Proof of Collision Resistance

Sebastien Duc

First let's see how we can get a collision given that we query the three block ciphers f_1, f_2, f_3. Suppose that we are at the i-th query and no collision occurred. We want to see how we can get a collision given this new query with the old queries in $\mathcal{Q}_{i-1}$. Suppose that the i-th query is an f_1-query (a_1, a_2, a_3). Three possible cases can occur:

Case I. Two compatible and colliding couples $(a_1, a_2, a_3)-(b_1, b_2, b_3, b_4)-(c, d, e)$ and $(a_1', a_2', a_3') - (b_1', b_2', b_3', b_4') - (c', d', e')$ are formed with the quadruple $\{(a_1', a_2', a_3'), (b_1, b_2, b_3, b_4), (c, d, e), (b_1', b_2', b_3', b_4'), (c', d', e')\} \subseteq \mathcal{Q}_{i-1}$ (where $(a_1, a_2, a_3) \neq (a_1', a_2', a_3')$).

Case II. Two distinct compatible and colliding couples exist with $(a_1, a_2, a_3) = (a_1', a_2', a_3')$ and $\{(b_1, b_2, b_3, b_4), (c, d, e), (b_1', b_2', b_3', b_4'), (c', d', e')\} \subseteq \mathcal{Q}_{i-1}$, where $(b_1, b_2, b_3, b_4) \neq (b_1', b_2', b_3', b_4') \vee (c, d, e) \neq (c', d', e')$

As for the 2-dimensional case, we can define find an upper bound on the probability of collision as follows:

$$\Pr[\mathsf{coll}_](\mathcal{Q}) \leq \Pr[E_1] + \Pr[E_2] + \Pr[E_3]$$

with $E_1 = \mathsf{coll}_I(\mathcal{Q}) \wedge \neg\mathsf{bad}_{\mathsf{cl}[\kappa]}(\mathcal{Q})$, $E_2 = \neg\mathsf{coll}_{II}(\mathcal{Q}) \wedge \mathsf{bad}_{\mathsf{cl}[\kappa]}(\mathcal{Q})$, and $E_3 = \mathsf{coll}_{II}(\mathcal{Q})$. $\mathsf{bad}_{\mathsf{cl}[\kappa]}(\mathcal{Q})$ is defined the same way it was defined in the two dimensional case i.e. $\mathsf{bad}_{\mathsf{cl}[\kappa]}(\mathcal{Q})$ is set if and only if $\mathcal{Q}$ leads to more than κ collinear output points in $\mathbb{F}_{2^n}^3$.

Post Processing

For the post processing, we roughly proceed the same as in the two dimensional case.

$$C^{\mathrm{POST}}(a_1, a_2, a_3, b_1, b_2, c, y_1, y_2, y_3) = A \cdot (a_1, b_1, c, y_1, y_2, y_3, y_1y_2, y_1y_3, y_2y_3)^T$$

where $A = \begin{pmatrix} \omega_{11} & \omega_{12} & \omega_{13} & \omega_{14} & \omega_{15} & \omega_{16} & \omega_{17} & \omega_{18} & \omega_{19} \\ \omega_{21} & \omega_{22} & \omega_{23} & \omega_{24} & \omega_{25} & \omega_{26} & \omega_{27} & \omega_{28} & \omega_{29} \\ \omega_{31} & \omega_{32} & \omega_{33} & \omega_{34} & \omega_{35} & \omega_{36} & \omega_{37} & \omega_{38} & \omega_{39} \end{pmatrix}$

We also define lines in the output domain analogously. Let (a_1, a_2, a_3) be an f_1-query, let (b_1, b_2, b_3, b_4) and (c, d, e) be preceding (a_1, a_2, a_3)-compatible f_2

and f_3-query with $y_2 = f_2(b_1, b_2, b_3, b_4)$ and $y_2 = f_3(c, d, e)$. The output of the compression function lies on the line

$$\mathcal{L}_{1:,b_1,b_2,c,y_2,y_3;a_1} : \{B_{1:b_1,c,y_2,y_3;a_1} + y_1 S_{1:y_2,y_3} \mid y_1 \in \mathbb{F}_{2^n}\}$$

$$B_{1:b_1,c,y_2,y_3;a_1} = \begin{pmatrix} a_1\omega_{11} & + & b_1\omega_{12} & + & c\omega_{13} & + & y_2\omega_{15} & + & y_3\omega_{16} & + & y_2y_3\omega_{19} \\ a_1\omega_{21} & + & b_1\omega_{22} & + & c\omega_{23} & + & y_2\omega_{25} & + & y_3\omega_{26} & + & y_2y_3\omega_{29} \\ a_1\omega_{31} & + & b_1\omega_{32} & + & c\omega_{33} & + & y_2\omega_{35} & + & y_3\omega_{36} & + & y_2y_3\omega_{39} \end{pmatrix}$$

$$S_{1:y_2,y_3} = \begin{pmatrix} \omega_{14} & + & y_2\omega_{17} & + & y_3\omega_{18} \\ \omega_{24} & + & y_2\omega_{27} & + & y_3\omega_{28} \\ \omega_{34} & + & y_2\omega_{37} & + & y_3\omega_{38} \end{pmatrix}$$

Similarly we can define $\mathcal{L}_{2:a_1,a_2,a_3,c,y_1,y_3;b_1}$ and $\mathcal{L}_{3:a_1,a_2,a_3,b_1,b_2,y_1,y_2;c}$.

These lines will add some constraints on matrix A as we do not want to have degenerated lines. For that we want that the determinant $|S_{1:y_2,y_3}| \neq 0$ so that the first line is not degenerated. Similarly, we want that $|S_{2:y_1,y_3}| \neq 0$ and $|S_{3:y_1,y_2}| \neq 0$.

Upper Bound on $\Pr[E_1]$

A similar to proof of Lemma 5.2.5 (in the 2-dimensional) can be done and leading to the same bound. Namely

$$\Pr[E_1] \leq \frac{\kappa Y}{2^n}.$$

Lemma. *Let i be a positive integer such that $i \leq q$ and let $\mathcal{Q}_i$ be an arbitrary query list that satisfies $\neg\mathsf{bad}_{\mathsf{cl}[\kappa]}(\mathcal{Q}_i)$. Then the probability that the i-th query causes a collision with an element in* $\mathrm{yieldset}(\mathcal{Q}_{\mathrm{i}-1})$ *can be upper bounded by $n_i\kappa/2^n$, where n_i denotes the number of elements in $\mathcal{Q}_{i-1}$ that are compatible with the i-th query. Furthermore*

$$\Pr[E_1] = \Pr[\mathsf{coll}_I(\mathcal{Q}_\wedge)\mathsf{bad}_{\mathsf{cl}[\kappa]}(\mathcal{Q})] \leq \frac{\kappa Y}{2^n}$$

with $Y = \mathrm{yield}(q)$.

Proof. We have $\neg\mathsf{bad}_{\mathsf{cl}[\kappa]}(\mathcal{Q}_i)$ implying that at most κ points are collinear. As seen above, every compatible couple with the new query defines a line. And at most κ points are on this line, giving a probability of at most $\kappa/2^n$ of collision with this line. As stated in the lemma, there are n_i compatible couples with the i-th query. By applying the union bound, we have that the probability is upper bounded by $n_i\kappa/2^n$. By setting $B_\Sigma = \kappa Y/2^n$ we can apply Proposition 3 (as $\sum_{i=1}^{3q} n_i \leq Y$) and get our upper bound on the probability of E_1. □

Upper Bound on $\Pr[E_3]$

Lemma. *Let i be a positive integer such that $i \leq q$ and let $\mathcal{Q}$ be generated by an adaptive adversary. Then*

$$\Pr[\mathsf{coll}_{II}(\mathcal{Q}_i) \wedge \neg\mathsf{coll}_{II}(\mathcal{Q})i - 1 \wedge \neg\mathsf{bad}_{\mathsf{slc}[\gamma]}(\mathcal{Q}_i)] \leq \frac{\gamma^2}{2^n}$$

and

$$\Pr[E_3] \leq \frac{q\gamma^2}{2^{n-1}} + \Pr[\mathsf{bad}_{\mathsf{slc}[\gamma]}(\mathcal{Q})]$$

for any integer $\gamma > 0$.

Proof. We are studying a type II collision, so suppose $(b_1, b_2, b_3, b_4), (c, d, e)$ and $(b'_1, b'_2, b'_3, b'_4), (c', d', e')$ are distinct (a_1, a_2, a_3)-compatible queries in $\mathcal{Q}_{i-1}$. They are such that $y_2 = f_2(b_1, b_2, b_3, b_4), y'_2 = f_2(b'_1, b'_2, b'_3, b'_4)$ and $y_3 = f_3(c, d, e), y'_3 = f_3(c', d', e')$. We suppose that our i-th query (a_1, a_2, a_3) is such that $y_1 = f_1(a_1, a_2, a_3)$. Given this and the definition of the output lines, we have a type II collision if

$$L = y_1 \begin{pmatrix} (y_2 - y'_2)\omega_{17} + (y_3 - y'_3)\omega_{18} \\ (y_2 - y'_2)\omega_{27} + (y_3 - y'_3)\omega_{28} \\ (y_2 - y'_2)\omega_{37} + (y_3 - y'_3)\omega_{38} \end{pmatrix}$$

with

$$L = \begin{pmatrix} (b_1 - b'_1)\omega_{12} + (c - c')\omega_{13} + (y_2 - y'_2)\omega_{15} + (y_3 - y'_3)\omega_{16} + (y_2 y_3 - y'_2 y'_3)\omega_{19} \\ (b_1 - b'_1)\omega_{22} + (c - c')\omega_{23} + (y_2 - y'_2)\omega_{25} + (y_3 - y'_3)\omega_{26} + (y_2 y_3 - y'_2 y'_3)\omega_{29} \\ (b_1 - b'_1)\omega_{32} + (c - c')\omega_{33} + (y_2 - y'_2)\omega_{35} + (y_3 - y'_3)\omega_{36} + (y_2 y_3 - y'_2 y'_3)\omega_{39} \end{pmatrix}$$

... (I am stuck right here) $\square$