_N$. Since $x \neq x'$, there exists such N.

- $N > 1$: By the maximization of N, we have $I_{N+1} = I'_{N+1}$, that is $z_N = z'_N$. However, $I_N \neq I'_N$. So we find a collision in h^s.
- $N = 1$: $I_2 = I'_2 \Rightarrow z_1 = z'_1$, and $I_1 \neq I'_1$. we find a collision.

But h^s is collision resistant, a contradiction. So it's collision resistant.

(d).No.

Fixed $x_0 \in \{0,1\}^n$ Given hash function h, construct H as followed:

$$\forall x \in \{0,1\}^{2n}, s, H^s(x) = h^s(x) \oplus h^s(2L \| x_0) \oplus L.$$

Thus,

$$\forall s, H^s(2L \| x_0) = L.$$

For the claim we prove above, H^s is collision resistant.

Use H^s as the hash block in Merkle-Damgård transform and define as H^s_m, we have

$$H^s_m(x_0\|x_0) = H^s(H^s(2L\|x_0)\|x_0) = H^s(L\|x_0) = H^s_m(x_0),$$

a collision.

So it's not collision resistant.

Problem 5.10(a). Randomly choose m, the adversary $\mathcal{A}$ access the oracle $\mathrm{Mac}_{s,k}(\cdot) = H^s(k\|\cdot)$ and get t. Let m' denotes the message after padding m and adding the string length. So m' is exactly the input of hash function.

Assume $L = |m'|$. Then $\mathcal{A}$ compute $h^s(t\|L) = t'$, and output (m', t'). Since

$$H^s(m') = h^s(m'\|L) = h^s(h^s(m)\|L) = h^s(t\|L) = t',$$

we have $\mathrm{Vrfy}_k(m', t') = 1$ and (m', t') was not asked by $\mathcal{A}$.

Thus $\mathcal{A}$ succeeds with probability 1. And it's not a secure Mac.

Problem 5.13. If t is not a power of 2, use an incomplete binary tree. To construct a collision, first randomly choose $(x'_1, \cdots, x'_{2t})$. Get its Merkle tree construction: define the hash values in the first step as $(H(x'_1, x'_2), \cdots, H(x'_{2t-1}, x'_{2t}))$. Then define $(x_1, x_2, \cdots, x_t) = (H(x'_1, x'_2), \cdots, H(x'_{2t-1}, x'_{2t}))$.

Thus, $\mathcal{MT}_t(x_1, x_2, \cdots, x_t) = \mathcal{MT}_{2t}(x'_1, x'_2, \cdots, x'_{2t})$.

Problem 5.14. (a).

Assume $\mathcal{F}, \mathcal{V}, \mathcal{H}$ denotes the set of files, verify codes and the messages saved by clients. A setting $\Pi = (\mathrm{Hash}, \text{Get-Vrfy}, \mathrm{Vrfy})$ is contained of:

- H: $\mathcal{F}^* \to \mathcal{H}$. Given file set $F \subset \mathcal{F}$, the function return a value h that should be saved by the client.
- Get-Vrfy: $\mathcal{F} \to \mathcal{V}$. When a client wants to verify the exists of $f \in \mathcal{F}$, the function return $v \in \mathcal{V}$ for the client to check.
- Vrfy: $\mathcal{F} \times \mathcal{V} \to \{0, 1\}$. Return if the file $f \in \mathcal{F}$ can be verified by $v \in \mathcal{V}$.

Experiment Verify-file $\Pi = (\mathrm{Hash}, \text{Get-Vrfy}, \mathrm{Vrfy})$ of an `PPT` adversary $\mathcal{A}$:

1. Run $\mathcal{A}(1^n)$. $\mathcal{A}$ is given $\Pi = (\mathrm{Hash}, \text{Get-Vrfy}, \mathrm{Vrfy})$.
2. $\mathcal{A}$ outputs f, v.
3. Output 1 if and only if $\mathrm{Vrfy}(f, v) = 1$ and $v \neq \text{Get-Vrfy}(f)$.

Definition 3. *The files that the client saves on the server are* ***secure*** *if and only if*

$$\Pr[\textit{Vrfy-file}_{\mathcal{A},\Pi}(n) = 1] \leq \textit{negl}(n).$$

(b).

The Merkle trees' construction $\Pi = (\mathrm{Hash}, \text{Get-Vrfy}, \mathrm{Vrfy})$ is as followed:

- Hash $= h^s$: Given file set $F = \{f_1, \cdots, f_n\}$. Assume $2^{t-1} < n \leq 2^t$, then set $f_{n+1} = \cdots = f_{2^t} =$ null. Let $n' = 2^t$.
  - Use hash function h^s to compute $h^s(f_1, f_2), \cdots, h^s(f_{n'-1}, f_{n'}) = h_{1,2}, \cdots, h_{n'-1,n'}$.
  - $h^s(h_{1,2}, h_{3,4}), \cdots, h^s(h_{n'-3,n'-2}, h_{n'-1,n'}) = h_{1\cdots 4}, \cdots, h_{n'-3\cdots n'}$.

    $\cdots$
  - $h^s(h_{1\cdots n'/2}, h_{n'/2+1\cdots n'}) = h_{n'/2+1\cdots n'} = h$

  Then the server save all hash values and give h to the client.
- Get-Vrfy: The client wants to verify x_i. Without loss of generality, assume $i = 1$. Then the server give him $x_2, h_{3,4}, h_{5\cdots 8}, \cdots, h_{n'/2+1\cdots n'}$.

  (That is, give the client the other child nodes along the binary tree, such that the client can use these values to compute h.)

- Vrfy: Without loss of generality, assume the client wants to verify x_1. The client computes $h^s(h^s(\cdots(h^s(x_1, h_{3,4})\cdots), h_{n'/2+1\cdots n'})$ equals to h or not.

  If so, the verification succeeds; otherwise, the verification fails.

(c).

If $\Pi' = (\text{Gen}_h, h)$ is collision resistant, then $\forall \mathcal{A}'$,

$$\Pr[\text{Hash-coll}_{\mathcal{A}',\Pi'}(n) = 1] \leq \text{negl}(n).$$

Assume an adversary $\mathcal{A}$ has find a collision in Merkle trees $\Pi = (h, \text{Get-Vrfy}, \text{Vrfy})$. Then we construct an adversary $\mathcal{A}'$ to find a collision in h^s.

To write succinctly, if client asks to verify x_i, denotes the values that the server gives as $(h_1, h_2, \cdots, h_t)$. Denote x_i as h_0. The client should compute:

$$\begin{aligned} h^s(h_0, h_1) &= h'_1 \\ h^s(h'_1, h_2) &= h'_2 \\ &\cdots \\ h^s(h'_{t-1}, h_t) &= h'_t \end{aligned}$$

Then verify if $h'_t = h$.

Assume $v = (h_1, h_2, \cdots, h_t)$ are generated by the correct files. If there are $v^0 = (h_1^0, h_2^0, \cdots, h_t^0) \neq v$, such that $\text{Vrfy}(v') = 1$, we say there is a collision in the construction based on Merkle trees.

Define

$$\begin{aligned} &I_1 = (h_0, h_1) \\ &I_2 = (h'_1, h_2) \\ &\cdots \\ &I_t = (h'_{t-1}, h_t) \\ &I_{t+1} = h'_t \end{aligned}$$

Similarly, define $I_1^0, \cdots, I_{t+1}^0$. Let N be the largest index such that $I_N \neq I_N^0$. Since $v \neq v^0$ there exists $h_i \neq h_i^0$, and exists $I_i \neq I_i^0$, so such N exists.

Since $h'_t = h = h_t^{0'}$, $I_{t+1} = I_{t+1}^0$. Thus $N \leq t$.

For the maximization of N, we have $I_{N+1} = I_{N+1}^0$, so $h'_N = {h_N^0}'$. But $(h'_{N-1}, h_N) = I_i \neq I_i^0 = ({h_{N-1}^0}', h_N^0)$, and there hash value $h'_N = {h_N^0}'$, thus we find a collision in h^s.

Thus a collision in Merkle trees $\Rightarrow$ a collision in h^s:

$$\Pr[\text{Vrfy-file}_{\mathcal{A},\Pi}(n) = 1] \leq \Pr[\text{Hash-coll}_{\mathcal{A}',\Pi'}(n) = 1] \leq \text{negl}(n).$$

Thus, the construction based on Merkle trees is secure.