UNIVERZITA KOMENSKÉHO V BRATISLAVE
Fakulta matematiky, fyziky a informatiky

Kombinátory hašovacích funkcií

Diplomová práca

Vedúci diplomovej práce:
RNDr. Michal Rjaško

Diplomant:
Bc. Marika Mitrengová

Bratislava 2011

UNIVERZITA KOMENSKÉHO V BRATISLAVE
Fakulta matematiky, fyziky a informatiky

Kombinátory hašovacích funkcií

Diplomová práca

Študijný odbor: 9.2.1 informatika
evidenčné číslo: 0a734ff6-bac7-484c-ba5d-3dcab2cee7b1

Vedúci diplomovej práce:
RNDr. Michal Rjaško

Diplomant:
Bc. Marika Mitrengová

Bratislava 2011

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Bc. Marika Mitrengová
Študijný program: informatika (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: 9.2.1. informatika
Typ záverečnej práce: diplomová
Jazyk záverečnej práce: slovenský

Názov: Kombinátory hašovacích funkcií

Cieľ: Na súčasne kryptografické hešovacie funkcie sú kladené silné nároky, existuje veľa rôznych vlastností, ktoré by mala ``dobrá'' hešovacia funkcia spĺňať. Kombinátori zachovávajúce viacero vlastností sú konštrukcie, ktoré umožňujú spojiť dve hešovacie funkcie s dvoma rôznymi vlastnosťami do jednej hešovacej funkcie, ktorá spĺňa obidve vlastnosti. Cieľom práce je skompletizovať výsledky v tejto oblasti, t.j. analyzovať existujúce konštrukcie kombinátorov voči ďalším vlastnostiam (ktoré neboli analyzované v pôvodných prácach) a prípadne navrhnúť nové konštrukcie, ktoré by tieto vlastnosti zachovávali.

Vedúci: RNDr. Michal Rjaško

Dátum zadania: 20.10.2010

Dátum schválenia: 25.10.2010

prof. RNDr. Branislav Rovan, PhD.
garant študijného programu

Mitrengová

študent

vedúci

Čestné prehlásenie

Vyhlasujem, že som diplomovú prácu vypracovala samostatne s použitím uvedenej odbornej literatúry.

Bratislava 05. 05. 2011 .....

Vlastnoručný podpis

Poďakovanie

Chcela by som poďakovať vedúcemu tejto diplomovej práce RNDr. Michalovi Rjaškovi za cenné rady a usmerňovanie počas písania práce. Moja veľká vďaka patrí tiež rodine za ich podporu, Ivke Selečéniovej a Ivanovi Kováčovi za ich pomoc.

Abstrakt

V práci sa venujeme rodinám hašovacích funkcií a robustným kombinátorom, ktoré zachovávajú viacero vlastností. V prvej časti uvádzame základné vlastnosti rodín hašovacích funkcií. V ďalšej časti prezentujeme výsledky pre už známe robustné kombinátory, ktoré rozšírime o naše zistenia. Pre tieto kombinátory dokazujeme zachovávanie vlastností, ktoré neboli dokázané. V závere našej práce navrhneme konštrukciu kombinátora vlastností, ktorá kombinuje vlastnosti čiastkových funkcií.

Kľúčové slová: kryptografická hašovacia funkcia, rodina hašovacích funkcií, robustný kombinátor.

Abstract

In this thesis we present families of hash functions and strong multi property preserving combiners. In the first part we analyze basic properties of families of hash functions. Next we present achievements for known combiners which we will extend about our discoveries. For these combiners we prove preservation of properties which were not proved. In the end of our thesis we proposed a construction of a combiner which combine several properties of partial functions.

Keywords: cryptographic hash function, family of hash functions, strong multi property preserving combiner.

Obsah

Zoznam obrázkov

Zoznam tabuliek

Úvod

Komunikácia v dnešnej dobe obsahuje dôverné informácie, preto ju treba chrániť. Avšak nemusíme chrániť iba komunikáciu, ale aj zálohované informácie na diskoch. Potrebná je aj autentifikácia používateľov, čo sa využíva napr. v elektronickom bankovníctve. Dôležité sú aj metódy potvrdzujúce celistvosť informácie, aby nemohla byť zmenená bez nášho povšimnutia. V elektronickom obchode potrebujeme zabezpečiť, aby sme nemohli poprieť, že sme si niečo objednali. Hašovacie funkcie sú zo všetkých kryptografických primitív najuniverzálnejšie a sú používané pri riešení spomenutých problémov.
Hašovacia funkcia je funkcia, ktorá zoberie ako vstup reťazec ľubovoľnej dĺžky a transformuje ho na výsledok fixnej dĺžky, výstup funkcie nazývame odtlačok. Táto vlastnosť sa využíva napríklad v digitálnych podpisoch na kompresiu správy kvôli efektívnosti. Namiesto podpísania správy M sa podpisuje odtlačok $H(M)$, kde H je nejaká hašovacia funkcia použitá v podpisovej schéme. Podpisovanie odtlačku $H(M)$ je oveľa jednoduchšie a rýchlejšie ako podpisovanie správy M, pretože dĺžka odtlačku je výrazne kratšia. Od hašovacej funkcie použitej v digitálnych podpisoch požadujeme, aby bola odolná voči kolíziám a mala vlastnosť odolnosť druhého vzoru. Ak by nebola odolná voči kolízii, útočník by vedel nájsť dve rôzne správy, ktoré majú rovnaký odtlačok, a teda aj podpis. Odolnosť druhého vzoru zabezpečí, aby útočník k už podpísanej správe M nevedel s veľkou pravdepodobnosťou nájsť inú správu M', ktorá by mala rovnaký odtlačok. Ak by takú správu M' získal, mal by k nej korektný podpis.

Jedno z najjednoduchších použití hašovacích funkcií na autentifikáciu je chránenie hesiel. UNIX-ové systémy aplikujú hašovacie funkcie na heslá používateľov a ukladajú si ich odtlačky, nie heslo samotné. Na autentifikáciu používateľa je potrebné heslo, ktoré zadá používateľ do systému, systém ho zahašuje a porovná zadané zahašované heslo s uloženou hodnotou v systéme. Ak sa zhodujú, používateľ je úspešne autentifikovaný. Na správne fungovanie takejto autentifikácie požadujeme, aby použitá hašovacia funkcia bola nein-

vertovateľná. Ak by sa dala funkcia ľahko invertovať, útočník by vedel ľahko získať heslo k danému odtlačku.

Sťahovanie softvéru zo siete využíva hašovacie funkcie na kontrolu integrity a autentickosti dát. Spolu so sťahovanými dátami je posielaný aj MAC, čo je odtlačok hašovacej funkcie, ktorá bola parametrizovaná tajným kľúčom. Príslušný tajný kľúč zdieľajú obe komunikujúce strany. Používateľ si prijaté dáta zahašuje a porovná, či sa odtlačok prijatej správy zhoduje s príslušným MAC-om. Ak sa zhodujú, dáta neboli zmenené.

Veľa kryptografických schém (medzi ktoré patria aj hašovacie funkcie) je založených na nedokázaných predpokladoch o zložitosti nejakého výpočtového problému. Existujú predpoklady o zložitosti problémov, ktorých platnosť je podporovaná desiatkami výskumov (napríklad diskrétny logaritmus, faktorizácia). Avšak veľa nových výskumov ponúka nové možnosti riešenia problémov. Často sa nevieme rozhodnúť, ktoré predpoklady sú dôveryhodné. Teda pri viacerých implementáciách kryptografických primitív (každé je založené na rôznych predpokladoch) je ťažké rozhodnúť, ktorá implementácia je najlepšia. Takisto aj viaceré útoky na hašovacie funkcie odolné voči kolízii [21] [22] [10] [23] nastolili otázku, ako dosiahnuť konštrukciu, ktorá je viac bezpečná. Riešením sú robustné kombinátory. Robustný kombinátor zoberie ako vstup viacero kandidátov a skonštruuje schému, ktorej bezpečnosť je garantovaná, ak aspoň nejakí kandidáti sú bezpeční. Takže výsledná schéma je bezpečná tak dlho, ako je bezpečných dostatočne veľa použitých primitív. Táto metóda poskytuje toleranciu proti zlým predpokladom, pretože zlomenie algoritmu nejakého kandidáta nerobí kombinovanú schému neistú.

V tejto práci sa budeme venovať rodinám hašovacích funkcií, ich vlastnostiam a kombinovaniu rodín hašovacích funkcií. Naša práca je rozdelená na tri kapitoly. V prvej kapitole sa venujeme vlastnostiam rodín hašovacích funkcií, vzťahom medzi vlastnosťami a pojmu kombinátor. Druhá kapitola obsahuje konštrukcie robustných kombinátorov C_{4P} a C_{6P}, ktoré zachovávajú viaceré vlastnosti. Autormi týchto kombinátorov sú Fischlin, Lehmann a Pietrzak. K daným kombinátorom dokážeme, že zachovávajú aj niektoré iné vlastnosti, ktorých zachovávanie ešte nebolo dokázané. V tabuľke 1 je znázornený prehľad výsledkov autorov a našich výsledkov.

Tretia kapitola obsahuje konštrukcie kombinátorov, ktoré navrhol Rjaško v [16]. Jeho vý-

	Coll	Sec	aSec	eSec	Pre	aPre	ePre	Ctfp	aCtfp	Mac	Prf	Pro
C_{4P}	•	√	√	•	×	?	√	√	√	•	•	×
C_{6P}	•	√	√	•	•	?	√	√	√	•	•	•

Tabuľka 1: Prehľad predchádzajúcich a našich výsledkov pre kombinátory C_{4P} a C_{6P}. Symbol • znamená, že robustnosť kombinátora bola dokázaná, √ robustnosť sme dokázali my, × bolo dokázané, že nie je robustný a ? symbolizuje otvorený problém.

	Coll	Sec	aSec	eSec	Pre	aPre	ePre	Ctfp	aCtfp	Mac	Prf	Pro
C_1	•	⇝	√	⇝	⇝	⇝	•	⇝	√	⇝	•	□
C_2	•	⇝	×	⇝	⇝	□	•	⇝	□	□	□	□
C_3	•	⇝	√	⇝	⇝	⇝	√	⇝	√	□	□	□
C_4	√	⇝	√	⇝	⇝	⇝	√	⇝	√	⇝	√	□

Tabuľka 2: Prehľad predchádzajúcich a našich výsledkov pre konštrukcie C_1, C_2, C_3 a C_4. Symbol • znamená, že bezpečnosť bola dokázaná, √ bezpečnosť sme dokázali my, × dokázali sme, že nie je bezpečný, □ bezpečnosťou sme sa nezaoberali a ⇝ bezpečnosť vyplýva z bezpečnosti pre inú vlastnosť.

sledky sme rozšírili a navrhli sme novú konštrukciu C_4. V tabuľke 2 je uvedený prehľad predchádzajúcich a našich výsledkov.

Kapitola 1

Vlastnosti rodín hašovacích funkcií

V tejto kapitole sa budeme venovať rodinám hašovacích funkcií a ich vlastnostiam. Vychádzali sme z článkov [12, 3, 5, 2, 6, 9]. Rodiny hašovacích funkcií sú dôležité kryptografické primitíva, ktoré v posledných rokoch pritiahli veľa výskumov, najmä po tom, čo pomerne často používané hašovacie funkcie ako MD5 a SHA-1 boli zlomené [21, 22]. Simon v [19] dokázal, že hašovacie funkcie odolné voči kolíziám nemôžu byť konštruované z jednosmerných funkcií cez tzv. „black box“ redukciu. Merkle a Damgård navrhli konštrukciu, kde iterovaním hašovacích funkcií odolných voči kolíziám s fixnou vstupnou dĺžkou dostaneme hašovaciu funkciu odolnú voči kolízii pre vstup rôznej dĺžky. Avšak Coron, Dodis, Malinaud a Puniya [4] ukázali, že Merkle-Damgård konštrukcia sa nespráva dostatočne náhodne, aj keď je inštanciovaná s náhodnou funkciou. Joux [8] dokázal, že hľadanie viacerých hodnôt, ktoré sa zahašujú na tú istú hodnotu pre iterované hašovacie funkcie nie je o veľa ťažšie ako hľadanie bežnej kolízie.

1.1 Označenia a štandardná notácia

Definícia 1.1.1 (Hašovacia funkcia)**.** Nech $\mathcal{M} = \{0,1\}^*$, $\mathcal{Y} = \{0,1\}^n$ a $n > 0$ je prirodzené číslo. Hašovacia funkcia je výpočtovo efektívna funkcia $H : \mathcal{M} \to \mathcal{Y}$. Množina $\mathcal{M}$ sa nazýva priestor správ a množina $\mathcal{Y}$ priestor hašov (odtlačkov).

V tejto práci budeme používať nasledujúce označenia: $M \xleftarrow{\$} S$ znamená náhodný výber prvku z množiny S. Ak je S konečná množina, M je vyberané uniformne z S. Zreťazenie reťazcov M_1 a M_2 budeme označovať $M_1 \parallel M_2$. Označenie $\langle i \rangle_r$ symbolizuje r-bitový reťazec čísla i zapísaného v binárnej sústave. *Náhodné orákulum* je funkcia $f : M \to Y$ uniformne

vybraná z množiny všetkých funkcií s definičným oborom hodnôt M a oborom hodnôt Y. Náhodný výber orákula budeme označovať $f \xleftarrow{\$} Func(M, Y)$.

Namiesto konkrétnej hašovacej funkcie budeme často používať rodinu hašovacích funkcií kvôli jej univerzálnosti.

Definícia 1.1.2 (Rodina hašovacích funkcií)**.** Nech $\mathcal{K} = \{0,1\}^k$, $\mathcal{M} = \{0,1\}^*$ a $\mathcal{Y} = \{0,1\}^n$ kde $k, n > 0$ sú prirodzené čísla. Rodina hašovacích funkcií je funkcia $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$. Množinu $\mathcal{K}$ voláme priestor kľúčov, k dĺžka kľúča, $\mathcal{M}$ priestor správ, $\mathcal{Y}$ priestor hašov a n dĺžka hašu.

Rodina hašovacích funkcií sa využíva pri konštrukcii MAC (je to hašovacia funkcia parametrizovaná súkromným kľúčom). So správou M je poslaný aj autentifikačný kód $H_k(M) = C$, overovateľ si potom môže overiť, či $C = H_k(M)$ pre prijatú správu M a autentizačný kód C. Ďalej je rodina hašovacích funkcií riešením tzv. *základnej dilemy hašovania (foundations-of-hashing dilemma)*. V kryptografickej praxi hašovacia funkcia odolná voči kolíziám mapuje ľubovoľne dlhý reťazec na reťazec fixnej dĺžky. Avšak kryptografická teória používa funkcie odolné voči kolízii rozšírené o kľúč, pretože každý kľúč špecifikuje inú funkciu. Formálne definovanie odolnosti voči kolíziám pre nekľúčové hašovacie funkcie nefunguje. Ku každej funkcii $H : \{0,1\}^* \to \{0,1\}^n$ vždy existuje algoritmus, ktorý dá kolíziu. Tento algoritmus ju má „zadrôtovanú“ v sebe, a teda je krátky a rýchly. Z Dirichletovho princípu musia existovať reťazce $X \neq X'$ také, že $H(X) = H(X')$. Pre človeka môže byť ťažké nájsť takýto kolízny pár, a teda aj napísať program. To znamená, že hašovacia funkcia je odolná voči kolíziám, nie však, ak existuje efektívny algoritmus, ktorý dá kolíziu, ale ak existuje efektívny algoritmus známy ľuďom, ktorý dá kolíziu pre H. Viac v [1] a [17]. Vďaka rodine hašovacích funkcií vieme definovať iné kryptografické vlastnosti (pseudonáhodnosť, pseudonáhodné orákulum a mnohé iné). Nevýhodou použitia rodiny hašovacích funkcií je strata efektívnosti, lebo je potrebných ďalších k bitov na spracovanie každej správy.

Definícia 1.1.3 (Útočník)**.** Útočník je algoritmus, ktorý má ľubovoľne veľa vstupov. Niektoré vstupy môžu byť dlhé reťazce, preto predpokladáme, že útočník môže čítať i-tý bit j-tého vstupu v konštantnom čase. Označme písmenom $\mathcal{A}$ útočníka, potom $\mathbf{Adv}_H^{\text{xxx}}(\mathcal{A})$ je miera útočníkovej výhody definovaná ako pravdepodobnosť, že $\mathcal{A}$ uspeje na nejakom vstupe, pri riešení daného problému xxx pre funkciu H. Čas behu útočníka na nejakom vstupe je priemerný čas potrebný na výpočet výsledku plus veľkosť $\mathcal{A}$.

Napríklad majme vlastnosť odolnosť voči kolíziám, označme ju Coll. Výhodu útočníka

$\mathcal{A}$ pri hľadaní kolízií (pravdepodobnosť, že pre náhodne zvolený kľúč K útočník $\mathcal{A}$ nájde také správy M a M', pričom $M \neq M'$ a $H_K(M) = H_K(M')$) v rodine hašovacích funkcií H budeme označovať $\mathbf{Adv}_H^{\text{Coll}}(\mathcal{A})$.

Veľkosť algoritmu $\mathcal{A}$ je zahrnutá v čase behu útočníka kvôli nasledujúcej konštrukcii útočníka. Predpokladajme, že konštruujeme útočníka, ktorý útočí na vlastnosť jednosmernosť funkcie $H : \mathcal{M} \to \mathcal{Y}$. Teda pre dané Y hľadá správu M takú, že $H_K(M) = Y$. Algoritmus $\mathcal{A}$ by obsahoval tabuľku, ktorá obsahuje všetky hodnoty z Y a k nim príslušné vzory. Hľadanie vzoru by potom znamenalo pozrieť sa do tabuľky, aká hodnota sa zahašovala na konkrétny prvok. Útočník, ktorý má vo svojom tele zahrnutú vyššie spomenutú tabuľku, je neefektívny kvôli priestorovej zložitosti. Preto budeme uvažovať nasledujúceho útočníka.

Definícia 1.1.4 (Efektívny útočník)**.** Nech $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií. Efektívny útočník je útočník, pre ktorého existuje polynóm p taký, že čas behu útočníka je $O(p(k + y + l))$, kde l je dĺžka vstupu útočníka.

Definícia 1.1.5 (Zanedbateľná funkcia)**.** Funkcia $f : N \to R^+$ je zanedbateľná, ak pre každú konštantu $c > 0$ existuje číslo $N_0 \in N$ také, že $\forall n \in N, n > N_0$ platí $f(n) < \frac{1}{n^c}$

Funkcia je zanedbateľná, ak klesá rýchlejšie ako ktorákoľvek polynomiálna funkcia umocnená na -1. Nech xxx označuje nejakú vlastnosť rodín hašovacích funkcií H a $\mathcal{A}$ je útočník útočiaci na H v zmysle xxx, budeme hovoriť, že H je xxx bezpečná, ak $\mathbf{Adv}_H^{\text{xxx}} \leq f(n)$, kde f je zanedbateľná funkcia.

1.2 Vlastnosti hašovacích funkcií

Hašovacia funkcia môže mať viacero vlastností (označujeme $P_1, P_2, \ldots, P_n$), z ktorých najdôležitejšie sú:

Definícia 1.2.1 (Odolnosť voči kolíziám - collision resistance - Coll)**.** Nech $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií a nech $\mathcal{A}$ je útočník. Rodina hašovacích funkcií sa nazýva odolná voči kolízii, ak pre každého efektívneho útočníka $\mathcal{A}$ je nasledujúca výhoda zanedbateľná:

$$\mathbf{Adv}_H^{\text{Coll}}(\mathcal{A}) = \Pr\left[K \xleftarrow{\$} \mathcal{K}; (M, M') \leftarrow \mathcal{A}(K) : (M \neq M') \wedge (H_K(M) = H_K(M'))\right]$$

Hovoríme, že rodina hašovacích funkcií H je $(t, \varepsilon) - Coll$, ak každý útočník $\mathcal{A}$ bežiaci v čase najviac t, má výhodu $\mathbf{Adv}_H^{\text{Coll}}(\mathcal{A}) \leq \varepsilon$.

Coll znamená, že je ťažké nájsť dve hodnoty z definičného oboru hodnôt funkcie také, že ich haš má rovnakú hodnotu. Táto vlastnosť sa využíva v digitálnych podpisoch. Ak by rodina hašovacích funkcií H používaná v digitálnych podpisoch nebola odolná voči kolíziám, potom môžeme nájsť dve správy M_1 a M_2 také, že $H_K(M_1) = H_K(M_2)$. Ak jednu z týchto správ podpíše používateľ, nech je to M_1 (pre M_2 podobne), digitálny podpis správy M_1 je zároveň korektným podpisom správy M_2.

Mauer, Renner a Holenstein sa v [12] venovali pojmom *neodlíšiteľnosť* a *nerozoznateľnosť*. Často potrebujeme dokázať bezpečnosť nejakého kryptosystému $\mathcal{C}(\mathcal{S})$ obsahujúceho komponent $\mathcal{S}$. Zväčša sa to dokazuje tak, že vezmeme systém $\mathcal{C}(\mathcal{T})$ získaný z $\mathcal{C}(\mathcal{S})$ tak, že komponent $\mathcal{S}$ je nahradený idealizovaným komponentom $\mathcal{T}$. Systém $\mathcal{C}(\mathcal{S})$ je bezpečný, ak je bezpečný systém $\mathcal{C}(\mathcal{T})$ a komponent $\mathcal{S}$ je neodlíšiteľný od komponentu $\mathcal{T}$.
Dva systémy $\mathcal{S}$ a $\mathcal{T}$ sú neodlíšiteľné, ak žiaden efektívny algoritmus $\mathcal{D}(\cdot)$ komunikujúci s $\mathcal{S}$ alebo $\mathcal{T}$ nevie rozoznať, či komunikuje s $\mathcal{S}$ alebo $\mathcal{T}$.
Napr. nech $\mathcal{T}$ je zdroj skutočne náhodných bitov (tajný pre dve komunikujúce strany A a B) a nech $\mathcal{S}$ je pseudonáhodný generátor bitov (so súkromným kľúčom zdieľaným s A a B). Ak $\mathcal{C}(\cdot)$ značí šifrovanie založené na operácii XOR (napr. $\mathcal{C}(\mathcal{T})$ je one-time pad a $\mathcal{C}(\mathcal{S})$ označuje aditívnu prúdovú šifru s prúdovým generátorom kľúčov S), potom bezpečnosť $\mathcal{C}(\mathcal{S})$ vyplýva z bezpečnosti $\mathcal{C}(\mathcal{T})$ a faktu, že pre každého efektívneho útočníka (rozlišovača) sa $\mathcal{S}$ správa v podstate ako $\mathcal{T}$, teda $\mathcal{S}$ a $\mathcal{T}$ sú výpočtovo neodlíšiteľné.

Definícia 1.2.2 (Neodlíšiteľnosť)**.** Nech I a F sú dva kryptosystémy, hovoríme, že I a F sú neodlíšiteľné, ak pre žiadneho efektívneho útočníka $\mathcal{D}$ je nasledujúca výhoda zanedbateľná:

$$|\Pr[1 \leftarrow \mathcal{D}(I)] - \Pr[1 \leftarrow \mathcal{D}(F)]|$$

Toto platí za predpokladu, že každý komponent kryptosystému je prístupný len nejakej špecifickej časti, ktorá má k nej prístup. Nech $\mathcal{T}$ z predchádzajúceho príkladu je náhodné orákulum $\mathcal{R}$ (napr. verejne dostupná náhodná funkcia) a nech $\mathcal{S}$ je náhodná hašovacia funkcia $\mathcal{H}(\mathcal{F})$, kde $\mathcal{H}$ je hašovací algoritmus závislý na verejnom parametri $\mathcal{F}$. V kontraste s pseudonáhodnosťou (kde je parameter tajný), žiadna hašovacia funkcia nemôže implementovať náhodné orákulum v predchádzajúcom zmysle, dokázali to Canetti, Goldreich a Halevi [3]. Teda existuje kryptosystém $\mathcal{C}(\cdot)$ taký, že $\mathcal{C}(\mathcal{R})$ je bezpečný, zatiaľčo $\mathcal{C}(\mathcal{H}(\mathcal{F}))$

nie je bezpečný pre žiadny hašovací algoritmus $\mathcal{H}$. Mauer, Renner a Holenstein zovšeobecnili pojem neodlíšiteľnosť na nerozoznateľnosť. Nerozoznateľnosť znamená, že nejaký kryptosystém $\mathcal{C}(\mathcal{T})$ založený na komponente $\mathcal{T}$ nie je ovplyvnený, keď sa komponent $\mathcal{T}$ nahradí komponentom $\mathcal{S}$. Rozoznateľnosť $\mathcal{S}$ od $\mathcal{T}$ implikuje existenciu kryptosystému $\mathcal{C}(\cdot)$, pre ktorý toto nahradenie komponent nie je možné, teda $\mathcal{C}(\mathcal{T})$ je bezpečný, ale $\mathcal{C}(\mathcal{S})$ nie.

Definícia 1.2.3 (Nerozoznateľnosť)**.** Nech H^1 (resp. I^1) je súkromný komponent a H^2 (resp. I^2) verejný komponenta kryptosystému H (resp. I). Kryptosystém H je nerozoznateľný od I, ak pre každého efektívneho útočníka $\mathcal{D}$ existuje simulátor S taký, že výhoda

$$|\Pr[1 \leftarrow \mathcal{D}(H^1, H^2)] - \Pr[1 \leftarrow \mathcal{D}(I^1, S(I^2))]|$$

je zanedbateľná. Simulátor S je algoritmus, ktorý simuluje verejný komponent H^2, aby sťažil odlišovanie H^1 a H^2 od I^1 a I^2.

Hovoríme, že funkcia H je pseudonáhodná, ak útočník nevie rozoznať, či komunikuje s H alebo s náhodnou funkciou f, teda H je neodlíšiteľná od f.

Definícia 1.2.4 (Pseudonáhodnosť - Pseudorandomness - PRF)**.** Nech $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií a nech $\mathcal{A}$ je útočník. Hašovacia funkcia sa nazýva pseudonáhodná, ak pre každého efektívneho útočníka $\mathcal{A}$ je nasledujúca výhoda zanedbateľná

$$\mathbf{Adv}_H^{\mathrm{Prf}}(\mathcal{A}) = |\Pr[K \xleftarrow{\$} \mathcal{K}; 1 \leftarrow \mathcal{A}^{H(K,\cdot)}(H)] - \Pr[f \xleftarrow{\$} Func(M, Y); 1 \leftarrow \mathcal{A}^f(H)]|.$$

Hovoríme, že H je (t, q, ε)-Prf, ak pre každého efektívneho útočníka $\mathcal{A}$ bežiaceho v čase najviac t a pýtajúceho sa orákula najviac q dotazov, je výhoda $\mathbf{Adv}_H^{\mathrm{Prf}}(\mathcal{A}) \leq \varepsilon$.

Definícia 1.2.5 (Nerozoznateľnosť od náhodného orákula - Pseudorandom oracle - Pro)**.** Nech $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií a nech $\mathcal{A}$ je útočník. Rodina hašovacích funkcií H_K^f je nerozoznateľná od náhodného orákula F, ak pre každého efektívneho útočníka $\mathcal{A}$ existuje efektívny algoritmus S (simulátor) taký, že výhoda

$$\begin{aligned}\mathbf{Adv}_{H,f,S}^{\mathrm{Pro}}(\mathcal{A}) = |\Pr[K \xleftarrow{\$} \mathcal{K}; 1 \leftarrow \mathcal{A}^{H_K^f(\cdot), f(\cdot)}(K)] - \\ - \Pr[K \xleftarrow{\$} \mathcal{K}; \mathcal{F} \xleftarrow{\$} Func(\mathcal{M}, \mathcal{Y}); 1 \leftarrow \mathcal{A}^{\mathcal{F}(\cdot), S^{\mathcal{F}}(K,\cdot)}(K)]|\end{aligned}$$

je zanedbateľná.
Hovoríme, že H je $(t_{\mathcal{A}}, t_S, q_1, q_2, \varepsilon)$-Pro, ak pre každého efektívneho útočníka $\mathcal{A}$ bežiaceho

v čase najviac $t_{\mathcal{A}}$ pýtajúceho sa najviac $q_1(g_2)$ dotazov jeho prvého(druhého) orákula existuje simulátor S bežiaci v čase t_S, je výhoda $\mathbf{Adv}^{\mathrm{Pro}}_{H,f,S}(\mathcal{A}) \leq \varepsilon$.

Simulátor S simuluje f, takže žiaden útočník nevie rozoznať, či interaguje s H_K a f alebo s $\mathcal{F}$ a $S^{\mathcal{F}}$.

Definícia 1.2.6 (Nefalšovateľnosť - Message authentication - MAC)**.** Nech $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií. Hašovacia funkcia je MAC, ak pre každého efektívneho útočníka $\mathcal{A}$ je nasledujúca výhoda zanedbateľná:

$$\mathbf{Adv}^{\mathrm{MAC}}_{H} = Pr\big[K \xleftarrow{\$} \mathcal{K}; (M, Y) \leftarrow \mathcal{A}^{H_K} : H_K(M) = Y \wedge \text{ na } M \text{ sa nebolo pýtané}\big]$$

Vlastnosť MAC sa využíva pri zachovaní integrity a autenticity dát. Napríklad pri distribúcii softvérových balíkov potrebujeme vedieť, či prijatý softvér nebol počas prenosu zmenený a či je autorom komunikujúca strana. S daným softvérom sa posiela aj príslušný MAC, teda odtlačok správy (posielaného softvéru), ktorý závisí od správy, ale aj od kľúča. Kľúč poznajú len komunikujúce strany. Na kontrolu integrity a autenticity stačí vypočítať odtlačok prijatej správy a porovnať ho s prijatým MAC-om či sa zhodujú.

Definícia 1.2.7 (Odolnosť prvého vzoru - Preimage resistance - Pre)**.** Nech $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií a $\{0,1\}^{\mathrm{m}} \subseteq \mathcal{M}$. Rodina hašovacích funkcií je Pre, ak pre každého efektívneho útočníka $\mathcal{A}$, je výhoda

$$\mathbf{Adv}^{\mathrm{Pre}[m]}_{H}(\mathcal{A}) = Pr\big[K \xleftarrow{\$} \mathcal{K}; M \xleftarrow{\$} \{0,1\}^m; Y \leftarrow H_K(M); M' \leftarrow \mathcal{A}(K, Y) : H_K(M') = Y\big]$$

zanedbateľná pre všetky $m \in \mathbb{N}$.
Hovoríme, že rodina hašovacích funkcií H je (t, ε)-Pre, ak každý útočník $\mathcal{A}$ bežiaci v čase najviac t má výhodu $\mathbf{Adv}^{\mathrm{Pre}[m]}_{H}(\mathcal{A}) \leq \varepsilon$ pre všetky m také, že $\{0,1\}^m \subseteq \mathcal{M}$.

Parameter m sa pridáva k výhode útočníka kvôli ohraničeniu dĺžky náhodne vybranej správy.
Vlastnosť odolnosť prvého vzoru (jednosmernosť) hovorí o tom, ako ťažko je útočník schopný nájsť vzor pre hodnotu z oboru hodnôt hašovacej funkcie.

Niektoré vlastnosti rodiny hašovacích funkcií sa môžu maximalizovať cez nejakú veličinu, môžeme si predstaviť, že ju vie útočník. Sú to vlastnosti *odolnosť prvého vzoru*,

odolnosť druhého vzoru a *odolnosť prvého vzoru s voliteľným cieľom a vynúteným prefixom*. U týchto vlastností definujeme dve ekvivalentné výhody útočníka, ktoré sa líšia tým, že jedna má tzv. dvojstavového útočníka. Ekvivalenciu týchto výhod dokážeme pre vlastnosť vždy odolnosť prvého vzoru s voliteľným cieľom a vynúteným prefixom v 1.3, dôkazy ekvivalencie pre ostatné vlastnosti sú podobné.

V niektorých z týchto definícií vracia útočník napríklad okrem správy M aj stav, označuje sa S. Je to reťazec ľubovoľnej dĺžky, kde si útočník môže uložiť nejakú informáciu, ktorá mu pomôže v ďalšej časti výpočtu, napríklad kľúč K.

Vlastnosť Pre sa môže maximalizovať cez množinu všetkých odtlačkov $\mathcal{Y}$, dostaneme *všade odolnosť prvého vzoru*, alebo cez množinu všetkých kľúčov $\mathcal{K}$ a dostaneme *vždy odolnosť prvého vzoru*.

Definícia 1.2.8 (Všade odolnosť prvého vzoru - Everywhere preimage resistance - ePre)**.** Nech $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií a nech $\mathcal{A}$ je útočník. Definujme tieto dve výhody:

$$\mathbf{Adv}_H^{\mathrm{ePre}}(\mathcal{A}) = \max_{Y \in \mathcal{Y}} \left\{ \Pr\left[K \xleftarrow{\$} \mathcal{K}; M \leftarrow \mathcal{A}(K) : H_K(M) = Y\right] \right\}$$
$$\mathbf{Adv}_H^{\mathrm{ePre}}(\mathcal{A}) = \Pr\left[(Y, S) \leftarrow \mathcal{A}; K \xleftarrow{\$} \mathcal{K}; M \leftarrow \mathcal{A}(K, S) : H_K(M) = Y\right]$$

Rodina hašovacích funkcií je ePre, ak pre každého efektívneho útočníka $\mathcal{A}$, sú dané výhody zanedbateľné.
Hovoríme, že rodina hašovacích funkcií H je (t, ε)-ePre, ak každý útočník $\mathcal{A}$ bežiaci v čase najviac t, má výhodu $\mathbf{Adv}_H^{\mathrm{ePre}}(\mathcal{A}) \leq \varepsilon$.

Predstavme si, že máme rodinu hašovacích funkcií, ktorá pre každý kľúč K správu 0^m zahašuje na 0^y ($\forall K \ H_K(0^m) = 0^y$). Pre reťazec 0^y nie je ťažké nájsť správu, ktorá sa naň zahašovala. Vlastnosť ePre zosilňuje vlastnosť Pre v zmysle, že je ťažké nájsť vzor pre každý obraz.

Definícia 1.2.9 (Vždy odolnosť prvého vzoru - Always preimage resistance - aPre)**.** Nech

$H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií a $\{0,1\}^{\mathrm{m}} \subseteq \mathcal{M}$. Definujme výhody:

$$\mathbf{Adv}_H^{\mathrm{aPre}[m]}(\mathcal{A}) = \max_{K \in \mathcal{K}} \left\{ \Pr\left[M \xleftarrow{\$} \{0,1\}^m; Y \leftarrow H_K(M); M' \leftarrow \mathcal{A}(Y) : H_K(M') = Y\right] \right\}$$

$$\mathbf{Adv}_H^{\mathrm{aPre}[m]}(\mathcal{A}) = \Pr\left[(K,S) \leftarrow \mathcal{A}; M \xleftarrow{\$} \{0,1\}^m; Y \leftarrow H_K(M); \right.$$
$$\left. M' \leftarrow \mathcal{A}(Y,S) : H_K(M') = H_K(M)\right]$$

Rodina hašovacích funkcií je aPre, ak pre každého efektívneho útočníka $\mathcal{A}$, sú obe výhody zanedbateľné pre všetky $m \in \mathbb{N}$.
Hovoríme, že rodina hašovacích funkcií H je (t,ε)-aPre, ak každý útočník $\mathcal{A}$ bežiaci v čase najviac t, má výhodu $\mathbf{Adv}_H^{\mathrm{aPre}}(\mathcal{A}) \leq \varepsilon$ pre všetky m také, že $\{0,1\}^m \subseteq \mathcal{M}$.

Predstavme si rodinu hašovacích funkcií, ktorá pre kľúč K_0 mapuje každú správu M na 0^y ($\forall M\ H_{K_0}(M) = 0^y$). To znamená, že pre kľúč K_0 nie je ťažké nájsť vzor k odtlačku 0^y. Vlastnosť aPre zosilňuje Pre v zmysle, že je ťažké nájsť vzor pre ktorúkoľvek funkciu H_K.

Definícia 1.2.10 (Odolnosť druhého vzoru - Second preimage resistance - Sec)**.** Nech $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií a $\{0,1\}^m \subseteq \mathcal{M}$. Rodina hašovacích funkcií je Sec, ak pre každého efektívneho útočníka $\mathcal{A}$, je výhoda

$$\mathbf{Adv}_H^{\mathrm{Sec}[m]}(\mathcal{A}) = \Pr\left[K \xleftarrow{\$} \mathcal{K}; M \xleftarrow{\$} \{0,1\}^m; M' \leftarrow \mathcal{A}(K,M) : \right.$$
$$\left. (M \neq M') \wedge (H_K(M) = H_K(M'))\right]$$

zanedbateľná pre $m \in \mathbb{N}$.
Hovoríme, že rodina hašovacích funkcií H je (t,ε)-Sec, ak každý útočník $\mathcal{A}$ bežiaci v čase najviac t má výhodu $\mathbf{Adv}_H^{\mathrm{Sec}}(\mathcal{A}) \leq \varepsilon$ pre všetky m také, že $\{0,1\}^m \subseteq \mathcal{M}$.

Neformálne povedané, rodina hašovacích funkcií je Sec, ak pre správu M je ťažké nájsť M' takú, že M a M' majú rovnaký haš, teda $H_K(M) = H_K(M')$.

Odolnosť druhého vzoru je dôležitá pri kontrole integrity dát. Pomocou hašovacej funkcie sa vypočítajú odtlačky dát. Odtlačky sa odložia tak, aby nemohli byť zmenené. Kontrola prebehne vypočítaním odtlačkov daných dát a ich porovnaním s odloženými odtlačkami. Ak by použitá hašovacia funkcia nebola Sec, útočník by vedel vypočítať iné súbory s rovnakými odtlačkami a pôvodné súbory by mohol nahradiť vypočítanými. Podobne aj pri digitálnych podpisoch by vedel útočník k podpísanému odtlačku nájsť inú správu, ktorá má rovnaký odtlačok.

Vlastnosť Sec môžeme maximalizovať cez množinu všetkých správ $\mathcal{M}$, dostaneme *všade odolnosť druhého vzoru - eSec*, čo znamená, že je ťažké ku každej správe M nájsť správu M' takú, aby ich haše boli rovnaké. Maximalizovaním cez množinu všetkých kľúčov $\mathcal{K}$ dostaneme *vždy odolnosť druhého vzoru - aSec*, ktorá zosilňuje vlastnosť Sec v zmysle, že pre každý kľúč je ťažké nájsť správu M' k náhodne vybranej správe M, pričom ich haše sa rovnajú. Teda nemôže existovať žiaden „slabý“ kľúč K_0 taký, že pre funkciu H_{K_0} by sa správa M' hľadala ľahko.

Definícia 1.2.11 (eSec a aSec)**.** Nech $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií a $\{0,1\}^m \subseteq \mathcal{M}$. Definujme výhody:

$$\mathbf{Adv}_H^{\mathrm{eSec}[m]}(\mathcal{A}) = \max_{M \in \{0,1\}^m} \left\{ \Pr\left[K \xleftarrow{\$} \mathcal{K}; M' \leftarrow \mathcal{A}(K) : (M \neq M') \wedge (H_K(M) = H_K(M'))\right]\right\}$$

$$\mathbf{Adv}_H^{\mathrm{eSec}}(\mathcal{A}) = \Pr\left[(M,S) \leftarrow \mathcal{A}; K \xleftarrow{\$} \mathcal{K}; M' \leftarrow \mathcal{A}(K,S) : (M \neq M') \wedge (H_K(M) = H_K(M'))\right]$$

Rodina hašovacích funkcií je eSec, ak pre každého efektívneho útočníka $\mathcal{A}$ a $m \in \mathbb{N}$ sú obe výhody zanedbateľné.
Hovoríme, že rodina hašovacích funkcií H je (t,ε)-eSec, ak každý útočník $\mathcal{A}$ bežiaci v čase najviac t má výhodu $\mathbf{Adv}_H^{\mathrm{eSec}[m]}(\mathcal{A}) \leq \varepsilon$ pre všetky m také, že $\{0,1\}^m \subseteq \mathcal{M}$.

$$\mathbf{Adv}_H^{\mathrm{aSec[m]}}(\mathcal{A}) = \max_{K \in \mathcal{K}} \left\{ \Pr\left[M \xleftarrow{\$} \{0,1\}^m; M' \leftarrow \mathcal{A}(M) : (M \neq M') \wedge (H_K(M) = H_K(M'))\right]\right\}$$

$$\mathbf{Adv}_H^{\mathrm{aSec[m]}}(\mathcal{A}) = \Pr\left[(K,S) \leftarrow \mathcal{A}; M \xleftarrow{\$} \{0,1\}^m; M' \leftarrow \mathcal{A}(M,S) : (M \neq M') \wedge (H_K(M) = H_K(M'))\right]$$

Rodina hašovacích funkcií je aSec, ak pre každého efektívneho útočníka $\mathcal{A}$ a $m \in \mathbb{N}$, sú obe výhody zanedbateľné.

Hovoríme, že rodina hašovacích funkcií H je (t,ε)-aSec, ak každý útočník $\mathcal{A}$ bežiaci v čase najviac t má výhodu $\mathbf{Adv}_H^{\mathrm{aSec}}(\mathcal{A}) \leq \varepsilon$ pre všetky m také, že $\{0,1\}^m \subseteq \mathcal{M}$.

Ďalšou dôležitou vlastnosťou je *odolnosť prvého vzoru s voliteľným cieľom a vynúteným prefixom (Chosen target forced prefix preimage resistance - Ctfp).* Túto vlastnosť by mala spĺňať rodina hašovacích funkcií, ak má byť odolná voči tzv. *Nostradamovmu útoku* , ktorý uviedol Kelsey a Kohno [9]. Predstavme si nasledujúcu situáciu: V jeden deň na začiatku

roka 2006 sa objaví v novinách reklama:

Ja, Nostradamus, týmto zverejňujem MD5 haš Y veľmi dôležitej predpovede, ktorá obsahuje veľmi blízke ceny všetkých akcií v S&P500 z posledného obchodného dňa roku 2006.

Pár týždňov po poslednom obchodnom dni v roku 2006 Nostradamus uverejní správu M obsahujúcu v prvom bloku ceny S&P500 akcií. Správa potom pokračuje s veľmi nepravdivými predpoveďami. Otázka je, či Nostradamus mohol klamať o jeho schopnostiach predpovedať. Ak nie je funkcia Ctfp bezpečná, s veľkou pravdepodobnosťou Nostradamus klamal.

Definícia 1.2.12 (Ctfp)**.** Nech $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií a $\{0,1\}^m \subseteq \mathcal{M}$. Rodina hašovacích funkcií je Ctfp, ak pre každého efektívneho útočníka $\mathcal{A}$ a $m \in \mathbb{N}$ je nasledujúca výhoda zanedbateľná:

$$\mathbf{Adv}_H^{\mathrm{Ctfp[m]}}(\mathcal{A}) = Pr\big[K \xleftarrow{\$} \mathcal{K}; (Y,S) \leftarrow \mathcal{A}(K); P \xleftarrow{\$} \{0,1\}^m; \\ M \leftarrow \mathcal{A}(P,S) : H_K(P \parallel M) = Y\big]$$

Hovoríme, že rodina hašovacích funkcií H je (t,ε)-Ctfp, ak každý útočník $\mathcal{A}$ bežiaci v čase najviac t má výhodu $\mathbf{Adv}_H^{\mathrm{Ctfp}}(\mathcal{A}) \leq \varepsilon$ pre všetky m také, že $\{0,1\}^m \subseteq \mathcal{M}$.

Definícia 1.2.13 (aCtfp)**.** Nech $H : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$ je rodina hašovacích funkcií a $\{0,1\}^m \subseteq \mathcal{M}$. Definujeme nasledujúce výhody:

$$\mathbf{Adv}_H^{\mathrm{aCtfp}[m]}(\mathcal{A}) = \max_{K \in \mathcal{K}} \big\{ \Pr\big[(Y,S) \leftarrow \mathcal{A}; P \xleftarrow{\$} \{0,1\}^m; \\ M \leftarrow \mathcal{A}(P,S) : H_K(P \parallel M) = Y\big]\big\}$$

$$\mathbf{Adv}_H^{\mathrm{aCtfp}[m]}(\mathcal{A}) = \Pr\big[(Y,K,S) \leftarrow \mathcal{A}; P \xleftarrow{\$} \{0,1\}^m; M \leftarrow \mathcal{A}(P,S) : H_K(P \parallel M) = Y\big]$$

Rodina hašovacích funkcií je aCtfp, ak pre každého efektívneho útočníka $\mathcal{A}$ a $m \in \mathbb{N}$ sú definované výhody zanedbateľné.
Hovoríme, že rodina hašovacích funkcií H je (t,ε)-aCtfp, ak pre každého útočníka $\mathcal{A}$ bežiaceho v čase najviac t je výhoda $\mathbf{Adv}_H^{\mathrm{aCtfp}[m]}(\mathcal{A}) \leq \varepsilon$ pre všetky m také, že $\{0,1\}^m \subseteq \mathcal{M}$.

1.3 Ekvivalencia definícií aCtfp

V nasledujúcej časti dokážeme ekvivalenciu výhod definovaných pre aCtfp. Prvú z nich označíme aCtfp_1 a druhú aCtfp_2, teda:

$$\mathbf{Adv}_H^{\text{aCtfp}_1[m]}(\mathcal{A}) = \max_{K\in\mathcal{K}}\Big\{\Pr\big[(Y,S)\leftarrow\mathcal{A}; P\xleftarrow{\$}\{0,1\}^m; \\ M\leftarrow\mathcal{A}(P,S) : H_K(P\parallel M)=Y\big]\Big\}$$

$$\mathbf{Adv}_H^{\text{aCtfp}_2[m]}(\mathcal{A}) = \Pr\big[(Y,K,S)\leftarrow\mathcal{A}; P\xleftarrow{\$}\{0,1\}^m; M\leftarrow\mathcal{A}(P,S) : H_K(P\parallel M)=Y\big]$$

Veta 1.3.1. *Definície výhod $aCtfp_1$ a $aCtfp_2$ sú ekvivalentné.*

Dôkaz. Predpokladajme, že máme útočníka $\mathcal{A}_1$ útočiaceho na H v zmysle $aCtfp_1$ a nech K je kľúč, pre ktorý má $\mathcal{A}_1$ maximálnu výhodu. Zostrojíme útočníka $\mathcal{A}_2$ útočiaceho na H v zmysle $aCtfp_2$, ktorý bude využívať $\mathcal{A}_1$ nasledovne:

Útočník $\mathcal{A}_2$
[1. fáza, K je kľúč, pre ktorý má $\mathcal{A}_1$ maximálnu výhodu]
    $(Y,S)\leftarrow\mathcal{A}_1$
    return (Y,K,S)
[2. fáza so vstupom $P\xleftarrow{\$}\{0,1\}^m$]
    $M\leftarrow\mathcal{A}_1(P,S)$
    return M

Zjavne platí, že výhoda útočníka $\mathcal{A}_2$ je aspoň tak veľká, ako výhoda $\mathcal{A}_1$, teda $\mathbf{Adv}_H^{\text{aCtfp}_2}(\mathcal{A}_2) \geq \mathbf{Adv}_H^{\text{aCtfp}_1}(\mathcal{A}_1)$.
Obrátene predpokladajme, že máme útočníka $\mathcal{A}_2$, skonštruujeme útočníka $\mathcal{A}_1$ s využitím $\mathcal{A}_2$ nasledovne:

Útočník $\mathcal{A}_1$
[1. fáza]
    $(Y,K,S)\leftarrow\mathcal{A}_2$
    return (Y,S)
[2. fáza so vstupom $P\xleftarrow{\$}\{0,1\}^m$]
    $M\leftarrow\mathcal{A}_2(P,S)$
    return M

Pre útočníka $\mathcal{A}_1$ platí $\mathbf{Adv}_H^{\mathrm{aCtfp}_1}(\mathcal{A}_1) \geq \mathbf{Adv}_H^{\mathrm{aCtfp}_2}(\mathcal{A}_2)$. Dostávame, že $\mathbf{Adv}_H^{\mathrm{aCtfp}_1}(\mathcal{A}_1) = \mathbf{Adv}_H^{\mathrm{aCtfp}_2}(\mathcal{A}_2)$.

□

1.4 Implikácie

Rogaway a Shrimpton v [18] skúmali vzťahy medzi jednotlivými vlastnosťami rodiny hašovacích funkcií. Definovali *bežnú (conventional)* a *dočasnú (provisional)* implikáciu. Bežná implikácia predstavuje štandardnú definíciu implikácie, keď jej sila nezáleží na tom, ako veľmi hašovacia funkcia komprimuje (pomer mohutnosti oboru hodnôt funkcie k mohutnosti definičného oboru). Sila dočasnej implikácie závisí na stupni kompresie dosiahnutej hašovacou funkciou. Venovali sa vlastnostiam Pre, ePre, Sec, eSec, aSec a Coll. Naor a Reingold v [13] dokázali vzťah medzi Mac a Prf. Rjaško vo svojej práci [15] nadviazal na ich zistenia a rozšíril ich o vlastnosti Mac, Ctfp, aCtfp, Prf a Pro.
Keď vlastnosť xxx implikuje yyy, znamená to, že ak rodina hašovacích funkcií je xxx bezpečná, potom je aj yyy bezpečná. Ak xxx neimplikuje yyy, tak niektorá rodina hašovacích funkcií je xxx bezpečná, ale nie je yyy bezpečná. Analyzovanie vzťahov medzi vlastnosťami rodín hašovacích funkcií nám uľahčuje skúmanie vlastností pre nejakú konkrétnu rodinu hašovacích funkcií. Napríklad majme rodinu hašovacích funkcií H, o ktorej vieme, že je aSec bezpečná. Vlastnosť aSec implikuje Pre, aPre a Sec, preto je H aj Pre, aPre a Sec bezpečná. V tabuľke 1.1 sú znázornené dokázané implikácie.

1.5 Robustné kombinátory zachovávajúce viaceré vlastnosti

Robustné kombinátory sú v kryptológii dôležitou konštrukciou. Kombinujú rôzne kryptografické primitíva, pričom požadujeme, aby výsledný systém bol xxx bezpečný, ak aspoň jedno z použitých primitív je xxx bezpečné. Robustnými kombinátormi sa zvyšuje bezpečnosť, pretože ak chceme rozbiť kombinátor, musíme rozbiť všetky čiastkové primitíva použité v kombinátore. Využitie kombinátorov je v rôznych kryptosystémoch, v ktorých používame rodiny hašovacích funkcií a o týchto rodinách nevieme s istotou povedať, že sú úplne bezpečné. Aplikovaním kombinátora na také funkcie zvýšime bezpečnosť celého kryptosystému, pretože na rozbitie kombinátora musí útočník rozbiť všetky použité funkcie.

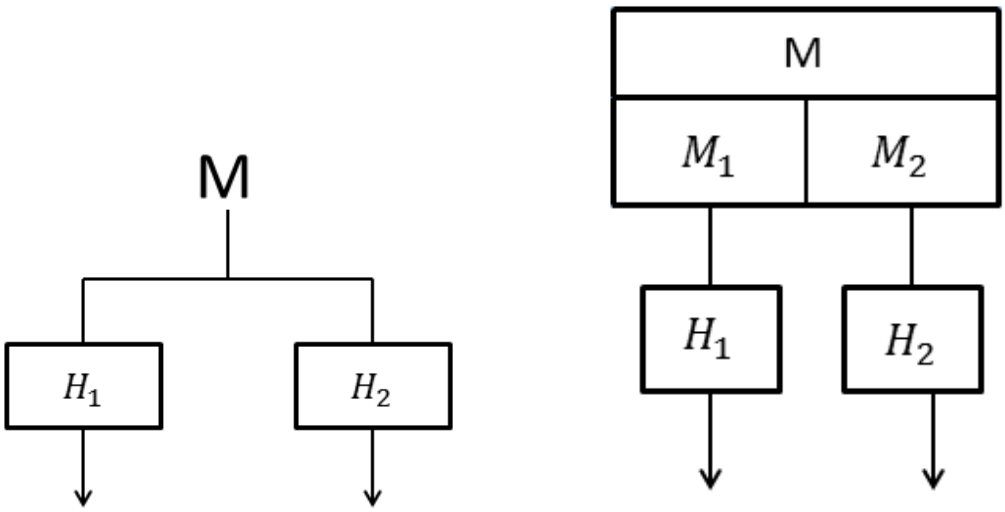
	Pre	aPre	ePre	Sec	aSec	eSec	Coll	Mac	Ctfp	aCtfp	Prf	Pro
Pre	×	↛	↛	↛	↛	↛	↛	↛	↛	↛	↛	↛
aPre	→	×	↛	↛	↛	↛	↛	↛	↛	↛	↛	↛
ePre	→	↛	×	↛	↛	↛	↛	↛	↛	↛	↛	↛
Sec	→	↛	↛	×	↛	↛	↛	↛	↛	↛	↛	↛
aSec	→	→	↛	→	×	↛	↛	↛	↛	↛	↛	↛
eSec	→	↛	↛	→	↛	×	↛	↛	↛	↛	↛	↛
Coll	→	↛	↛	→	↛	→	×	↛	→	↛	↛	↛
Mac	↛	↛	↛	↛	↛	↛	↛	×	↛	↛	↛	↛
Ctfp	↛	↛	↛	↛	↛	↛	↛	↛	×	↛	↛	↛
aCtfp	↛	↛	↛	↛	↛	↛	↛	↛	→	×	↛	↛
Prf	↛	↛	↛	↛	↛	↛	↛	→	↛	↛	×	↛
Pro	→	↛	→	→	↛	→	→	→	→	↛	→	×

Tabuľka 1.1: Vzťahy medzi definíciami vlastností rodín hašovacích funkcií.

Definícia 1.5.1 (Robustný kombinátor)**.** Nech $H_1, H_2, \ldots, H_l$ sú rodiny hašovacích funkcií, $l \in \mathbb{N}$. Kombinátor pre l rodín hašovacích funkcií je algoritmus C, ktorého vstup je množina rodín hašovacích funkcií a výstupom je jedna rodina hašovacích funkcií $C(H_1, H_2, \ldots, H_l)$. Nech $\mathcal{P}$ je množina vlastností rodín hašovacích funkcií a $P_i \in \mathcal{P}$. Hovoríme, že kombinátor $C(H_1, H_2, \ldots, H_l)$ je P_i-robustný práve vtedy, ak aspoň jedna z použitých rodín hašovacích funkcií je P_i bezpečná.

Nech $H_1, H_2, \ldots, H_l$ sú rodiny hašovacích funkcií. Kombinátor, ktorý kombinuje tieto funkcie, budeme označovať: $\mathbf{C^{H_1,H_2,\ldots,H_l}}$.
Medzi najznámejšie kombinátory patria nasledujúce tri konštrukcie: Majme dve rodiny hašovacích funkcií H_1 a H_2. Kombinátor zreťazenia znázornený na obrázku 1.1, je definovaný: $Comb_1^{H_1,H_2}(M) = H_1(M) \parallel H_2(M)$. Tento kombinátor je Coll-robustný vďaka tomu, že ak aspoň jedna z použitých funkcií je Coll bezpečná, nevieme nájsť kolíziu pre kombinátor (výstup Coll bezpečnej funkcie je s veľkou pravdepodobnosťou rôzny, a teda aj výstup kombinátora musí byť rôzny). Avšak tento kombinátor nezachováva pseudonáhodnosť, pretože útočník môže ľahko rozoznať zreťazený výstup od náhodnej hodnoty. Stačí, ak vyskúša tú časť výstupu funkcie, ktorá nie je Prf. Kombinátor $Comb_1$ nie je ani Pre-robustný. Ďalším veľmi známym kombinátorom, je $Comb_2^{H_1,H_2}(M) = H_1(M) \oplus H_2(M)$. Kombinátor $Comb_2$ je Prf robustný, ale nie je odolný voči kolízii, pretože kolízia kombinátora neznamená kolíziu v oboch funkciách. Kombinátor $Comb_3^{H_1,H_2}(M_1 \parallel M_2) = H_1(M_1) \parallel H_2(M_2)$ znázornený na obrázku 1.1 je Pre-robustný, ale nie je Coll a Prf-robustný.

Ak kombinátor zachováva viac vlastností, môžeme žiadať, aby každú z nich mala aspoň



Obr. 1.1: Kombinátor $Comb_1$ (vľavo) a kombinátor $Comb_3$ (vpravo)

jedna funkcia. Napr. ak H_i má vlastnosť MAC, tak ju má aj kombinátor nezávisle od ostatných vlastností. Takýto kombinátor nazývame kombinátor zachovávajúci silno viaceré vlastnosti.
Ak kombinátor zdedí nejakú množinu vlastností, ktoré má aspoň jedna (všetky jej vlastnosti) hašovacia funkcia, hovoríme, že je to kombinátor zachovávajúci slabo viaceré vlastnosti.
Kombinátor stredne zachovávajúci viaceré vlastnosti je taký, ktorý garantuje viaceré vlastnosti, ale rôzne hašovacie funkcie pokrývajú rôzne vlastnosti.

Definícia 1.5.2. Pre množinu $PROP = \{P_1, P_2, \ldots, P_N\}$ vlastností je kombinátor C pre rodiny hašovacích funkcií H_0, H_1:

- *slabo zachovávajúci viaceré vlastnosti (wMPP)* pre $PROP$, ak

$$PROP = PROP(H_0) \vee PROP = PROP(H_1) \Rightarrow PROP = PROP(C)$$

- *stredne zachovávajúci viaceré vlastnosti (mMPP)* pre $PROP$, ak

$$PROP = PROP(H_0) \cup PROP(H_1) \Rightarrow PROP = PROP(C)$$

- *silno zachovávajúci viaceré vlastnosti (sMPP)* pre $PROP$, ak

$$P_i \in PROP(H_0) \cup PROP(H_1) \Rightarrow P_i \in PROP(C)$$

V prípade kombinátora, ktorý slabo a stredne zachováva množinu vlastností $PROP$, môže vlastnosť P_i závisieť na vlastnosti P_j. Potom pre množinu $PROP' \subseteq PROP$ takú, že $PROP' = PROP - P_j$ daný kombinátor nezachováva množinu vlastností $PROP'$. V prípade silno zachovávajúceho kombinátora to nemôže nastať (z definície).

1.6 Predchádzajúce výsledky

Herzberg v [7] zaviedol teóriu robustných kombinátorov pre rôzne kryptografické primitíva. Boneh a Boyen [2] dokázali, že neexistuje konštrukcia zachovávajúca odolnosť voči kolízii pre kombinovanie hašovacích funkcií odolných voči kolízii tak, aby bol výstup kratší, než zreťazenie týchto funkcií. Predpokladali, že kombinátor sa pýta každého komponentu práve raz. Pietrzak [14] ukázal, že dĺžka výstupu kombinátora l hašovacích funkcií s výstupom dĺžky v musí byť aspoň $(v - O(log_2(q)))l$, kde q je počet orákulovských volaní.

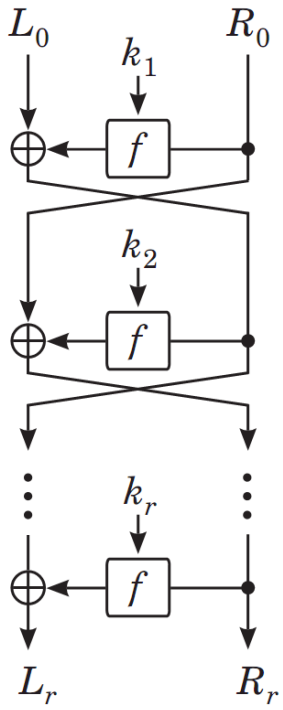
Kapitola 2

Príklady kombinátorov zachovávajúcich viaceré vlastnosti

V nasledujúcej kapitole analyzujeme robustné kombinátory, ktoré zachovávajú viaceré vlastnosti. Autormi týchto kombinátorov sú Fischlin, Lehmann a Pietrzak. K daným kombinátorom dokážeme, že zachovávajú aj niektoré iné vlastnosti, ktorých zachovávanie ešte nebolo dokázané.
Nech F_n označuje množinu všetkých funkcií $\{0,1\}^n \to \{0,1\}^n$ a $P_n \subset F_n$ množinu všetkých permutácií na množine $\{0,1\}^n$.

Definícia 2.0.1 (Feistelovská permutácia)**.** Nech $f_1, f_2, \ldots, f_r \in F_n$, Feistelovská permutácia $\text{Feistel}_{f_1,f_2,\ldots,f_r}$ vznikne postupným aplikovaním r kôl. Vstup do i-tého kola je rozdelený na dve časti L_{i-1} a R_{i-1}. Výstup i-tého kola je definovaný ako $L_i = R_{i-1}$ a $R_i = L_{i-1} \oplus f_i(R_{i-1})$, kde f_i je funkcia použitá v i-tom kole.

Feistelovská permutácia znázornená na obrázku 2.1 je ľahko invertovateľná, ak poznáme funkcie $f_1, f_2, \ldots, f_r$. Nech (L_r, R_r) je výstup permutácie, potom $\text{Feistel}^{-1}_{f_1,f_2,\ldots,f_r} = \text{Feistel}_{f_r,f_{r-1},\ldots,f_1}(L_r, R_r)$. Feistelovská permutácia umožňuje vytvorenie invertovateľných funkcií aj za použitia neinvertovateľných funkcií. Luby a Rackoff v [11] dokázali, že ak $f_1, f_2, f_3 \overset{\$}{\leftarrow} F_n$ sú tri nezávislé náhodné funkcie, potom $\text{Feistel}_{f_1,f_2,f_3} : \{0,1\}^{2n} \to \{0,1\}^{2n}$ je neodlíšiteľná od náhodnej permutácie. Ak $F : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ je pseudonáhodná funkcia, potom aj $\text{Feistel}_{F_{K_1},F_{K_2},F_{K_3}}$ je pseudonáhodná permutácia.
Ak $F : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ je pseudonáhodná funkcia, potom aj funkcia $F'(K, M) = \text{Prefix}_l(F(K, M))$, kde $l \leq n$ je tiež pseudonáhodná. Aplikovaním takejto funkcie F' na pseudonáhodné kolové funkcie použité vo Feistelovskej permutácii dostávame opäť pseudonáhodnú permutáciu.



Obr. 2.1: Feistelovská permutácia [20]

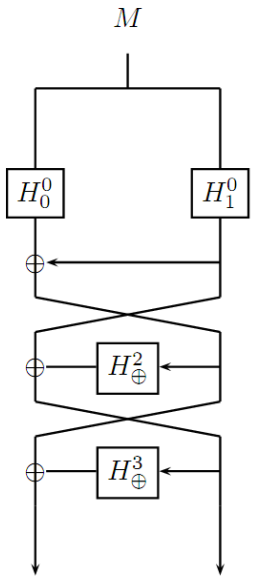
2.1 C_{4P} kombinátor

Kombinátor C_{4P} znázornený na obrázku 2.2, správu dĺžky n bitov zahašuje na odtlačok dĺžky $2n$ bitov. Je založený na zreťazovacom kombinátore a trojkolovej Feistelovskej permutácii Feistel^3 na množine $\{0,1\}^{2n}$. Kolové funkcie sú definované $H^i_{\oplus}(\cdot) = H^i_0(\cdot) \oplus H^i_1(\cdot)$ pre $i = 2, 3$, kde $H^i_b(\cdot)$ označuje funkciu $H_b(\langle i \rangle_2 \| \cdot)$. Prvý krok je identická funkcia $H^1_{\oplus}(X) = X$. V i-tom kole je vstup (L_i, R_i) mapovaný na výstup $(R_i, L_i \oplus H^i_{\oplus}(R_i))$.
Kombinátor C_{4P} pre rodiny hašovacích funkcií H_0, H_1 a vstupnú správu M je definovaný:

$$C_{4P}^{H_0,H_1}(M) = \text{Feistel}^3(H^0_0(M) \| H^0_1(M))$$

Veta 2.1.1. *C_{4P} je kombinátor silno zachovávajúci vlastnosti Coll, Prf, eSec, MAC, aSec, Sec, aCtfp, Ctfp a ePre.*

Dôkaz zachovávania vlastností Coll, Prf, eSec a MAC bol dokázaný v [6]. My dokážeme, že zachováva vlastnosti aSec, Sec, aCtfp, Ctfp a ePre.



Obr. 2.2: Kombinátor C_{4P} [6]

Veta 2.1.2. *Kominátor C_{4P} je Sec-robustný.*

Dôkaz. Nech aspoň jedna z rodín hašovacích funkcií použitá v kombinátore je Sec bezpečná, nech je to H_b, $b \in \{0,1\}$. Predpokladajme, že máme útočníka $\mathcal{A}$, ktorý s pravdepodobnosťou $\varepsilon(n)$ pre náhodne zvolenú správu M a náhodne zvolený kľúč K, nájde $M' \neq M$ takú, že $C_{4P}(M) = C_{4P}(M')$. Tohto útočníka použijeme na zostrojenie útočníka $\mathcal{B}_b$ útočiaceho na rodinu hašovacích funkcií H_b.

Útočník $\mathcal{B}_\mathbf{b}$
[1. fáza so vstupom $K_b \xleftarrow{\$} \mathcal{K}$ a $M \xleftarrow{\$} \mathcal{M}$]
$K_{\bar{b}} \xleftarrow{\$} \mathcal{K}$
$M' \leftarrow A(M, K_b, K_{\bar{b}})$
return $00 \parallel M'$

Predpokladajme, že útočník $\mathcal{A}$ dá na výstup $M' \neq M$, pričom $C_{4P}(M) = C_{4P}(M')$, potom útočník $\mathcal{B}_b$ našiel k H_b správu $00 \parallel M'$ takú, že $H_b(00 \parallel M) = H_b(00 \parallel M')$. Cez permutáciu Feistel[3] nemohla nastať kolízia, teda musela nastať jedine pred ňou, pred vstupom do Feistel$^3(\cdot)$. Výhoda útočníka $\mathcal{B}_b$, že nájde správu $00 \parallel M'$, pre správu $00 \parallel M$ je aspoň taká, ako výhoda útočníka $\mathcal{A}$. Lenže pre funkciu H_b sa správa $00 \parallel M$ nevolila náhodne, ale iba jej časť M. Pravdepodobnosť, že sa náhodne vyberie reťazec 00, je 1/4. Celkovo

dostávame, že výhoda útočníka $\mathcal{B}_b$ je $\mathbf{Adv}^{\mathrm{Sec}}_{H_b}(\mathcal{B}_b) \geq \frac{1}{4}\varepsilon(n)$. Keďže H_b je Sec bezpečná, $\mathbf{Adv}^{\mathrm{Sec}}_{H_b}(\mathcal{B}_b) \leq \mathrm{negl(n)}$, preto je hodnota $\varepsilon(n) \leq \mathrm{negl(n)}$ zanedbateľná. □

Veta 2.1.3. *Kominátor C_{4P} je aSec-robustný.*

Dôkaz. Predpokladajme, že aspoň jedna z rodín hašovacích funkcií je aSec bezpečná, označme ju H_b, $b \in \{0,1\}$. Ďalej predpokladajme, že máme útočníka $\mathcal{A}$ útočiaceho na C_{4P} v zmysle aSec a jeho výhoda je $\varepsilon(n)$. Útočník $\mathcal{A}$ dá najprv na výstup trojicu $(K_b, K_{\bar{b}}, S)$, potom na vstup dostane náhodne vybranú správu M a stav S. Útočník $\mathcal{A}$ vypočíta správu M' takú, že $C_{4P}(M) = C_{4P}(M')$, ale $M' \neq M$. Tohto útočníka použijeme na zostrojenie útočníka $\mathcal{B}_b$ na funkciu H_b v zmysle aSec, o ktorej predpokladáme, že je aSec bezpečná. Konštrukcia útočníka:

Útočník $\mathcal{B}^{\mathrm{aSec}}_{\mathbf{b}}$
[1. fáza]
$(K_b, K_{\bar{b}}, S) \leftarrow \mathcal{A}$
return (K_b, S)
[2. fáza so vstupom $M \xleftarrow{\$} \mathcal{M}$]
$M' \leftarrow \mathcal{A}(M, S)$
return $00 \parallel M'$

Predpokladajme, že útočník $\mathcal{A}$ dá na výstup správu $M' \neq M$, pričom $C_{4P}(M) = C_{4P}(M')$, potom útočník $\mathcal{B}_b$ našiel správu $00 \parallel M'$ takú, že $M \neq M'$, ale $H_b(00 \parallel M) = H_b(00 \parallel M')$. Opäť, kolízia mohla nastať jedine pred permutáciou Feistel[3]. Výhoda útočníka $\mathcal{B}_b$, že nájde správu $00 \parallel M'$, pre správu $00 \parallel M$ je aspoň taká, ako výhoda útočníka $\mathcal{A}$. Lenže pre funkciu H_b sa správa $00 \parallel M$ nevolila náhodne, ale iba jej časť M. Pravdepodobnosť, že sa náhodne vyberie reťazec 00, je $1/4$. Celkovo dostávame, že výhoda útočníka $\mathcal{B}_b$ je $\mathbf{Adv}^{\mathrm{aSec}}_{H_b}(\mathcal{B}_b) \geq \frac{1}{4}\varepsilon(n)$. Rodina hašovacích funkcií H_b je aSec bezpečná, preto je hodnota $\varepsilon(n)$ zanedbateľná. □

Veta 2.1.4. *Kominátor C_{4P} je Ctfp-robustný.*

Dôkaz. Predpokladajme, že aspoň jedna z rodín hašovacích funkcií použitých v kombinátore je Ctfp bezpečná, označme ju H_b, $b \in \{0,1\}$. Predpokladajme, že existuje úspešný útočník $\mathcal{A}$, ktorý útočí na C_{4P} v zmysle Ctfp s výhodou $\varepsilon(n)$. Útočník $\mathcal{A}$ dostane na vstup náhodne vygenerované kľúče (špecifikujú rodiny hašovacích funkcií) a vygeneruje dvojicu (Y, S). Potom dostane útočník $\mathcal{A}$ na vstup náhodne zvolený reťazec P a stav S. $\mathcal{A}$ vypočíta

reťazec M taký, že $C_{4P}(P \parallel M) = Y$ s pravdepodobnosťou $\varepsilon(n)$. Tohto útočníka využijeme na zostrojenie útočníka $\mathcal{B}_b$ útočiaceho na rodinu hašovacích funkcií H_b v zmysle Ctfp.

Útočník $\mathcal{B}_{\mathbf{b}}^{\text{Ctfp}}$
[1. fáza so vstupom $K_b \xleftarrow{\$} \mathcal{K}$]
$K_{\bar{b}} \xleftarrow{\$} \mathcal{K}$
$(Y, S) \leftarrow \mathcal{A}(K_b, K_{\bar{b}})$
$Y_0 \parallel Y_1 = \text{Feistel}^{3^{-1}}(Y)$
return (Y_b, S)
[2. fáza so vstupom $P \xleftarrow{\$} \{0,1\}^m$]
$M \leftarrow \mathcal{A}(P, S)$
return M

Výstup útočníka $\mathcal{A}$, reťazec M, pre ktorý s vysokou pravdepodobnosťou platí $C_{4P}(P \parallel M) = Y$, dá útočník $\mathcal{B}_b$ tiež na výstup. S aspoň takou istou pravdepodobnosťou, ako pre útočníka $\mathcal{A}$, platí : $H_b(00 \parallel P \parallel M) = Y_b$. Avšak časť reťazca $00 \parallel P$ nebola volená náhodne, pravdepodobnosť náhodného výberu 00 je $1/4$. Dostávame, že pre výhodu útočníka $\mathcal{B}_b$ platí $\mathbf{Adv}_{H_b}^{Ctfp}(\mathcal{B}_b) \geq \frac{1}{4}\varepsilon(n)$. Keďže H_b je Ctfp bezpečná, hodnota $\varepsilon(n)$ je zanedbateľná.

□

Veta 2.1.5. *Kominátor C_{4P} je aCtfp-robustný.*

Dôkaz. Predpokladajme, že aspoň jedna z rodín hašovacích funkcií použitých v kombinátore je aCtfp bezpečná, označme ju H_b, $b \in \{0,1\}$. Ďalej predpokladajme, že existuje úspešný útočník $\mathcal{A}$, ktorý útočí na C_{4P} v zmysle aCtfp. Teda výhoda útočníka $\mathcal{A}$

$$\varepsilon(n) = \Pr\left[(Y, K, S) \leftarrow \mathcal{A}; P \xleftarrow{\$} \{0,1\}^m; M \leftarrow \mathcal{A}(P, S) : H_K(P \parallel M) = Y\right].$$

Útočníka $\mathcal{A}$ využijeme na zostrojenie útočníka $\mathcal{B}_b$ útočiaceho na rodinu hašovacích funkcií H_b.

Útočník $\mathcal{B}_{\mathbf{b}}^{\text{aCtfp}}$
[1. fáza]
$(Y, K_b, K_{\bar{b}}, S) \leftarrow \mathcal{A}$
$Y_0 \parallel Y_1 = \text{Feistel}^{3^{-1}}(Y)$
return (Y_b, K_b, S)
[2. fáza so vstupom $P \xleftarrow{\$} \{0,1\}^m$]
$M \leftarrow \mathcal{A}(P, S)$
return M

Pre výstup útočníka $\mathcal{A}$, reťazec M, s pravdepodobnosťou $\varepsilon(n)$ platí $C_{4P}(P \parallel M) = Y$. Útočník $\mathcal{B}_b$ s aspoň takou istou pravdepodobnosťou, ako pre útočníka $\mathcal{A}$, dá na výstup reťazec M, pre ktorý platí $H_b(00 \parallel P \parallel M) = Y_b$. Časť reťazca $00 \parallel P$ nebola volená náhodne, pravdepodobnosť náhodného výberu 00 je 1/4. Preto je výhoda útočníka $\mathcal{B}_b$ $\mathbf{Adv}_{H_b}^{\text{aCtfp}}(\mathcal{B}_b) \geq \frac{1}{4}\varepsilon(n)$. Rodina hašovacích funkcií H_b je aCtfp bezpečná, preto je $\varepsilon(n)$ zanedbateľná. □

Veta 2.1.6. *Kominátor C_{4P} je ePre-robustný.*

Dôkaz. Dokážeme, že kombinátor zreťazenia $Comb_1(M) = H_0(M) \parallel H_1(M)$ je ePre-robustný. Predpokladajme, že aspoň jedna z rodín hašovacích funkcií v $Comb_1$ je ePre bezpečná, označme ju H_b, $b \in \{0,1\}$. Ďalej predpokladajme, že máme útočníka $\mathcal{A}$ útočiaceho na $Comb_1$ v zmysle ePre a jeho výhoda je $\varepsilon(n)$. Tohto útočníka použijeme na zostrojenie útočníka $\mathcal{B}_b$ na funkciu H_b v zmysle ePre, o ktorej predpokladáme, že je ePre bezpečná. Konštrukcia útočníka:

Útočník $\mathcal{B}_{\mathbf{b}}^{\text{ePre}}$
[1. fáza]
$(Y, S) \leftarrow \mathcal{A}$
$Y = Y_0 \parallel Y_1$
return (Y_b, S)
[2. fáza so vstupom $K_b \xleftarrow{\$} \{0,1\}^k$]
$K_{\bar{b}} \xleftarrow{\$} \{0,1\}^k$
$M \leftarrow \mathcal{A}(K_b, K_{\bar{b}}, S)$
return M

Predpokladajme, že útočník $\mathcal{A}$ dá na výstup správu M, pričom $Comb_1^{K_0,K_1}(M) = Y$, útočník $\mathcal{B}_b$ našiel správu M takú, že $H_b(M) = Y_b$. Zjavne platí, že výhoda útočníka $\mathcal{B}_b$ je

aspoň taká, ako výhoda útočníka $\mathcal{A}$, teda $\mathbf{Adv}^{\mathrm{ePre}}_{H_b}(\mathcal{B}_b) \geq \varepsilon(n)$. Hodnota $\varepsilon(n)$ je zanedbateľná, lebo rodina hašovacích funkcií H_b je ePre bezpečná.
Aplikovaním permutácie sa neporuší vlastnosť ePre, preto je kombinátor C_{4P} ePre-robustný.
□

Robustnosť kombinátora C_{4P} pre vlastnosť aPre sa nám nepodarila dokázať, a preto ostáva otvoreným problémom.

Kombinátor C_{4P} nezachováva vlastnosť Pre, pretože ju nezachováva kombinátor zreťazenia. Fischlin, Lehmann a Pietrzak [6] upravili kombinátor zreťazenia na kombinátor $C_{CR\&OW}$, aby zachovával jednosmernosť. Na vstup jednej z použitých funkcií aplikovali tzv. párovo nezávislú permutáciu π (viď. C_{6P} kombinátor):

$$C^{H_0,H_1,\pi}_{CR\&OW}(M) = H_0(\pi(M)) \parallel H_1(M)$$

O kombinátore $C_{CR\&OW}$ dokázali, že je Coll, eSec, Mac a Pre robustný. Dôkaz je uvedený v [6]. Ďalej kombinátor C_{4P} rozšírili na kombinátor $C_{4P\&IRO}$ znázornený na obrázku 2.3 a definovaný ako:

$$C^{H_0,H_1,g}_{CR\&IRO}(M) = \mathrm{Feistel}^2_\alpha(H^0_0(M) \parallel H^0_1(M)) \parallel \mathrm{lsb}_{3m}(H^3_\oplus(\alpha_M)) \oplus g(\alpha_M),$$

kde $\mathrm{Feistel}^2_\alpha(\cdot) = \psi[H^1_\oplus(\alpha_M \parallel \cdot), H^2_\oplus(\alpha_M \parallel \cdot)]$, H_0, H_1 sú rodiny hašovacích funkcií,
$g : \{0,1\}^m \to \{0,1\}^{3m}$ pre $m \leq n/3$ je PIF a $lsb_a(x)$ označuje a najmenej významných bitov x. Úpravou Feistelovskej permutácie dosiahli Pro robustnosť. Nahradením kombinátora zreťazenia za $C_{CR\&OW}$ v kombinátore $C_{4P\&IRO}$ dostali kombinátor C_{6P}.

2.2 C_{6P} kombinátor

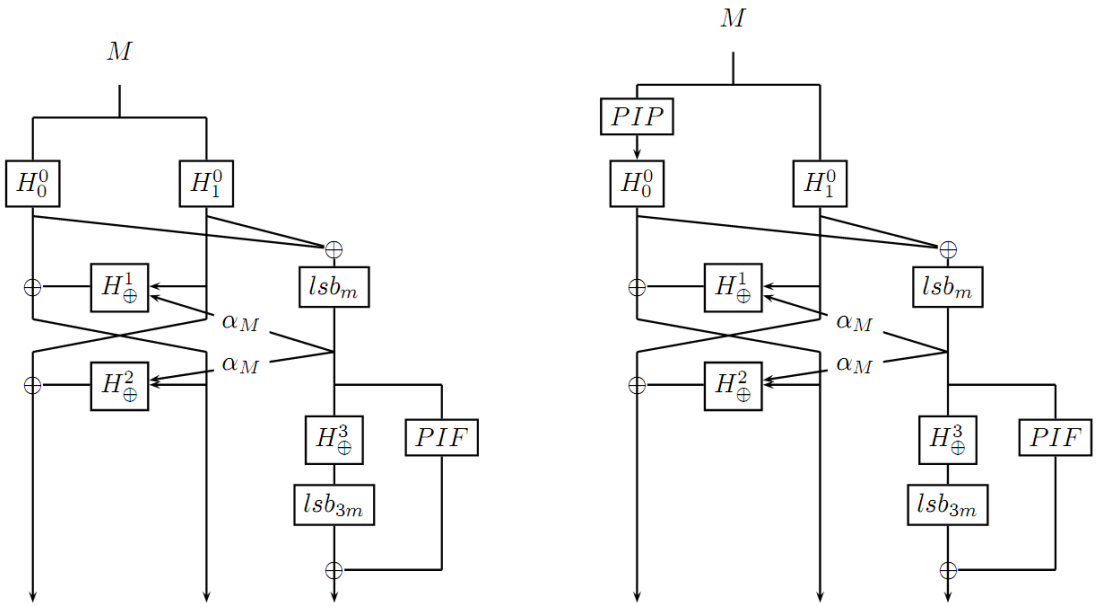
V tejto časti sa venujeme robustnému kombinátoru C_{6P}, ktorý vo svojej konštrukcii využíva párovo nezávislú permutáciu a párovo nezávislú funkciu.

Definícia 2.2.1 (párovo nezávislá funkcia - PIF a permutácia - PIP)**.** Rodina funkcií $G : A \to B$ sa nazýva párovo nezávislá, ak pre všetky $x \neq x' \in A$ a $z \neq z' \in B$ je $\Pr_{g\in G}[g(x) = z \wedge g(x') = z'] = |B|^{-2}$.
Rodina funkcií $\Pi : A \to A$ je párovo nezávislá permutácia, ak pre všetky $x \neq x'$ a $z \neq z' \in A$ platí $\Pr_{g\in\Pi}[g(x) = z \wedge g(x') = z'] = \frac{1}{|B|(|B|-1)}$.

Napríklad funkcia $g : \{0,1\}^n \to \{0,1\}^n$ definovaná pre náhodne vybrané čísla $a, b \in \{0,1\}^n$ ako $g_{(a,b)}(x) = (ax + b)$, pričom sčítanie a násobenie je v poli $GF(2^n)$ je PIF. Ak je a vyberané náhodne z $\{0,1\}^n - 0^n$, potom je g aj PIP.

Nech $g : \{0,1\}^m \to \{0,1\}^{3m}$ pre $m \leq n/3$ je PIF a $\pi : \{0,1\}^n \to \{0,1\}^n$ je PIP. Kombinátor C_{6P}, znázornený na obrázku 2.3, správu dĺžky n bitov zahašuje na odtlačok dĺžky $2n+3m$ bitov. Zo vstupu kombinátora M sa najprv vypočíta $C_{CR\&OW}(M) = H_0^0(\pi(M)) \parallel H_1^0(M)$ a hodnota α_M, ktorá sa nazýva „podpis M“ ako $\alpha_M = \text{lsb}_m(H_\oplus^0(M))$, kde $H_\oplus^0(M) = H_0^0(\pi(M)) \oplus H_1^0(M)$ a $lsb_a(x)$ označuje a najmenej významných bitov x. Hodnota α_M sa používa ako extra prefix v kolových funkciách dvojkolovej Feistelovskej permutácie $\text{Feistel}^2_{\alpha_M}(\cdot) = \Psi[H^1_\oplus(\alpha_M \| \cdot), H^2_\oplus(\alpha_M \| \cdot)]$. Aplikovaním Feistel^2_α na $H_0^0(\pi(M)) \parallel H_1^0(M)$ dostávame prvú časť výstupu kombinátora. Druhá časť kombinátora sa počíta ako $\text{lsb}_{3m}(H^3_\oplus(\alpha_M)) \oplus g(\alpha_M)$. Kombinátor počíta pre vstupnú správu M a jej korešpondujúci podpis $\alpha_M = lsb_M(H_\oplus^0(M))$ nasledujúci výstup:

$$C_{6P}^{H_0,H_1,g,\pi}(M) = \text{Feistel}^2_{\alpha_M}(H_0^0(\pi(M)) \parallel H_1^0(M)) \parallel lsb_{3m}(H^3_\oplus(\alpha_M)) \oplus g_\alpha(M)$$



Obr. 2.3: Kombinátor $C_{CR\&IRO}$ (vľavo) a kombinátor C_{6P} (vpravo) [6]

Veta 2.2.1. *C_{6P} je Coll, Prf, eSec, MAC, PRO a Pre-robustný.*

Dôkaz tejto vety je uvedený v [6]. My dokážeme, že C_{6P} je Sec, aSec, ePre, Ctfp a aCtfp-robustný.

Veta 2.2.2. *Kominátor C_{6P} je Sec-robustný.*

Dôkaz. Nech aspoň jedna z rodín hašovacích funkcií použitá v kombinátore je Sec bezpečná, nech je to H_b, $b \in \{0,1\}$. Predpokladajme, že máme útočníka $\mathcal{A}$, ktorý s pravdepodobnosťou, označme ju $\varepsilon(n)$, pre náhodne zvolenú správu M a náhodne zvolený kľúč K nájde $M' \neq M$ také, že $C_{6P}(M) = C_{6P}(M')$. Tohto útočníka použijeme na zostrojenie útočníka $\mathcal{B}_b$ útočiaceho na rodinu hašovacích funkcií H_b.

Útočník $\mathcal{B}_\mathbf{b}$
[1. fáza so vstupom $K_b \xleftarrow{\$} \mathcal{K}$ a $M \xleftarrow{\$} \mathcal{M}$]
$K_{\bar{b}} \xleftarrow{\$} \mathcal{K}$
$M' \leftarrow \mathcal{A}(M, K_b, K_{\bar{b}})$
ak $b = 0$ potom **return** $00 \parallel \pi(M')$
inak **return** $00 \parallel M'$

Predpokladajme, že $\mathcal{A}$ dá na výstup $M' \neq M$, pričom $C_{6P}(M) = C_{6P}(M') = Y \parallel Y'$, kde $Y \in \{0,1\}^{2n}$ a $Y' \in \{0,1\}^{3m}$. Mohli nastať dva prípady. Ak $\alpha_M = \alpha_{M'}$, potom $\pi(M), \pi(M')$ je kolízia pre H_0^0 a M, M' je kolízia pre H_1^0, pretože

$$H_0^0(\pi(M)) \parallel H_1^0(M) = \text{Feistel}^2_{\alpha_M}{}^{-1}(Y) = \text{Feistel}^2_{\alpha_{M'}}{}^{-1}(Y) = H_0^0(\pi(M')) \parallel H_1^0(M'),$$

kde Feistelovské permutácie $\text{Feistel}^2_{\alpha_M}, \text{Feistel}^2_{\alpha_{M'}}$ sú identické, ak $\alpha_M = \alpha_{M'}$.
Ak $\alpha_M \neq \alpha_{M'}$, kolízia $C_{6P}^{H_0,H_1,\pi,g}(M) = C_{6P}^{H_0,H_1,\pi,g}(M')$ neimplikuje kolíziu pre funkcie H_0^0 a H_1^0. Ukážeme však, že s pravdepodobnosťou 2^{-3m} cez výber funkcie g pre kolíziu M, M' kombinátora C_{6P} nastane $\alpha_M \neq \alpha_{M'}$. Stačí dokázať, že pre $\alpha_M \neq \alpha_{M'}$ s veľkou pravdepodobnosťou platí

$$\text{lsb}_{3m}(H^3_\oplus(\alpha_M)) \oplus g(\alpha_M) \neq \text{lsb}_{3m}(H^3_\oplus(\alpha_{M'})) \oplus g(\alpha_{M'}). \tag{2.1}$$

Hodnoty $\text{lsb}_{3m}(H^3_\oplus(\alpha_X))$ sú fixné a nezávislé od $g(\alpha_X)$ pre $X \in \{M, M'\}$, preto môžeme nerovnosť (2.1) zredukovať na nerovnosť $g(\alpha_M) \neq g(\alpha_{M'})$ (je veľmi malá šanca, že pre fixné hodnoty $\text{lsb}_{3m}(H^3_\oplus(\alpha_X))$ a náhodné $g(\alpha_X)$ pre $X \in \{M, M'\}$ nastane kolízia). Ak g je

PIF, pre každé $\alpha_M \neq \alpha_{M'}$ predchádzajúca nerovnosť nastane s pravdepodobnosťou aspoň $1 - 2^{-3m}$:

$$\begin{aligned}
\Pr_{\alpha_M \neq \alpha_{M'}}[g(\alpha_M) \neq g(\alpha_{M'})] &= \Pr_{g:\{0,1\}^m \to \{0,1\}^{3m}}[g(\alpha_M) = z \wedge g(\alpha_{M'}) = z' \wedge z \ \neq z'] \\
&= \sum_{z \neq z'} \Pr[g(\alpha_M) = z \wedge g(\alpha_{M'} = z')] \\
&= \sum_{z \neq z'} \frac{1}{2^{6m}} = \binom{2^{3m}}{1}\binom{2^{3m}-1}{1}\frac{1}{2^{6m}} \\
&= \frac{2^{6m} - 2^{3m}}{2^{6m}} = 1 - \frac{1}{2^{3m}}
\end{aligned}$$

Zjednotením cez všetkých $2^m(2^m - 1)/2 \leq 2^{2m}$ rôznych hodnôt $\alpha_M \neq \alpha_{M'}$, pre ktoré platí (2.1), dostávame pravdepodobnosť, že také $\alpha_M \neq \alpha_{M'}$, pre ktoré neplatí (2.1), je najviac $2^{2m}/2^{3m} = 2^{-1m}$. Ak útočník $\mathcal{A}$ dá na výstup $M' \neq M$, pričom $C_{6P}(M) = C_{6P}(M')$, s pravdepodobnosťou aspoň $(1 - 2^{-m})$ platí $\alpha_M = \alpha_{M'}$. Preto v permutácii Feistel$^2(\cdot)$ nemohla nastať kolízia (keďže $\alpha_M = \alpha_{M'}$ s pravdepodobnosťou aspoň $1 - 2^{-m}$), teda musela nastať jedine pred ňou, pred vstupom do Feistel$^2(\cdot)$. Výhoda útočníka $\mathcal{B}_b$, že nájde správu $00 \parallel M'$ (resp. $00 \parallel \pi(M')$), pre správu $00 \parallel M$ (resp. $00 \parallel \pi(M)$) je aspoň taká, ako výhoda útočníka $\mathcal{A}$. Lenže pre funkciu H_b sa správa $00 \parallel M$ nevolila náhodne, ale iba jej časť M (resp. $\pi(M)$). Pravdepodobnosť, že sa náhodne vyberie reťazec 00 je 1/4. Celkovo dostávame, že výhoda útočníka $\mathcal{B}_b$ je $\mathbf{Adv}^{\mathrm{Sec}}_{H_b}(\mathcal{B}_b) \geq \frac{1}{4}\varepsilon(n)$. Predpokladali sme, že funkcia H_b je Sec bezpečná, preto je výhoda útočníka $\mathcal{B}_b$ $\mathbf{Adv}^{\mathrm{Sec}}_{H_b}(\mathcal{B}_b)$ zanedbateľná, a teda aj hodnota $\varepsilon(n)$ je zanedbateľná. □

Veta 2.2.3. *Kominátor C_{6P} je aSec-robustný.*

Dôkaz. Predpokladajme, že aspoň jedna z rodín hašovacích funkcií je aSec bezpečná, označme ju H_b, $b \in \{0, 1\}$. Ďalej predpokladajme, že máme útočníka $\mathcal{A}$ útočiaceho na C_{6P} v zmysle aSec a jeho výhoda je $\varepsilon(n)$. Útočník $\mathcal{A}$ dá najprv na výstup trojicu $(K_b, K_{\bar{b}}, S)$, potom na vstup dostane náhodne vybranú správu M a stav S. Útočník $\mathcal{A}$ vypočíta správu M' takú, že $C_{6P}(M) = C_{6P}(M')$, ale $M' \neq M$. Tohto útočníka použijeme na zostrojenie útočníka $\mathcal{B}_b$ na funkciu H_b v zmysle aSec, o ktorej predpokladáme, že je aSec bezpečná. Konštrukcia útočníka:

Útočník $\mathcal{B}_{\mathbf{b}}^{\text{aSec}}$
[1. fáza]
$(K_b, K_{\bar{b}}, S) \leftarrow \mathcal{A}$
return (K_b, S)
[2. fáza so vstupom $M \xleftarrow{\$} \mathcal{M}$]
$M' \leftarrow \mathcal{A}(M, S)$
ak $b = 0$ potom **return** $00 \parallel \pi(M')$
inak **return** $00 \parallel M'$

Predpokladajme, že $\mathcal{A}$ dá na výstup správu $M' \neq M$, pričom $C_{6P}(M) = C_{6P}(M')$, útočník $\mathcal{B}_b$ našiel správu $00 \parallel M'$ (resp. $00 \parallel \pi(M')$) takú, že $M \neq M'$ (resp. $\pi(M) \neq \pi(M')$), ale $H_b(00 \parallel M) = H_b(00 \parallel M')$ (resp. $H_b(00 \parallel \pi(M)) = H_b(00 \parallel \pi(M'))$). Opäť kolízia mohla nastať jedine pred permutáciou Feistel$^2(\cdot)$, pričom $\alpha_M = \alpha_{M'}$ s pravdepodobnosťou $1 - 2^{-m}$ (viď. dôkaz vety 2.2.2). Výhoda útočníka $\mathcal{B}_b$, že nájde správu $00 \parallel M'$, pre správu $00 \parallel M$ je aspoň taká, ako výhoda útočníka $\mathcal{A}$. Lenže pre funkciu H_b sa správa $00 \parallel M$ nevolila náhodne, ale iba jej časť M. Pravdepodobnosť, že sa náhodne vyberie reťazec 00 je 1/4. Celkovo dostávame, že výhoda útočníka $\mathcal{B}_b$ je $\mathbf{Adv}_{H_b}^{\text{aSec}}(\mathcal{B}_b) \geq \frac{1}{4}\varepsilon(n)$. Keďže H_b je aSec, hodnota $\varepsilon(n)$ je zanedbateľná. □

Veta 2.2.4. *Kominátor C_{6P} je Ctfp-robustný.*

Dôkaz. Predpokladajme, že aspoň jedna z rodín hašovacích funkcií použitých v kombinátore je Ctfp bezpečná, označme ju H_b, $b \in \{0, 1\}$. Predpokladajme, že existuje úspešný útočník $\mathcal{A}$, ktorý útočí na C_{6P} v zmysle Ctfp. Útočník $\mathcal{A}$ dostane na vstup náhodne vygenerované kľúče (špecifikujú rodiny hašovacích funkcií) a vygeneruje dvojicu (Y, S). Potom dostane útočník $\mathcal{A}$ na vstup náhodne zvolený reťazec P a stav S. Útočník $\mathcal{A}$ vypočíta reťazec M taký, že $C_{6P}(P \parallel M) = Y$ s pravdepodobnosťou $\varepsilon(n)$. Tohto útočníka využijeme na zostrojenie útočníka $\mathcal{B}_b$ útočiaceho na rodinu hašovacích funkcií H_b v zmysle Ctfp.

Útočník $\mathcal{B}_{\mathbf{b}}^{\text{Ctfp}}$
[1. fáza so vstupom $K_b \xleftarrow{\$} \mathcal{K}$]
$K_{\bar{b}} \xleftarrow{\$} \mathcal{K}$
$(Y, S) \leftarrow \mathcal{A}(K_b, K_{\bar{b}})$
$P_0 \xleftarrow{\$} \{0,1\}^m$
$M_0 \leftarrow \mathcal{A}(P_0, S)$
ak $b = 0$ potom $Y_b = H_b(00 \parallel \pi(P_0 \parallel M_0))$
inak $Y_b = H_b(00 \parallel P_0 \parallel M_0)$
return (Y_b, S)
[2. fáza so vstupom $P_1 \xleftarrow{\$} \{0,1\}^m$ a S]
$M_1 \leftarrow \mathcal{A}(P_1, S)$
return M_1

Útočník $\mathcal{A}$ dá na výstup reťazce M_0 a M_1 také, že pre náhodne vygenerované správy P_0 a P_1 platí: $C_{6P}^{H_0,H_1,\pi,g}(P_0 \parallel M_0) = C_{6P}^{H_0,H_1,\pi,g}(P_1 \parallel M_1)$. S pravdepodobnosťou $1 - 2^{-m}$ (viď. dôkaz vety 2.2.2) platí, že $\alpha_{P_0\parallel M_0} = \alpha_{P_1\parallel M_1}$, preto kolízia pre správy $P_0 \parallel M_0$ a $P_1 \parallel M_1$ musela nastať ešte pred vstupom do permutácie Feistel$^2(\cdot)$. Potom

$$Y_0 = H_0^0(\pi(P_0 \parallel M_0)) \parallel H_1^0(P_0 \parallel M_0) = H_0^0(\pi(P_1 \parallel M_1)) \parallel H_1^0(P_1 \parallel M_1) = Y_1$$

Nech $\mathcal{E}_0$ označuje udalosť $Y = C_{6P}^{H_0,H_1,\pi,g}(K_0, K_1, P_0 \parallel M_0)$ (tzn. $\mathcal{A}$ vyhrá v prvej simulácii) a $\mathcal{E}_1$ je udalosť $Y = C_{6P}^{H_0,H_1,\pi,g}(K_0, K_1, P_1 \parallel M_1)$ ($\mathcal{A}$ vyhrá v druhej simulácii). Označme $K = K_0 \parallel K_1$ a $AVYHRA(P, Y, K) \Leftrightarrow M \leftarrow \mathcal{A}(K, P) \wedge C_{6P}^{H_0,H_1,\pi,g}(K_0, K_1, P \parallel M) = Y$. Pravdepodobnosť výhody útočníka $\mathcal{A}$ násobíme 1/4, pretože útočník $\mathcal{B}_b$ nedostal správu $00 \parallel P_1$ náhodne, ale iba jej časť P_1, reťazec 00 nastane s pravdepodobnosťou 1/4. Pre výhodu útočníka $\mathcal{B}_b$ platí:

$$\begin{aligned}
\mathbf{Adv}_{H_b}^{\text{Ctfp}}(\mathcal{B}_b) &\geq \frac{1}{4}\Pr\left[K \xleftarrow{\$} \{0,1\}^{2k}, P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1 \wedge \alpha_{P_0\parallel M_0} = \alpha_{P_1\parallel M_1}\right] = \\
&= \frac{1}{4}\Pr\left[K \xleftarrow{\$} \{0,1\}^{2k}; P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1\right] - \\
&\quad - \frac{1}{4}\Pr\left[K \xleftarrow{\$} \{0,1\}^{2k}; P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1 \wedge \alpha_{P_0\parallel M_0} \neq \alpha_{P_1\parallel M_1}\right] \geq \\
&\geq \frac{1}{4}\Pr\left[K \xleftarrow{\$} \{0,1\}^{2k}; P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1\right] - \frac{1}{4}2^{-m} = \\
&= \frac{1}{4}\Pr\left[K \xleftarrow{\$} \{0,1\}^{2k}; P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1\right] - \text{negl}(n) \qquad (2.2)
\end{aligned}$$

Nájdeme dolnú hranicu pre člena výrazu (2.2). Udalosti $\mathcal{E}_0$ a $\mathcal{E}_1$ zdieľajú ten istý, náhodne vybraný kľúč $K = K_0 \parallel K_1$, a teda nemusia byť nezávislé. Preto

$$\begin{aligned}
&\Pr[K \xleftarrow{\$} \{0,1\}^{2k}; P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1] = \\
&\quad = \frac{1}{2^{2k}} \sum_{K \in \{0,1\}^{2k}} \Pr[P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1] = \\
&\quad = \frac{1}{2^{2k}} \sum_{K \in \{0,1\}^{2k}} \Pr[P_0 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0] \cdot \Pr[P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_1] = \\
&\quad = \frac{1}{2^{2k}} \sum_{K \in \{0,1\}^{2k}} \left(\Pr[P \xleftarrow{\$} \{0,1\}^m; AVYHRA(P,Y,K)] \right)^2 \qquad (2.3)
\end{aligned}$$

Nech $DOBRE \subseteq \{0,1\}^{2k}$ označuje množinu kľúčov $K = K_0 \parallel K_1$, pre ktorú je pravdepodobnosť, že $\mathcal{A}$ vyhrá aspoň $\varepsilon(n)/2$. To znamená

$$\forall K_0 \parallel K_1 \in DOBRE : \Pr[K_1, K_2 \xleftarrow{\$} \{0,1\}^k; (Y,S) \leftarrow C_{6P}(K_0, K_1); P \xleftarrow{\$} \{0,1\}^m;$$

$$M \leftarrow \mathcal{A}(P,S) : C_{6P}(P \parallel M) = Y] \geq \frac{\varepsilon(n)}{2}$$

ZLE bude množina všetkých ostatných kľúčov. Rovnosť (2.3) môžeme ohraničiť:

$$\begin{aligned}
&\frac{1}{2^{2k}} \sum_{K \in \{0,1\}^{2k}} \left(\Pr[P \xleftarrow{\$} \{0,1\}^m; AVYHRA(P,Y,K)] \right)^2 = \\
&\quad = \frac{1}{2^{2k}} \sum_{K \in DOBRE} \left(\Pr[P \xleftarrow{\$} \{0,1\}^m; AVYHRA(P,Y,K)] \right)^2 + \\
&\qquad + \frac{1}{2^{2k}} \sum_{K \in ZLE} \left(\Pr[P \xleftarrow{\$} \{0,1\}^m; AVYHRA(P,Y,K)] \right)^2 \geq \\
&\quad \geq \frac{1}{2^{2k}} \sum_{K \in DOBRE} \frac{\varepsilon(n)^2}{4} \qquad (2.4)
\end{aligned}$$

Na druhej strane vieme, že:

$$\begin{aligned}
\varepsilon(n) &= \frac{1}{2^{2k}} \sum_{K \in DOBRE} \Pr[P \xleftarrow{\$} \{0,1\}^m; AVYHRA(P,Y,K)] + \\
&\quad + \frac{1}{2^{2k}} \sum_{K \in ZLE} \Pr[P \xleftarrow{\$} \{0,1\}^m; AVYHRA(P,Y,K)] \leq \\
&\leq \frac{1}{2^{2k}} \sum_{K \in DOBRE} 1 + \frac{1}{2^{2k}} \sum_{K \in ZLE} \frac{\varepsilon(n)}{2} = \\
&= \frac{1}{2^{2k}} \big(|DOBRE| + \frac{\varepsilon(n)}{2} |ZLE| \big) = \\
&= \frac{1}{2^{2k}} \big(|DOBRE| + (2^{2k} - |DOBRE|) \frac{\varepsilon(n)}{2} \big) = \\
&= \frac{1}{2^{2k}} \big(|DOBRE| + \frac{2^{2k} \cdot \varepsilon(n)}{2} - \frac{|DOBRE| \cdot \varepsilon(n)}{2} \big) = \\
&= \frac{1}{2^{2k}} \big(|DOBRE| (1 - \frac{\varepsilon(n)}{2}) + 2^{2k} \frac{\varepsilon(n)}{2} \big) = \\
&= |DOBRE| \frac{2 - \varepsilon(n)}{2^{2k+1}} + \frac{\varepsilon(n)}{2} \\
\text{a teda } |DOBRE| &\geq \frac{\varepsilon(n) \cdot 2^{2k+1}}{2(2 - \varepsilon(n))} = \frac{2^{2k} \cdot \varepsilon(n)}{2 - \varepsilon(n)} \qquad (2.5)
\end{aligned}$$

Kombinovaním rovností a nerovností (2.2), (2.3), (2.4), (2.5) dostávame:

$$\begin{aligned}
\mathbf{Adv}^{\mathrm{Ctfp}}_{H_b}(\mathcal{B}_b) &\geq \frac{1}{2^{2k}} \sum_{K \in \{0,1\}^{2k}} \frac{1}{4} \big(\Pr[P_1 \xleftarrow{\$} \{0,1\}^m, \mathrm{AVYHRA}(P_1,Y,K)] \big)^2 - \mathrm{negl(n)} \geq \\
&\geq \frac{1}{2^{2k}} \sum_{K \in \mathrm{DOBRE}} \frac{1}{4} \frac{\varepsilon(n)^2}{4} - \mathrm{negl(n)} \geq \\
&\geq \frac{1}{2^{2k}} \frac{2^{2k} \cdot \varepsilon(n)}{2 - \varepsilon(n)} \cdot \frac{1}{4} \cdot \frac{\varepsilon(n)^2}{4} - \mathrm{negl(n)} = \\
&= \frac{\varepsilon(n)^3}{16(2 - \varepsilon(n))} - \mathrm{negl(n)} \geq \\
&\geq \frac{\varepsilon(n)^3}{16 \cdot 2} - \mathrm{negl(n)}
\end{aligned}$$

H_b je Ctfp bezpečná, preto musí byť hodnota $\varepsilon(n)$ zanedbateľná. □

Veta 2.2.5. *Kominátor C_{6P} je aCtfp-robustný.*

Dôkaz. Predpokladajme, že aspoň jedna z rodín hašovacích funkcií použitých v kombinátore je aCtfp bezpečná, označme ju H_b, $b \in \{0,1\}$. Predpokladajme, že existuje úspešný

útočník $\mathcal{A}$, ktorý útočí na C_{6P} v zmysle aCtfp. Teda výhoda útočníka $\mathcal{A}$ je

$$\varepsilon(n) = \Pr\left[(Y, K_1, K_2, S) \leftarrow \mathcal{A}; P \xleftarrow{\$} \{0,1\}^m; M \leftarrow \mathcal{A}(P, S) : C_{6P}(P \parallel M) = Y\right].$$

Útočníka $\mathcal{A}$ využijeme na zostrojenie útočníka $\mathcal{B}_b$ útočiaceho na rodinu hašovacích funkcií H_b.

Útočník $\mathcal{B}_{\mathbf{b}}^{\text{aCtfp}}$
[1. fáza so vstupom]
$(Y, K_b, K_{\bar{b}}, S) \leftarrow \mathcal{A}$
$P_0 \xleftarrow{\$} \{0,1\}^m$
$M_0 \leftarrow \mathcal{A}(P_0, S)$
ak $b = 0$ potom $Y_b = H_b(00 \parallel \pi(P_0 \parallel M_0))$
inak $Y_b = H_b(00 \parallel P_0 \parallel M_0)$
return (Y_b, K_b, S)
[2. fáza so vstupom $S, P_1 \xleftarrow{\$} \{0,1\}^m$]
$M_1 \leftarrow \mathcal{A}(P_1, S)$
return M_1

Predpokladajme, že útočník $\mathcal{A}$ dá na výstup reťazce M_0 a M_1 také, že pre náhodne vygenerované správy P_0 a P_1 platí: $C_{6P}^{H_0,H_1,\pi,g}(P_0 \parallel M_0) = C_{6P}^{H_0,H_1,\pi,g}(P_1 \parallel M_1)$. S pravdepodobnosťou $1 - 2^{-m}$ (viď. dôkaz vety 2.2.2) platí, že $\alpha_{P_0\parallel M_0} = \alpha_{P_1\parallel M_1}$, preto kolízia pre správy $P_0 \parallel M_0$ a $P_1 \parallel M_1$ musela nastať ešte pred vstupom do permutácie Feistel$^2(\cdot)$. Potom

$$Y_0 = H_0^0(\pi(P_0 \parallel M_0)) \parallel H_1^0(P_0 \parallel M_0) = H_0^0(\pi(P_1 \parallel M_1)) \parallel H_1^0(P_1 \parallel M_1) = Y_1$$

Nech $\mathcal{E}_0$ označuje udalosť $Y = C_{6P}^{H_0,H_1,\pi,g}(K_0, K_1, P_0 \parallel M_0)$ (tzn. $\mathcal{A}$ vyhrá v prvej simulácii) a $\mathcal{E}_1$ je udalosť $Y = C_{6P}^{H_0,H_1,\pi,g}(K_0, K_1, P_1 \parallel M_1)$ ($\mathcal{A}$ vyhrá v druhej simulácii). Pravdepodobnosť výhody útočníka $\mathcal{A}$ sa násobí 1/4, pretože útočník $\mathcal{B}_b$ nedostal správu $00 \parallel P_1$

náhodne, ale iba jej časť P_1, reťazec 00 sa náhodne vyberie s pravdepodobnosťou 1/4.

$$\begin{aligned}
\mathbf{Adv}^{\mathrm{aCtfp}}_{H_b}(\mathcal{B}_b) &\geq \frac{1}{4}\Pr\left[P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1 \wedge \alpha_{P_0\|M_0} = \alpha_{P_1\|M_1}\right] = \\
&= \frac{1}{4}\Pr\left[P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1\right] \\
&\quad - \frac{1}{4}\Pr\left[P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1 \wedge \alpha_{P_0\|M_0} \neq \alpha_{P_1\|M_1}\right] = \\
&= \frac{1}{4}\Pr\left[P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1\right] - \frac{1}{4}2^{-m} \geq \\
&\geq \frac{1}{4}\Pr\left[P_0, P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0 \wedge \mathcal{E}_1\right] - \mathrm{negl}(n) \\
&\geq \frac{1}{4}\big(\Pr[P_0 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_0] \cdot \Pr[P_1 \xleftarrow{\$} \{0,1\}^m; \mathcal{E}_1]\big) - \mathrm{negl}(n) = \\
&= \frac{1}{4}\varepsilon(n)^2 - \mathrm{negl}(n)
\end{aligned}$$

Keďže H_b je aCtfp bezpečná, musí byť $\varepsilon(n)$ zanedbateľné. □

Veta 2.2.6. *Kominátor C_{6P} je ePre-robustný.*

Dôkaz. Dokážeme, že kombinátor $C_{CR\&OW}(M) = H_0(\pi(M)) \parallel H_1(M)$ je ePre robustný. Predpokladajme, že aspoň jedna z rodín hašovacích funkcií v $C_{CR\&OW}$ je ePre bezpečná, označme ju H_b, $b \in \{0,1\}$. Ďalej predpokladajme, že máme útočníka $\mathcal{A}$ útočiaceho na $C_{CR\&OW}$ v zmysle ePre a jeho výhoda je $\varepsilon(n)$. Tohto útočníka použijeme na zostrojenie útočníka $\mathcal{B}_b$ na funkciu H_b v zmysle ePre, o ktorej predpokladáme, že je ePre bezpečná. Konštrukcia útočníka:

Útočník $\mathcal{B}^{\mathrm{ePre}}_{\mathbf{b}}$
[1. fáza]
  $(Y, S) \leftarrow \mathcal{A}$
  $Y = Y_0 \parallel Y_1$
  return (Y_b, S)
[2. fáza so vstupom $K_b \xleftarrow{\$} \{0,1\}^k$]
  $K_{\bar{b}} \xleftarrow{\$} \{0,1\}^k$
  $M \leftarrow \mathcal{A}(K_b, K_{\bar{b}}, S)$
  ak $b = 0$ potom **return** $\pi(M)$
  inak **return** M

Predpokladajme, že útočník $\mathcal{A}$ dá na výstup správu M, pričom $C_{CR\&OW}(M) = Y$, útočník $\mathcal{B}_b$ našiel správu M (resp. $\pi(M')$) takú, že $H_b(M) = Y_b$ (resp. $H_b(\pi(M)) = Y_b$ pre

$b = 0$). Zjavne platí, že výhoda útočníka $\mathcal{B}_b$ je aspoň taká, ako výhoda útočníka $\mathcal{A}$, teda $\mathbf{Adv}^{\mathrm{ePre}}_{H_b}(\mathcal{B}_b) \geq \varepsilon(n)$. Hodnota $\varepsilon(n)$ je zanedbateľná, lebo rodina hašovacích funkcií H_b je ePre bezpečná.
Aplikovaním permutácie sa neporuší vlastnosť ePre, preto je kombinátor C_{6P} ePre-robustný.
□

Robustnosť kombinátora C_{6P} pre vlastnosť aPre sa nám nepodarila dokázať, a preto ostáva otvoreným problémom.

Kapitola 3

Kombinovanie vlastností hašovacích funkcií

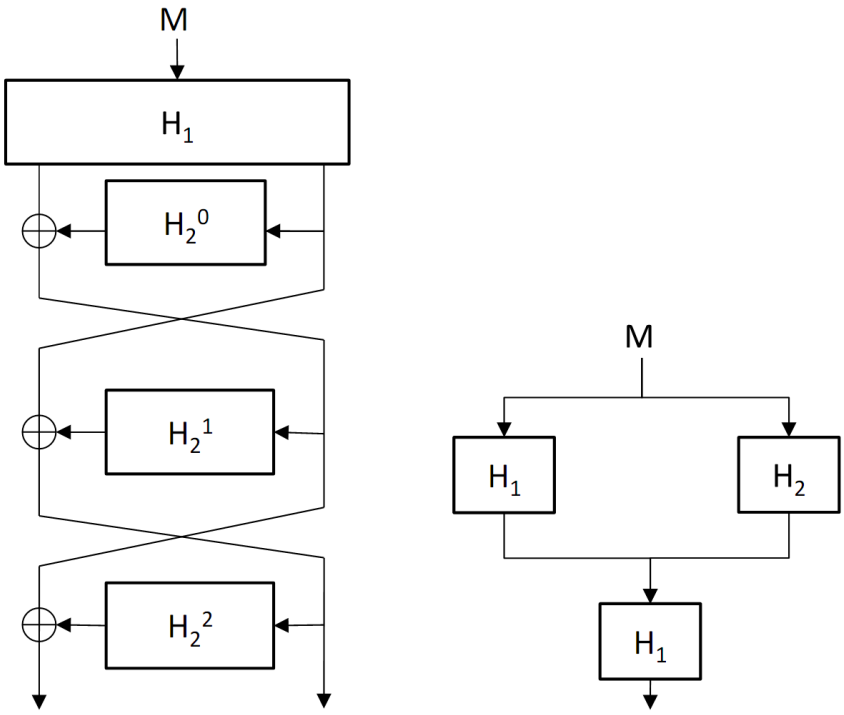
V tejto časti sa budeme venovať inému prístupu na konštrukciu kombinátorov. Majme rodiny hašovacích funkcií $H_0, H_1, \ldots, H_l$ a vlastnosti hašovacích funkcií $P_1, P_2, \ldots, P_l$, pričom platí, že rodina hašovacích funkcií H_i je P_i bezpečná. Budeme hovoriť, že konštrukcia $C^{H_i, H_{i+1} \ldots, H_{i+r}}$, ktorá je kombinátorom funkcií $H_i, H_{i+1}, \ldots, H_{i+r}$, zachováva vlasnosť P_j, ak H_j je P_j bezpečná. To znamená, že nepredpokladáme, že aspoň jedna z použitých rodín hašovacích funkcií je P_j bezpečná, ale predpokladáme, že konkrétna rodina hašovacích funkcií je P_j bezpečná.
Zoberieme konštrukcie C_1 a C_2, o ktorých Rjaško v [16] dokázal, že C_1 kombinuje vlastnosti Prf a Coll a C_2 kombinuje vlastnosti Coll a ePre. O konštrukcii C_1 dokážeme, že je aSec a aCtfp bezpečná. Pre kombinátor zreťazenia dokážeme, že je aSec a aCtfp-robustný. Na záver skombinuje všetky konštrukcie, dostaneme konštrukciu C_4, ktorá je Coll, Sec, eSec, aSec, Pre, ePre, aPre, Mac, Ctfp, aCtfp a Prf bezpečná.

3.1 Konštrukcia C_1

Konštrukcia C_1 znázornená na obrázku 3.1 je definovaná pre rodiny hašovacích funkcií H_1, H_2:$\{0,1\}^k \times \{0,1\}^* \to \{0,1\}^y$ nasledovne:

$$C_1^{H_1,H_2}(K_1, K_2, M) = \text{Feistel}_{H_1^1, H_1^2, H_1^3}(H_2(K_2, M)),$$

kde $H_1^i(M) = \text{Prefix}_{y/2}(H_1(K_1, \langle i \rangle_2 \parallel M))$.
Rjaško dokázal, že zachováva vlastnosti Coll a Prf, ak H_1 je Prf a H_2 je Coll. Vlastnosť Coll implikuje vlastnosť Pre, Sec, eSec a Ctfp. Podobne Prf implikuje Mac. Za predpokladu, že H_1 je Prf a H_2 je Coll máme konštrukciu, ktorá je Coll, Prf, Mac, Pre, Sec, eSec a Ctfp bezpečná. My dokážeme, že C_1 je aSec a aCtfp bezpečná, ak H_2 je aSec a aCtfp, pričom H_1 je Prf.



Obr. 3.1: Konštrukcia C_1 (vľavo) a konštrukcia C_2 (vpravo) [16]

Veta 3.1.1. *Konštrukcia C_1 je aSec, ak H_1 je Prf a H_2 je aSec.*

Dôkaz. Predpokladajme, že máme útočníka $\mathcal{A}$ útočiaceho na C_1 v zmysle aSec, označme výhodu jeho úspechu $\varepsilon(n)$. Tohto útočníka použijeme na zostrojenie útočníka $\mathcal{B}$ útočiaceho na H_2 nasledovne:

Útočník $\mathcal{B}^{\text{aSec}}$
[1. fáza]
$(K_1, K_2, S) \leftarrow \mathcal{A}$
return (K_2, S)
[2. fáza so vstupom $M \xleftarrow{\$} \{0,1\}^m$]
$M' \leftarrow \mathcal{A}(M, S)$
return M'

Predpokladajme, že útočník $\mathcal{A}$ dá na výstup správu $M' \neq M$, pričom $C_1(M) = C_1(M')$, potom útočník $\mathcal{B}$ našiel správu M' takú, že $M \neq M'$, ale $H_2(M) = H_2(M')$. Kolízia mohla nastať jedine pred Feistelovskou permutáciou $\text{Feistel}_{H_1^1, H_1^2, H_1^3}$. Zjavne výhoda útočníka $\mathcal{B}$ je aspoň taká, ako výhoda útočníka $\mathcal{A}$, teda $\mathbf{Adv}^{\text{aSec}}_{H_2}(\mathcal{B}) \geq \varepsilon(n)$. Keďže H_2 je aSec, výhoda útočníka $\mathcal{B}$ je zanedbateľná, a teda aj výhoda útočníka $\mathcal{A}$ $\varepsilon(n)$ je zanedbateľná. □

Veta 3.1.2. *Konštrukcia C_1 je aCtfp, ak H_1 je Prf a H_2 je aCtfp.*

Dôkaz. Dokážeme, že výhoda úspešného útočníka $\mathcal{A}$ útočiaceho na C_1 v zmysle aCtfp je zanedbateľná. Nech výhoda úspechu $\mathcal{A}$ je $\varepsilon(n)$. Pomocou útočníka $\mathcal{A}$ zostrojíme útočníka $\mathcal{B}$ útočiaceho na H_2 :

Útočník $\mathcal{B}^{\text{aCtfp}}$
[1. fáza]
$(Y, K_1, K_2, S) \leftarrow \mathcal{A}$
$Y' = \text{Feistel}^{-1}(Y)$
return (Y', K_2, S)
[2. fáza so vstupom $P \xleftarrow{\$} \{0,1\}^m$]
$M \leftarrow \mathcal{A}(P, S)$
return M

Predpokladajme, že útočník $\mathcal{A}$ dá na výstup správu M, pričom $C_1(P \parallel M) = Y$, potom útočník $\mathcal{B}$ našiel správu M takú, že $H_2(P \parallel M) = Y'$. Výhoda útočníka $\mathcal{B}$ je aspoň taká, ako výhoda útočníka $\mathcal{A}$, teda $\mathbf{Adv}^{\text{aCtfp}}_{H_2}(\mathcal{B}) \geq \varepsilon(n)$, ale H_2 je aCtfp bezpečná, preto výhoda útočníka $\mathcal{B}$ je zanedbateľná, a teda aj výhoda útočníka $\mathcal{A}$ $\varepsilon(n)$ je zanedbateľná. □

3.2 Konštrukcia C_2

Nech H_1, H_2:$\{0,1\}^k \times \{0,1\}^* \to \{0,1\}^y$ sú rodiny hašovacích funkcií.

$$C_2^{H_1,H_2}(K_1, K_2, M) = H_1(K_1, H_1(K_1, M) \parallel H_2(K_2, M))$$

O konštrukcii C_2 znázornenej na obrázku 3.1 dokázal Rjaško, že zachováva vlastnosti Coll a ePre, ak H_1 je Coll a H_2 je ePre. My ukážeme, prečo vo všeobecnosti C_2 nie je aSec, ak H_1 je Coll a H_2 je aSec.

Predpokladajme, že máme útočníka $\mathcal{A}$ útočiaceho na C_2 v zmysle aSec. S využitím tohto útočníka zostrojíme útočníka $\mathcal{B}_2$ na rodinu hašovacích funkcií H_2:

Útočník $\mathcal{B}_2^{\text{aSec}}$
[1. fáza]
$(K_1, K_2, S) \leftarrow \mathcal{A}$
return (K_2, S)
[2. fáza so vstupom $M \xleftarrow{\$} \{0,1\}^m$]
$M' \leftarrow \mathcal{A}(M, S)$
return M'

Môžu nastať dva prípady. Ak $H_2(K_2, M) = H_2(K_2, M')$, útočník $\mathcal{B}_2$ je úspešný, inak je dvojica $(H_1(K_1, M) \parallel H_2(K_2, M), H_1(K_1, M') \parallel H_2(K_2, M'))$ kolízia pre H_1. To znamená, že druhá možnosť môže nastať len so zanedbateľnou pravdepodobnosťou, a teda útočník $\mathcal{B}_2$ našiel hľadanú správu M'. Lenže vlastnosť Coll je definovaná pre náhodne zvolené kľúče (útočník $\mathcal{A}$ kľúče K_1 a K_2 v konštrukcii útočníka $\mathcal{B}_2$ vypočítal a mohol ich vybrať tak, aby sa kolízie v $H_1(K_1, \cdot)$ hľadali ľahko). Preto kolízia v $H_1(K_1, \cdot)$ môže nastať s nezanedbateľnou pravdepodobnosťou a nemusí platiť, že $H_2(K_2, M) = H_2(K_2, M')$.

3.3 Konštrukcia C_3

Konštrukciou C_3 budeme v tejto kapitole označovať kombinátor zreťazenia dvoch rodín hašovacích funkcií. Nech H_1, H_2:$\{0,1\}^k \times \{0,1\}^* \to \{0,1\}^y$ sú rodiny hašovacích funkcií. Konštrukcia C_3 je definovaná:

$$C_3^{H_1,H_2}(M) = H_1(M) \parallel H_2(M)$$

Veta 3.3.1. *Konštrukcia C_3 je aSec-robustná.*

Dôkaz. Predpokladajme, že aspoň jedna z rodín hašovacích funkcií je aSec bezpečná, označme ju H_b, $b \in \{0,1\}$. Ďalej predpokladajme, že máme útočníka $\mathcal{A}$ útočiaceho na C_3 v zmysle aSec a jeho výhoda je $\varepsilon(n)$. Tohto útočníka použijeme na zostrojenie útočníka $\mathcal{B}_b$ na funkciu H_b v zmysle aSec, o ktorej predpokladáme, že je aSec bezpečná. Konštrukcia útočníka:

Útočník $\mathcal{B}_{\mathbf{b}}^{\text{aSec}}$
[1. fáza]
$(K_b, K_{\bar{b}}, S) \leftarrow \mathcal{A}$
return (K_b, S)
[2. fáza so vstupom $M \xleftarrow{\$} \mathcal{M}$]
$M' \leftarrow \mathcal{A}(M, S)$;
return M'

Predpokladajme, že útočník $\mathcal{A}$ dá na výstup správu $M' \neq M$, pričom $C_3(M) = C_3(M')$. Potom útočník $\mathcal{B}_b$ našiel správu M' takú, že $M \neq M'$, ale $H_b(M) = H_b(M')$. Zjavne výhoda útočníka $\mathcal{B}_b$, že nájde správu M', pre správu M je aspoň taká, ako výhoda útočníka $\mathcal{A}$, teda je $\mathbf{Adv}_{H_b}^{\text{aSec}}(\mathcal{B}_b) \geq \varepsilon(n)$. Rodina hašovacích funkcií H_b je aSec bezpečná, preto je hodnota $\varepsilon(n)$ zanedbateľná. □

Veta 3.3.2. *Konštrukcia C_3 je aCtfp-robustná.*

Dôkaz. Predpokladajme, že aspoň jedna z rodín hašovacích funkcií použitých v kombinátore je aCtfp bezpečná, označme ju H_b, $b \in \{0,1\}$. Ďalej predpokladajme, že existuje úspešný útočník $\mathcal{A}$, ktorý útočí na C_3 v zmysle aCtfp, pričom jeho výhoda je $\varepsilon(n)$. Útočníka $\mathcal{A}$ využijeme na zostrojenie útočníka $\mathcal{B}_b$ útočiaceho na rodinu hašovacích funkcií H_b.

Útočník $\mathcal{B}_{\mathbf{b}}^{\text{aCtfp}}$
[1. fáza]
$(Y, K_b, K_{\bar{b}}, S) \leftarrow \mathcal{A}$
$Y = Y_0 \parallel Y_1$
return (Y_b, K_b, S)
[2. fáza so vstupom $P \xleftarrow{\$} \{0,1\}^m$]
$M \leftarrow \mathcal{A}(P, S)$
return M

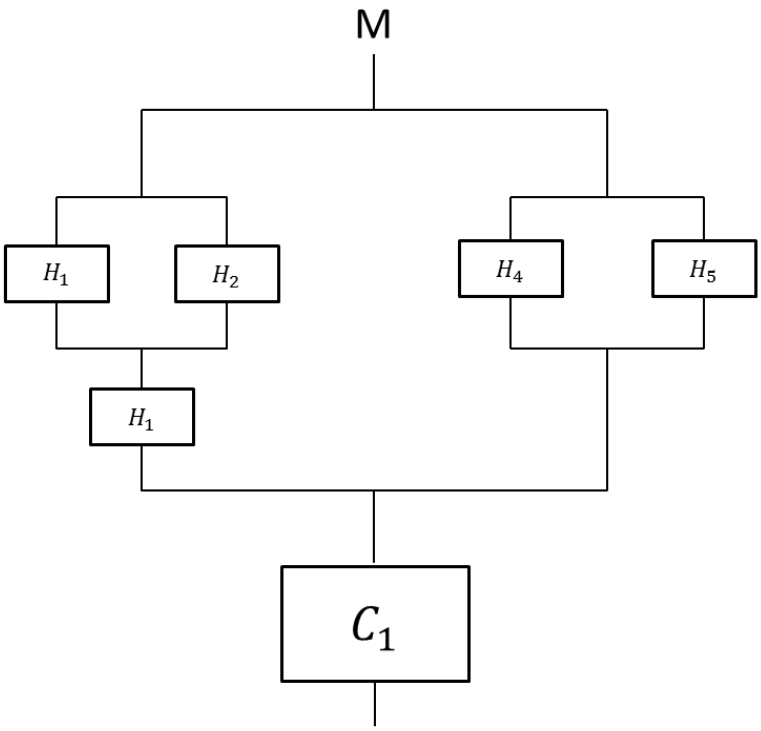
Pre výstup útočníka $\mathcal{A}$, reťazec M, s pravdepodobnosťou $\varepsilon(n)$ platí $C_3(P \parallel M) = Y$. Útočník $\mathcal{B}_b$ s aspoň takou istou pravdepodobnosťou ako pre útočníka $\mathcal{A}$, dá na výstup

reťazec M, pre ktorý platí $H_b(P \parallel M) = Y_b$, teda $\mathbf{Adv}^{\mathrm{aCtfp}}_{H_b}(\mathcal{B}_b) \geq \varepsilon(n)$. Rodina hašovacích funkcií H_b je aCtfp bezpečná, preto je hodnota $\varepsilon(n)$ zanedbateľná. □

3.4 Konštrukcia C_4

Skombinovaním všetkých konštrukcií dostaneme konštrukciu C_4 znázornenú na obrázku 3.2 definovanú nasledovne: Nech $H_1, H_2, H_3, H_4, H_5 : \{0,1\}^k \times \{0,1\}^* \to \{0,1\}^y$ sú rodiny hašovacích funkcií, pričom H_1 je Coll, H_2 je ePre, H_3 je Prf, H_4 je aSec a H_5 je aCtfp.

$$C_4^{H_1,H_2,H_3,H_4,H_5}(M) = C_1^{C_3^{C_2^{H_1,H_2},C_3^{H_4,H_5}},H_3}(K_1, K_2, K_3, K_4, K_5, M)$$



Obr. 3.2: Konštrukcia C_4

Veta 3.4.1. *Konštrukcia C_4 je Coll, Prf, ePre, aSec a aCtfp bezpečná, ak H_1 je Coll, H_2 je ePre, H_3 je Prf, H_4 je aSec a H_5 je aCtfp .*

Dôkaz. Nech H_1 je Coll, H_2 je ePre, H_3 je Prf, H_4 je aSec, H_5 je aCtfp bezpečná rodina hašovacích funkcií.

- **odolnosť voči kolíziám**
  Konštrukcia C_2 použitá v konštrukcii C_4 je odolná voči kolízii, ak jej prvým vstupom je Coll bezpečná rodina hašovacích funkcií. Prvým vstupom C_2 v C_4 je rodina hašovacích funkcií H_1, ktorá je odolná voči kolízii, preto je C_2 Coll bezpečná funkcia. Konštrukcia C_2 je zároveň vstupom do C_3, o ktorej bolo dokázané, že je Coll-robustná, teda aj konštrukcia C_3, ktorá je vstupom do C_1, je Coll bezpečná. Na záver dostávame, že konštrukcia C_4 je Coll bezpečná, pretože C_1 zachováva vlastnosť Coll.

- **ePre bezpečnosť**
  Konštrukcia C_2 použitá v konštrukcii C_4 je ePre bezpečná, ak jej prvým vstupom je Coll bezpečná rodina hašovacích funkcií a druhým vstupom je ePre bezpečná rodina hašovacích funkcií. Prvým vstupom C_2 v C_4 je rodina hašovacích funkcií odolná voči kolízii a druhým vstupom je ePre bezpečná rodina hašovacích funkcií, preto je C_2 ePre bezpečná. Konštrukcia C_2 je zároveň vstupom do C_3, ktorá je ePre-robustná, teda aj konštrukcia C_3, ktorá je vstupom do C_1, je ePre bezpečná. Na záver dostávame, že konštrukcia C_4 je ePre bezpečná, pretože C_1 zachováva vlastnosť ePre.

- **aSec a aCtfp bezpečnosť**
  Konštrukcia C_3 použitá v konštrukcii C_4 je aSec a aCtfp robustná. Vstupom C_3 v C_4 sú rodiny hašovacích funkcií H_4 a H_5, pričom H_4 je aSec a H_5 je aCtfp, preto je C_3 aSec a aCtfp bezpečná. Konštrukcia C_3 je zároveň vstupom do ďalšej konštrukcie C_3 a tá je vstupom do C_1. Na záver dostávame, že konštrukcia C_4 je aSec a aCtfp bezpečná, pretože C_1 zachováva vlastnosť aSec a aCtfp.

- **Prf bezpečnosť**
  Konštrukcia C_1 zachováva vlastnosť Prf, ak jej druhým vstupom je Prf bezpečná rodina hašovacích funkcií. V konštrukcii C_4 je rodina hašovacích funkcií H_3 Prf, tá je zároveň druhým vstupom do konštrukcie C_1 v C_4, preto je C_4 Prf bezpečná.

□

Vlastnosť Coll implikuje vlastnosti Pre, Sec, eSec, Ctfp. Vlastnosť aSec implikuje aPre a vlastnosť Prf implikuje MAC, preto konštrukcia C_4 za daných podmienok, že H_1 je Coll, H_2 je ePre, H_3 je Prf, H_4 je aSec a H_5 je aCtfp je aj Mac, Pre, aPre, Sec, eSec a Ctfp bezpečná.

Záver

V prvej časti tejto práce sme uviedli základné označenia a definície vlastností rodín hašovacích funkcií. Potom sme ukázali ekvivalenciu medzi jednostavovou a dvojstavovou verziou pre vlastnosť Ctfp. Dôkaz ekvivalencie ostatných vlastností je podobný. Ďalej sme uviedli vzťahy medzi vlastnosťami rodín hašovacích funkcií a v závere prvej kapitoly sme sa zaoberali robustnými kombinátormi.

Druhá kapitola našej práce obsahuje dôkazy vlastností pre kombinárory C_{4P} a C_{6P}, ktoré neboli dokázané autormi týchto kombinátorov v [6]. Pre kombinátory C_{4P} a C_{6P} sa nám nepodarilo dokázať ani vyvrátiť aPre robustnosť. Prehľad našich a predchádzajúcich výsledkov je uvedený v tabuľke 1.

V tretej kapitole našej práce sme sa zaoberali iným prístupom na konštrukciu kombinátorov. Skúmali sme konštrukcie kombinátorov, ktoré zachovávajú niektoré vlastnosti rodín hašovacích funkcií za určitých podmienok. Venovali sme sa konštrukciám C_1 a C_2, o ktorých Rjaško v [16] dokázal, že C_1 zachováva vlastnosť Prf a Coll a C_2 zachováva vlastnosť Coll a ePre za stanovených podmienok. O konštrukcii C_1 sme dokázali, že je aSec a aCtfp bezpečná. Ďalej sme ukázali, že konštrukcia C_2 nie je aSec bezpečná, ak rodina hašovacích funkcií H_1 je Coll a H_2 je aSec. Konštrukcia C_3 predstavuje zreťazovací kombinátor, ktorý je ePre, Coll, aSec a aCtfp robustný. Na záver sme skombinovali konštrukcie C_1, C_2, C_3 a dostali sme konštrukciu C_4, ktorá je Coll, Sec, eSec, aSec, Pre, ePre, Mac a Prf bezpečná za určitých predpokladov.

Literatúra

[1] M. Bellare and T. Ristenpart. Hash Functions in the Dedicated-Key Setting: Design Choices and MPP Transforms. In *International Colloquim on Automata, Languages, and Progamming, LNCS vol. 4596*, pages 399–410. Springer, 2006.

[2] D. Boneh and X. Boyen. On the Impossibility of Efficiently Combining Collision Resistant Hash Functions. In *Advances in Cryptology - CRYPTO 2006, LNCS vol. 4117*, pages 570–583. Springer, 2006.

[3] R. Canetti, O. Goldreich, and S. Halevi. The random oracle methodology, revisited. In *Journal of the ACM, vol. 51, issue 4*, pages 557–594. ACM, 2004.

[4] J. Coron, Y. Dodis, C. Malinaud, and P. Puniya. Merkle-Damgård Revisited: How to Construct a Hash Function. In *Advances in Cryptology – CRYPTO 2005, LNCS vol. 3621*, pages 430–448. Springer, 2005.

[5] M. Fischlin and A. Lehman. Multi-property Preserving Combiners for Hash Functions. In *Theory of Cryptography, LNCS vol. 4948*, pages 375–392. Springer, 2008.

[6] M. Fischlin, A. Lehmann, and K. Pietrzak. Robust Multi-property Combiners for Hash Functions Revisited. In *Automata, Languages and Programming, LNCS vol. 5126*, pages 655–666, 2009.

[7] A. Herzberg. Folklore, practice and theory of robust combiners. In *Journal of Computer Security, vol. 17, issue 2*, pages 159–189. IOS Press, 2009.

[8] A. Joux. Multicollisions in Iterated Hash Functions. Application to Cascaded Constructions. In *Advances in Cryptology – CRYPTO 2004, LNCS vol. 3152*, pages 306–316. Springer, 2004.

[9] J. Kelsey and T. Kohno. Herding Hash Functions and the Nostradamus Attack. In *Advances in Cryptology – EUROCRYPT 2006, LNCS vol. 4004*, pages 183–200. Springer, 2006.

[10] V. Klima. Tunnels in Hash Functions: MD5 Collisions Within a Minute. Technical report, Cryptology ePrint Archive, Report 2006/105, 2006.

[11] M. Luby and C. Rackoff. How to construct pseudorandom permutations from pseudorandom functions. In *SIAM Journal on Computing*, volume 17, pages 373–386, 1988.

[12] U. Maurer, R. Renner, and C. Holenstein. Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology. In *Theory of Cryptography, LNCS vol. 2951*, pages 21–39. Springer, 2004.

[13] M. Naor and O. Reingold. From Unpredictability to Indistinguishability: A Simple Construction of PseudoRandom Functions from MACs. In *Advances in Cryptology – CRYPTO '98, LNCS vol. 1462*, pages 267–281. Springer, 1998.

[14] K. Pietrzak. Non-Trivial Black-Box Combiners for Collision-Resistant Hash-Functions don't Exist. In *Advances in Cryptology - EUROCRYPT 2007, LNCS vol. 4515*, pages 23–33. Springer, 2007.

[15] M. Rjasko. Properties of Cryptographic Hash Functions. *Cryptology ePrint Archive, Report 2008/527*, 2008.

[16] M. Rjaško. Combining properties of cryptographic hash functions. Cryptology ePrint Archive, Report 2010/524, 2010. `http://eprint.iacr.org/`.

[17] P. Rogaway. Formalizing Human Ignorance: Collision-Resistant Hashing without the Keys. In *Progress in Cryptology - VIETCRYPT 2006, LNCS vol. 4341*, pages 211–228. Springer, 2006.

[18] P. Rogaway and T. Shrimpton. Cryptographic Hash-Function Basics: Definitions, Implications, and Separations for Preimage Resistance, Second-Preimage Resistance, and Collision Resistance. In *Fast Software Encryption, LNCS vol. 3017*, pages 371–388. Springer, 2004.

[19] D. R. Simon. Finding Collisions on a One-Way Street: Can secure hash functions be based on general assumptions? In *Advances in Cryptology - EUROCRYPT'98, LNCS vol. 1403*, pages 334–345. Springer, 1998.

[20] M. Stanek. Základy kryptológie, 2004. www.dcs.fmph.uniba.sk/štanek/crypto/main2.pdf.

[21] X. Wang, Y. L. Yin, and H. Yu. Finding Collisions in the Full SHA-1. In *Advances in Cryptology – CRYPTO 2005, LNCS vol. 3621*, pages 17–36, 2005.

[22] X. Wang and H. Yu. How to Break MD5 and Other Hash Functions. In *Advances in Cryptology - EUROCRYPT 2005, LNCS vol. 3494*, pages 19–35, 2005.

[23] H. Yu, G. Wang, G. Zhang, and X. Wang. The Second-Preimage Attack on MD4. In *Cryptology and Network Security, LNCS vol. 3810*, pages 1–12, 2005.