Middle East Technical University
Institute of Applied Mathematics

MARVELlous Family of STARK-Friendly Hash Challenge

Sümeyye Küçükdemir
(Cryptography)

Advisor: Assoc. Prof. Dr. Oğuz YAYLA

Term Project Report
January 2023

Abstract

As arithmetic oriented encryption is included in advanced protocols, block cipher which is effective in software and hardware, has become inconvenient. Therefore, a new design strategy was needed. Upon an agreement between Starkware and the Ethereum Foundation, the search began, and at the end of 2 years, designs which are suitable for the STARK complexity were revealed. It is also worth noting that in addition to being the winner of Rescue, it can give more effective results in different variants of other candidates. In this report, the design rationale of MARVELlous family and the two members of this family, including Rescue, will be examined.

Öz

Aritmetik yönelimli şifreleme gelişmiş protokollerde yer aldığından, yazılım ve donanımda etkili olan blok şifreleme kullanışsız hale gelmiştir. Bu nedenle, yeni bir tasarım stratejisine ihtiyaç doğdu. Starkware ve Ethereum Vakfı arasında yapılan anlaşma ile arayış başladı ve 2 yılın sonunda STARK karmaşıklığına uygun tasarımlar ortaya çıktı. Rescue kazanan olmasının yanı sıra diğer adayların farklı varyantlarında daha etkili sonuçlar verebileceğini de belirtmekte fayda var. Bu raporda, Marvellous ailesi'nin tasarım mantığı ve Rescue dahil bu ailenin iki üyesi incelenecektir.

Contents

Chapter 1

Introduction

One of the basic building blocks of modern cryptography is the block cipher, which uses a symmetric key structure. There are many studies in the literature on behalf of the block cipher. In addition, there is a great deal of information on design and safety principles. The design concerns of traditional block ciphers that are believed to be secure, such as AES, 3DES, SHA2-256, or SHA-3/Keccak, are to make them effective in hardware and software, and these designs achieve that. However, these designs with efficiency are quite different from arithmetic oriented designs.

In recent years, the use of zero-knowledge proof, homomorphic encryption and multi-party computation, which we call the advanced cryptographic protocol, has become widespread. However, the existing symmetric primitives were not found very suitable for applications in this field. Because in traditional symmetric cryptography, algorithms are designed to be effective on hardware and software. In advanced cryptographic protocols, designs are required to be arithmetization-oriented. For example, it is desired to reach a sufficient level of security with low-degree polynomials or a low-degree rational map on a finite field.

Traditional encryption algorithms and arithmetic oriented encryptions are different from each other. Likewise, security analyzes are different. For this, the designer should design new tools to be safe against attacks. Since arithmetic oriented encryption is a very new field, instead of trying to optimize something unknown, it would be a better move to optimize pre-existing design strategies that may be suitable for this field and move an arithmetic-oriented field. For example, one of the differences between these two design models is the order of the large polynomial, which is the most important factor determining the complexity of the algorithm. Compared to the STARK prover, the complexity level of traditional encryption provide computaional integrity is reported to be 100 times greater than that of an aritmetiztionoriented design.

This report will focus on the proof system STARK which is invented by Starkware to ensure integrity and privacy in blockcahin computation. STARK offers the fastest proof in the class and provides a higher security scale based on fewer cryptographic assumptions. It is thought to be safe in post-quantum technology as well, and the fact that there is no need for a trusted setup as in SNARK[1] protects it from dangers.The security evaluation of STARK relies heavily on hash functions. However, hash functions using traditional symmetric encryptions, such as SHA2 or Rijndael, offer costly solutions for STARK. Therefore, hash functions for STARK need to be designed. For this, an agreement was signed between the Ethereum Foundation and Starkware.

In this report, some preliminary information will be given about the structure of STARK and hash functions in Section 2. Then, the Starkware process of STARK-friendly hash challenge will be discussed in Section 3. The MiMC [8] which existing before grant and the new designs which are GMiMC [9], HadesMiMC [3] and MARVELlous [10] will be reviewed, and Rescue wtih specific variant is concluded to be secure among the candidates reviewed in terms of complexity, efficiency, and safety at limited time. Afterwards, the design rationale of the MARVELlous family that Rescue is included in will be examined in Section 4. The algorithms which are Vision and Rescue will be introduced, and security evaluations will be reviewed in Section 5 and Section 6.

Chapter 2

Some Preliminary

2.1 Sponge Construction

The sponge construction is an iterative structure that can produce outputs of the desired length by means of a transformation or a permutation with inputs of varying lengths. This transformation or permutation can operate on b bits, and this is called the width of the state, shown in Figure 2.1. The state b consists of two components as r and c such that $b = c + r$ where c capacity of state and r is rate of state. The first thing to do in the algorithm is to make reversible padding according to the determined rule so that stretching the input string up to a multiple of r-bit. Then, this string splits into r-bit blocks so that they can be injected into the structure.In addition, the state b is initialized to zero.

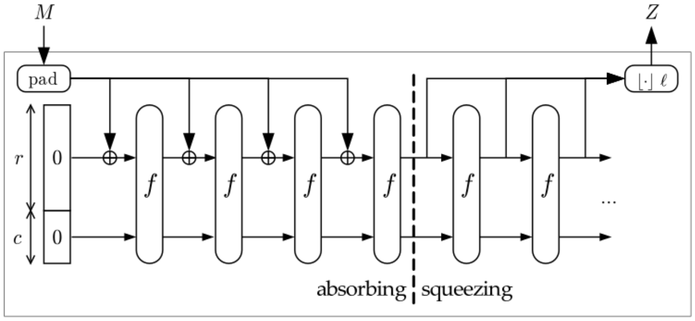

Figure 2.1: Sponge Construction

Sponge structure has absorbing and squeezing parts. The r-bit blocks are xored with the relevant bits of the state in the absorb phase and the updated state passess the f function. The absorbing phase continues until all r bits are included in the structure. It should be noted that the final output of the absorbing phase should not be zero. After absorbing phase, the squeezing phase is started, the first output block is the first r-bit of the state that switches to the squeezing phase. Then, the first r bits of outputs of the f functions are taken according to the desired output size.

2.2 ZK-STARK

STARK is a proof system. It uses the latest cryptography to provide polylogarithmic sources of verification and size of evidence with minimal and post-quantum secure assumptions.

A zero-knowledge proof is a protocol between a prover and a verifier. Instead of revealing the information explicitly, it gives information about the information and tries to convince the verifier that l is a member of the language $L \subset \{0,1\}*$. There are some conditions that a zero-knowledge proof must satisfy:

- **Completeness:** The proof can be accepted if the underlying claim is accurate and the prover and verifier are fair.
- **Soundness:** A dishonest prover cannot convince a faithful verifier into accepting a false claim as true.
- **Zero-knowledge:** Verifier should not know anything about information other than its validity. That is, a proof must be prepared that the verifier cannot make inferences about the hidden information.

If a zero-knowledge protocol is designed to provide the following conditions, it is called ZK-STARK[2] protocol.

- **Scalable:** It means that the proof of a knowledge in a short time and taking up little space..
- **Transparent:** The proof must be transparent. There is no need to trust anyone and trusted setup phase.
- **Argument of knowledge:** The prover must have solid evidence to convince the verifier that she knows the hidden information.

Perhaps the most important feature of ZK-STARK is its transparency. They do this with hash functions and by having random verifier confirm the proof. Since hash functions are used in the proof, it is considered to be post-quantum secure. Nowadays, decentralization is the main concern, so the lack of a trusted setup makes it preferable.

2.2.1 STARK Complexity

Definition: (Constraint systems). Let $\mathbb{F}$ be a finite field and w, t, d be integers. ($\mathbb{F}$,w,t,d)-constraint satify a multivariate polynomial C which is called constraint-polynomial with degree d by the variables $X[i, j]$ where $i \in \{1, 2, ..., t\}$ and $j \in \{1, 2, ..., t\}$

An ($\mathbb{F}$, w,t,d)-constraint system S is a set of ($\mathbb{F}$, w, t, d’)-constraints for d’ d, where d is the maximal degree of a constraint polynomial in constraint system S.

Definition: (Algebraic Execution Trace). The algebraic execution trace ($\mathbb{F}$,w,t)-(AET) is an array A consisting of t rows and w columns.Each entry of the array is an element of $\mathbb{F}_q$.

The i^{th} row in the array indicates the state at time i, and the j^{th} column represents the change of state at algebraic register over time. $A[i, j]$ refers to the entry in the i^{th} row and j^{th} column of AET array A. We say that A satisfies S if and only if for every constraint polynomial C in S satify

$$C(A[i_1, j_1], ..., A[i_k, j_k]) = 0.$$

Definition: (STARK complexity)Let $f : \mathbb{F}^n \to \mathbb{F}^m$ be a function. If the ($\mathbb{F}$, w, t, d) is a constraint system, The following conditions are satisfied:

- **i/o mapping:** There exist n input indices and m output indices such that $I_1, I_2, ..., I_n \in \{1, 2, ..., t\}$ and $O_1, O_2, ..., O_n \in \{1, 2, ..., w\}$. The AET of ($\mathbb{F}$, w, t, d) A shoul include the input and output of f in these locations.
- **Completeness:** For every input $x = (x_1, x_2, ..., x_n) \in \mathbb{F}^n$ and for every output $y = (y_1, y_2, ..., y_m) \in \mathbb{F}^m$, there exists a ($\mathbb{F}$, w, t, d)-AET A_x that satisfies the system S such that $\text{A}_x[I_i] = x_i$ for all $i \in \{1, 2, ..., n\}$ and $\text{A}_x[O_j] = y_j$ for all $j \in \{1, 2, ..., m\}$.
- **Efficiency:** There is a method that can generate the AET A_x for given $x \in \mathbb{F}^n$ where the above conditions are satisfied.
- **Soundness:** ($\mathbb{F}$, w, t, d)-AET A is a t$\times w$ array. We say that $x(A) = (A[I_1], A[I_2], ..., A[I_n])$ and $y(A) = (A[O_1], A[O_2], ..., A[O_n])$, if $y(A) \neq f(x(A))$then A does not satisfy the system S.

Now, we can defined the STARK complexity as

$$c(S) = \lceil log_2|\mathbb{F}/64|\rceil^2 \cdot (d + w) \cdot tlog_2t$$

and the STARK complexity of f is shown as c(f) and the minimal complexity of f is found by taking into account all system S implements f.

Chapter 3

STARK-friendly Hash Challenge

In 2018, the Ethereum Foundation gave Starkware a grant to choose a STARK-friendly open-source hash function. The project took 2 years and Starkware begins analysis of the candidates which is consists of the MiMC, GMiMC, HadesMiMC and MARVELlous families. Algebraic, symmetric cryptanalysis studies of these candidates are carried out, after that a public attack challenge is arranged and it is decided to standardize $Rescue_{122}$ from the MARVELlous family. The process of two years as follows:

July 2, 2018: Signing the agreement between StarkWare and the Ethereum Foundation.

Q3/2018: An agreement is signed with The Computer Security and Industrial Cryptography (COSIC) group to evaluate existing SFH candidates and it is concluded that none of them, except MiMC[8], are suitable for the appropriate STARK complexity. As a result, the Ethereum Foundation and Starkware decide to collect new hash candidates.

Q3/2018-Q2/2019: Among the new candidates were the pre-existing MiMC and the GMİMC which was generated independently from this process, and the MARVELlous[4] and HadesMiMC families[3], which were able to receive grants from this project to develop.

Q4/2018-Q4/2019: Starkware has contracted with the CryptoNext Security (CNS) company to conduct algebraic cryptanalysis evaluations of these SFH candidates.

Q3/2019-Q1/2020: The candidate list became exact in Q2/2019 which is consists of MiMC, GMiMC, HadesMiMC and MARVELlous. In these all structures, the same parameters were chosen to facilitate comparisons and were offered for attacks by the public. There were attacks working on some instantiations with a 45-bit security level, but no results above the 80-bit security. On the other hand, a committee convened for symmetric cryptanalysis studies and published the this report[5]. The MARVELlous family was found safe; however, a clear conclusion could not be reached for the CICO problem of Rescue and the targeted attacks for Rescue.

Q2/2020: Two groups of cryptographer have investigated these problems about Rescue and both groups have submitted reports[6]. [7] that Rescue is secure. Starkware recommended the Eherium Foundation as a SFH function.

All SFH functions mentioned are instantiationed according to their parameters such as security level, operation fields, rate and capacity of sponge construction. The parameters of the STARK-friendly Hash Functions in the candidate list have been customized and all cryptanalytic studies have been done on this specific version. The parameters were fixed to values shown in Table 3.1 [7] so that all candidates had a security level of about 120.

	Structure type	Field	# of Rounds	Security level (approximately)
MiMC	Fiestel Structure	prime field with $p = 2^{253} + 2^{199} + 1$	320	126
Starkad	SPN structure	binary field with 2^{63}	8 full / 43 partial	126
Poseidon		prime field with $p = 2^{61} + 20.2^{32} + 1$	8 full / 40 partial	122
Vision		binary field with 2^{63}	10	126
Rescue		prime field with $p = 2^{61} + 20.2^{32} + 1$	10	122
GMiMC		prime field with $p = 2^{61} + 20.2^{32} + 1$	101	122

Table 3.1: Specification of SFH candidates parameters

All constructions have sponge construction with 12 elements of fields and the rate of sponge is 8 elements and the capacity is 4; except $MiMC_{126}$ which has only 2 field elements in state. These versions are represented as follows according to their security level; $MiMC_{126}$, $Starkad_{126}$, $Poseidon_{122}$, $Vision_{126}$, $Rescue_{122}$, $GMiMC_{122}$.

In security assessments, GMiMC and HadesMiMc have security marjin against some types of attacks such as differential attack, integral attack and advanced algebraic attack; however, the sponge construction version represent some weaknesses in collision attack when applied to the GMiMC. Similarly, some cases of HadesMiMC family are weak against preimage attack[11]. In MiMC, attacks can be made to the block cipher version, but this attack cannot be effective in $MiMC_{126}$ sponge construction. A second attack can be implemented on the binary field, but in our specific instantiation of $MiMC_{126}$ is still considered secure since large prime is used.[12]. Any weaknesses were not found in the cryptanalytic evaluations of $Vision_{126}$ and $Rescue_{122}$ algorithms, so the absence of any security vulnerability about these two ciphers means that it is still secure.We focus on MARVELlous family in this report, so more detailed information about Vision and Rescue will be given in the next sections. As a result of all these evaluations, MiMC and MARVELlous family seem to be secure candidates for SFH.

Although arithmetic operations are easier to do in binary fields and bit-strings can be represented more easily, prime fields present difficulties in arithmetic operations but they have advantages for Stark construction. Therefore, $MiMC_{122}$ and $Rescue_{122}$ on prime field are preferred instead of $Vision_{126}$ on binary field. Finally, the fact that a state of MiMC consists of only 2 field elements is seen as disadvantageous against all other candidates.

3.1 STARK Complexity of Candidates

The t and w values in the Table 3.2 are optimized selections for both prover complexity and verifier complexity. Because reducing the complexity of the verifier results in an increased complexity of the prover. It is not possible to precisely calculate the STARK complexity of non-trivial functions. Therefore, the upper bound of STARK complexity is taken into account. It is also stated that changing the finite field will affect the complexity considerably. The values listed in the Table 3.2 [7] are practical examples. The complexity values given in the Table 3.2 do not take into account the difficulty of creating algebraic execution traces. Because it has similar completixy level for all algorithms.

The Table 3.2 shows the STARK complexity of the STARK-friendly Hash functions. The second column shows the finite fields which they are working on. In column 3, it specifies the length of the algebraic execution trace to be created in one call. t and w are the optimized row and column numbers of the AET as is known. The last column shows the STARK complexity of these algorithms. $Rescue_{122}$, which is the most suitable for the STARK in terms of complexity.

Name	**F**	**t / invocation**	**w**	***d***	**c / 10^5 invocations**
$MiMC_{126}$	253-bit prime	320	2	3	6.4×10^{10}
$GMiMC_{122}$	61-bit prime	101	1	3	9.4×10^8
$Starkad_{126}$	GF(2^{63})	10	14	3	3.4×10^8
$Poseidon_{122}$	61-bit prime	8	17	3	3.1×10^8
$Rescue_{122}$	61-bit prime	10	12	3	3×10^8
$Vision_{126}$	2*GF(2^{63})	20	12	6	7.5×10^8
		40	12	4	1.4×10^9

Table 3.2: STARK complexity of SFH Candidates

3.2 Efficiency of Candidates as Hash Function

Of course, the speed and efficiency of the algorithm is also important in the selection of the hash function. But efficiency is secondary when it comes to STARK complexity and security. The efficiency of the candidate algorithms with a 64-bit CPU are given in the Table 3.3 [7].

SFH	**log |F|**	**Rounds R**	**State m**	**# 64-bit multiplications**	**Total**
$Rescue_{122}$	61	10	12	$(2m^2 + 70m) \times R$	11280
$MiMC_{126}$	256	320	2	$16 \times 2 \times R$	10240
$Vision_{126}$	63	10	12	$(2m2 + 8m + 2(log\|F\| + +3m)) \times R$	5820
$Starkad_{126}$	63	$R_f = 8, R_p = 43$	12	$(m^2 + 2m) \times R_f + (2m + 2) \times R_p$	2462
$Poseidon_{122}$	61	$R_f = 8, R_p = 40$	12	$(m^2 + 2m) \times R_f + (2m + 2) \times R_p$	2384
$GMiMC_{122}$	61	101	12	$2 \times R$	202

Table 3.3: Efficiency of SFH Candidates

$Rescue_{122}$: In $Rescue_{122}$, which consists of 2 steps, there is a cubing map in the first step, it requires 2 multiplications. In the second step, there is a root-cubing map and it can be obtained with 68 multiplications on the finite field. The cost of matrix multiplication in both steps is $2m^2$ in total.

$MiMC_{126}$: A multiplication operation in 256-bit space is approximately 16 operations in 64-bit space, since it is over a larger field than the others.

$Vision_{126}$:consists of 2 steps. The first step is inversion mapping and the cost of calculating it is $(log|F|+3)$ multiplications. Then, the 4^th order polynomial transformation requires 4 multiplications (in the reverse case, 8 multiplications in total). The cost of matrix multiplication in both steps is $2m^2$ in total.

$Starkad_{126}$ and $Poseidon_{122}$: There are full rounds R_f and partial rounds R_p of Starkad and Poseidon. In full rounds, there are cube operations with a cost of 2 multiplications. There are m state elements. In addition, linear transformation costs m^2 with m state elements. There is only one cube operation in the partial round and it is summed by linear combinations of other state elements. There are $2m+2$ multiplication operations in the partial round.

$GMiMC_{122}$: Each cycle has 1 cube operation cost 2 multiplications.

Chapter 4

General Structure of MARVELlous Family

MARVELlous family is a block cipher in SPN network structure with the following parameters:

q – **either a power of 2 or a prime**;
m – **dimension of a state vector in** ${\mathbb{F}_q}^m$;
$\pi = (\pi_0, \pi_1)$ – **S-boxes**;
M – **MDS matrix**;
v – **first step constant**;
s – **security level of algorithm**

Some precaution is taken into account in the selection of parameters for security reasons:

- $m > 1$ and $log_2 q > 4$
- $\pi = (\pi_0, \pi_1)$ π_0 has a high degree and π_1 has low degree in encryption direction; vice versa in decryption direction.
- Make sure that the MDS matrix exists and $2m \leq q$

4.1 Primitives

The State : is a vector consisting of m field elements such that $x_0, x_1, ..., x_{m-1}$where m> 1. The most important factor that determines the cost in such algorithms is the difficulty of multiplication; however, it is thought that fast diffusion will make a great contribution in this algorithm and the matrix is added to the design. In fact, the size of the matrix and the scalar products do not affect the cost. This does not thwart the effectiveness of the design too much.

Round Function: a state and a subkey passes the round function as inputs and the function produces the next state as output. The input in the first round is palintext and masterkey, and the output of the last round is ciphertext. The MARVELlous structure consists of repeating this operation N times. In MARVELlous, each round consists of 2 steps and these steps include S-boxes, Matrix multiplication and key addition phase,respectively. There is also a keyless version of the structure. In this case, the step constant is derived by the key schedule instead of the subkey.

S-boxes: The MARVELlous design uses two different S-boxes. Even number steps of the design, the π_0 S-box used with high-degree in encryption direction and low-degree in decryption direction. On the other hand, the π_1 S-box used in odd number of steps is designed in the opposite aimful, i.e, it has a low degree of encryption and a high degree in decryption. This ensures that encryption and decryption have the same cost and also ensures that attacks mounted in both directions have the same security level. Using power maps in S-boxes gives the algorithm some cryptanalytic features and robustness against attacks[14]. There are two members of the MARVELlous family. In the Vision design, S-box of this design consists of the binary field inversion map combined with a linearized affine polynomial for some efficiency reasons. The polynomial in the form:

$$B(x) = b_{-1} + \sum_{i=0}^{n-1} b_i x^{2^i} \in \mathbb{F}_{2^n}[x]$$

After all, Vision S-boxes take the forms such that

$$\pi_0 = B^{-1}(x^{-1}) \quad \text{and} \quad \pi_1 = B(x^{-1})$$

The S-boxes selected on the prime field in Rescue are as follows:

$$\pi_0 = x^{\alpha} \text{ and } \pi_1 = x^{\frac{1}{\alpha}}$$

MDS Matrix: is the linear layer in the MARVELlous design and allows diffusion to occur. Due to SPN structure with an arithmetic oriented design, you cannot change the bits indivual. Therefore MDS matrix offers good solution. There is no restriction on the choice of MDS matrix because there is no factor affecting efficiency or cost. However, the $2m \leq q$ condition mentioned above must be satisfied to guarantee the existence of the matrix. In this design, an $m \times 2m$ Vandermonde matrix is presented as the MDS matrix, generated by the powers of a primitive element of the field. Then, when an $m \times m$ identity matrix is formed in the echelon form of this matrix, MDS matrix is obtained. This matrix propagation speed offered is very good and full diffusion can be achieved in just one round.

Round Constant: Including the round constant in the design is a precaution against malicious threats because it breaks symmetry in a symmetric structure and disconnects the next state from the previous one. SHAKE-256 is used in the design to derive the first constant. The important point in choosing a round constant is that the first value must be in $\mathbb{F}_q$, not in any subfield of $\mathbb{F}_q$. Then, these round constants are derived for each round by applying the trasnformation to the previous round one. The state-wide constant of the design is used in the key schedule, so in the keyless version of algortihm, a value comes out of the key schedule to be injected into the state.

Key Schedule: The round function is used in key generation. Just like the master key plaintext, the round constant is added state where the key should be added in the round function. In keyless version, masterkey is considered as zero and round constant is added where it is needed. Then, it is used as a fresh value where it should be injected into the algorithm. In contrast to the easy key schemes used in lightweight encryptions, a more difficult key generation scheme is used here. Due to the specified security measures in [10]

Number of Round: The number of rounds is decided in the light of cryptanalytic evaluations. In particular, the number of rounds that considered secure against Gröbner-based attack is paid attention because it is greater than the number of rounds that make it secure against other attacks. Finally, It is decided that the design will be 10 rounds which is more than two times the number of these rounds.

Chapter 5

Vision

Vision perfoms on binary fields F_{2^q}. The state is a column vector which consists of m elements of ${F_{2^q}}^m$. F_2- linearized affine polynomial of degree 4 which is

$$B(x) = b_0 + b_1.x + b_2.x^2 + b_3.x^4$$

In S-boxes structure s.t. ;

$$\pi_0 : \mathbb{F}_{2^n} \rightarrow \mathbb{F}_{2^n} : x \rightarrow B^{-1}(x^{-1})$$
$$\pi_1 : \mathbb{F}_{2^n} \rightarrow \mathbb{F}_{2^n} : x \rightarrow B(x^{-1});$$

In Vision cipher, the round function is iteratively applied to the plaintext N times and ciphertext is obtained. As seen in the Figure5.1, the round function consists of two steps. The key is injected before the first round, after the last round, and between the each two steps in the round function. The round function is also used for key generation, the step constant acts as a subkey in keyless versin. The keyless use of Vision has sponge structure.

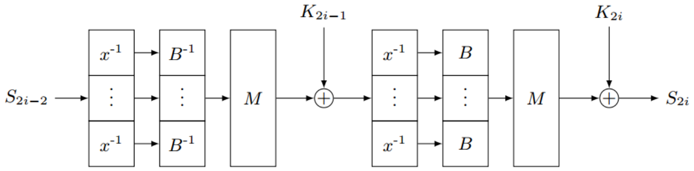

Figure 5.1: One round of Vision

To decide the number of rounds, the number of rounds r_0 that are considered secure against Gröbner-based attacks and the number of rounds r_1 that provides enough security marjin against other attacks, because Gröbner-based attacks are considered to be more efficient with respect to Vision. Then the number of rounds is calculated as $N = 2.max\{r_0, r_1, 5\}$ with sanity number 5. It is desired to be sure of its security by doubling the maximum number.

5.1 Security Analysis of Vision

5.1.1 Statistical Attacks

There are two factors that determine the security level of an algorithm in SPN structure against statistical attacks: the first is the number of active S-boxes, and the other is the propagation probability of an S-box. The large number of active S-boxes and the small propagation probability of each acitve S-boxes make the algorithm secure against these attacks. Using the number of active S-boxes and the propagation probability, a characteristic for the attack is generated. Looking at the S-box used in Vision, a power map that has good linear and differential properties is used. As the size of the finite fields $\mathbb{F}_{2^q}$ increases, the diffrential and linear uniformity decreases. Multiplication with the MDS matrix in each step ensures that at least $m+1$ S-boxes are activated in one round. Moreover the differential propagation probabilities of these activate S-boxes over the binary field are calculated as:

$$\delta = 2^{-log_2(q)+2}$$

Then the differential characteristic for N rounds becomes:

$$2^{N(m+1)(-log_2(q)+2)}$$

Using the security level parameter s, the results when the number of safe rounds N_s is calculated in which the best differential characteristic can be 2^{-2s}.This means ensuring that the Vision is secure against differential attack.

$$N_s = \frac{2s}{log_2(q^{m+1}) - 2(m+1)}$$

Similarly, linear characteristic has its correlation as stated in [10]

$$|\lambda| = 2^{\rceil log_2(q)/2\rceil+1}$$

and characteristic is written as follows:

$$2^{N(m+1)(log_4(q)+2)}$$

Like in the differential cryptanalysis, if we take the best characteristic probability as 2^{-2s}, the number of rounds is calculated as:

$$N_s = \frac{s}{log_4(q^{m+1}) - 2(m+1)}$$

5.1.2 Structural Attacks

Invariant subfield attack For the invariant subfield attack, two subfields of the binary field $\mathbb{F}_{2^q}$ are used s.t $F_{q_1} \subset \mathbb{F}_{2^q}$, $F_{q_2} \subset \mathbb{F}_{2^q}$.

When take an element $x \in \mathbb{F}_{q_1}$ as an input, the output $y \in \mathbb{F}_{q_2}$. These elements may be in the same subfields,i.e., $F_{q_1} = \mathbb{F}_{q_2}$. The attacker probably prefers this way for attacking. As a precaution, mentioned earlier, the coefficients of linearized-affine polynomial and step constants are not in any subfield of $\mathbb{F}_{2^q}$.

Self-similarity attack allows to attack minimized algorithms by splitting the algorithm into sub-algorithms. The MARVELlous family is not suitable for this cryptanalytic method. Because the step constant injected into the each round destroys the similarity between the sub-algorithms.

Higher-order differential cryptanalysis is effective in design which has a low algebraic degree cipher structure. MARVELlous design owes its resistance against this attack to the S-boxes. Thanks to the $B(x)$ and $B^{-1}(x)$ polynomials used, at the end of a round, algebraic degree becomes $n-1$, which is the largest polynomial degree in binary field $\mathbb{F}_{2^n}$.
Assuming the algorithm is N rounds, N rounds end up with algebraic degree $(n-1)^N$. The security parameter is expected to be smaller than $(n-1)^N$. The number of rounds N_s to be secure against the attack:

$$N_s = \frac{log_2 s}{log_2(n-1)}$$

Interpolation attack is a type of attack that takes advantage of the low degree in the algorithm. It creates a polynomial description of the input and output pairs in the algorithm, thanks to Lagrange interpolation. But the linearity of the Lagrange interpolation complicates the task at high degrees. In Vision,S-box is a component that increases degree quickly thanks to inversion mapping in the S-box. Therefore, the interpolation attack does not pose much of a danger to Vision. As stated in [10], 3 round Vision interpolation is resistant for the attack.

5.1.3 Algebraic Attacks

Gröbner Basis Attack The cryptanalysis technique that should be emphasized for arithmetic-oriented cipher Vision is the Gröbner basis attack. Because the efficiency of the algorithm and the complexity of attack are based on the multiplication complexity. When it is desired to make the algorithm more efficient, Gröbner basis attack is made available for the cipher. The following steps proceed to mount Gröbner basis attacks:

1. An equation system is set up and the Gröbner basis is computed in the order of degrevlex, this process is done with algorithms such as Buchberger's, F4, F5 or Macaulay matrices.
2. The term order of the constructed Gröbner basis is changed with the algorithm such as FGLM or Gröbner Walk algorithm that is, it is passed from degrevlex order to lex order.
3. The unvariate polynomial is factored and all its roots are found by back-substitution technique.

Considering the methods given above, the number of safe rounds based on the difficulty of finding a Gröbner basis is calculated on the principle stated in [10]. As a result, if the first step is not completed, the attack is out of question. What we need to specify here is the complexity of the calculation on the Gröbner basis[13].

Deciding the degree of regularity is a difficult problem. It is calculated as follows for systems considered to be regular. Regular system means that the number of polynomials is equal to the number of variables, i.e., k=n

When we look at Vision, the degree of regularity is found as

$$d_{reg} = 1 + 8mN - m$$

where m is number of state elements, N is round number.

In the calculation made assuming that the system is regular. It is also worth noting that the system is modeled by $2mN$ equations in $2mN$ variables. In addition to this, in the experimental attack, a single data point is detected in the experiment that runs for more than 60 hours. This data shows that the regularity degree is 9 for m=2, N=2. [10]

As a result of this, it is determined that there is a correlation between the concrete degree and the regularity degree such that $\frac{d_{reg}}{4} \leq d_{con}$. Finally, the complexity of Gröbner basis attack for Vision is computed as follow [10]:

$$\binom{(1+16mN-m)/4}{2mN}^2$$

where w=2.
The safe number of round for Vision Gröbner basis is calculated to be resistant to attack like that:

$$N_s = min(N) \text{ related to } \binom{(1+16mN-m)/4}{2mN}^2 \geq 2^s$$

Chapter 6

Rescue

Rescue and Vision are similar, but the Rescue operates over $\mathbb{F}_q$ with q is prime number not power of 2. In addition, there is a difference in the S-boxes of these two designs. As in Vision, a column vector in $\mathbb{F}_q^m$ with m elements is taken as state. S-boxes of Rescue use power mapping and inverse of power mapping, shown as:

$$\pi_0 : \mathbb{F}_q \to \mathbb{F}_q : x \to x^{\frac{1}{\alpha}} \tag{6.1}$$

$$\pi_1 : \mathbb{F}_q \to \mathbb{F}_q : x \to x^{\alpha} \tag{6.2}$$

A small prime is selected for α in encryption direction such that $gcd(q-1, \alpha) = 1$. When power mapping is reversed in other step S-box, the power grows rapidly.
For example, taking $\alpha = 3$ is an effective parameter choice for the design. Because inverse map becomes $x \to x^{\frac{2p-1}{3}}$. The process of deriving the ciphertext from the plaintext is the same as described in Vision. The design is also clearly explained in Figure 6.1. In keyless usage, sponge structure is used as described in Vision design. The number of rounds with the desired security level is decided in a similar way. The number of rounds r_0 that are considered secure against Gröbner-based attacks and the number of rounds r_1 that provides enough security marjin against other attacks, because Gröbner-based attacks are considered to be more efficient with respect to Vision. Then the number of rounds is calculated as $N = 2.max\{r_0, r_1, 5\}$ with sanity number 5. It is desired to be sure of its security by doubling the maximum number.

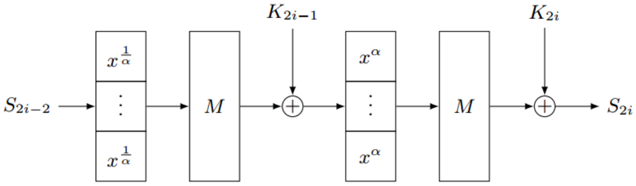

Figure 6.1: One round of Rescue

6.1 Security Analysis of Rescue

6.1.1 Statistical Attacks

Similar to Vision, we are dealing with active S-box numbers and propagation probability of statistical property of an S-box. There is no difference in the number of active S-boxes. At least m+1 S-boxes are activated in one round. Since the S-boxes operates over prime field $\mathbb{F}_q$, there are some differences in propagation probability evaluation. It is known from that the S-boxes of Rescue, which is a power map, are $(\alpha - 1)$-uniform, and the difference propagation probability is found in [10]

$$\delta = 2^{log_2(q)+log_2(\alpha-1)}$$

As a result of these operations, the best N-round differential characteristic is

$$2^{N(m+1)(log2(q)+log2(\alpha-1))}$$

Considering the differential caharacteristic as in Vision, the best 2^{-2s} with security parameter s, round number becomes

$$\frac{2s}{log_2(q^{m+1})log_2((\alpha-1)^{m+1})}$$

As for the linear cryptanalysis part, stated that it is very complex on prime field in [10]. It seems that linear cryptanalysis is adapted to work on binary fields and has no adaptation for prime fields. There is no definite conclusion as to whether linear attack can be applied to prime fields. This is open question.

6.1.2 Structural Attacks

Invariant subfield attack is not an attack that can be applied in prime field. Because the field must have two subfields which are the same or different from each other. But there are no non-nontirivial subfields in the prime fields. Therefore, the attack is not a valid attack for Rescue.

Self-similarity attack As we mentioned in the Vision, sub-algorithms that will be divided from the main algorithms are used for the attack. The step constant that destroys self-similarity also make invalid this attack on Rescue.

Interpolation attack Similar to Vision part, it is an attack that can be effective in low-degree algorithms. Therefore, interpolation attack is unlikely to be effective, because Rescue can recah high-degree quickly by the S-box. Again [10] as said 3 rounds Rescue has the sufficient security marjin.

These attacks were applied in different instantiations of reduced-round Rescue. Since the number of variables and the number of equality depend on r_q, changing r_q changes the necessity. So r_q is fixed to 1 which is best case for attacker.

6.1.3 Algebraic Attacks

Gröbner Basis Attack In Rescue, in the light of the same principle, the necessity of calculating the Gröbner basis is calculated [10] in order to be resistant to attacks. First, an equation system is created that encodes the Rescue preview. It is also worth noting that the system is modeled by $2mN + r_q$ equations in $2mN + r_q$ variables.Then the degree of regularity of this system is calculated and the result is

$$d_{reg} = 1 + (mN + rq)(\alpha - 1)$$

As in Vision, the results of an experimental attack are as follows:

$$\begin{aligned}
&\text{For } m = 2, \alpha = 3 \rightarrow \{(N, d_{con}) | (2, 6), (3, 8), (4, 10)\},\\
&\text{For } m = 3, \alpha = 3 \rightarrow \{(N, d_{con}) | (2, 8)\}\\
&\text{For } m = 2, \alpha = 5 \rightarrow \{(N, dcon) | (2, 10), (3, 14)\}.
\end{aligned}$$

It is concluded that $d_{con} = 0,5mN(\alpha - 1) + 2$ so that these points are obtained.

Here, it is observed that the correlation between the concrete degree value and the theoretical regularity degree is that $\frac{d_{reg}}{2} \leq d_{con}$. Finally, the complexity of Gröbner basis computation for Rescue is about

$$\binom{mN(0.5(\alpha - 1) + 1) + r_q + 2}{mN + r_q}^2$$

where w=2.

The safe number of round for Rescue Gröbner basis is calculated to be resistant to attack is calculated as:

$$N_s = min(N) \text{ related to } \binom{mN(0.5(\alpha - 1) + 1) + r_q + 2}{mN + r_q}^2 \geq 2^s$$

Bibliography

[1] Blockchain Council, 2022, *"Know The Difference Between Zk-SNARKS Vs. Zk-STARKS"*.

[2] Team, StarkWare, 2021, "ethSTARK Documentation", IACR Cryptol. ePrint Arch., 582.

[3] Grassi L, Khovratovich D., Rechberger C., Roy A., Schofnegger M., 2021, *"Poseidon: A New Hash Function for {Zero-Knowledge} Proof Systems"*, 30th USENIX Security Symposium (USENIX Security 21), 519-535.

[4] Ashur T., Dhooghe S., 2018, *"MARVELlous: a STARK-friendly family of cryptographic primitives"*, Cryptology ePrint Archive.

[5] Canteaut A., Beyne T., Dinur I., Eichlseder M., Leander G., Leurent G., Naya-Plasencia M., Perrin L., Sasaki Y., Todo Y., others, 2020, *"Report on the Security of STARK-friendly Hash Functions (Version 2.0)"*.

[6] Beyne T., Canteaut A., Leander G., Naya-Plasencia M., Perrin L., Wiemer F., 2020, ""On the security of the Rescue hash function", Cryptology ePrint Archive.

[7] Ben-Sasson E., Goldberg L., Levit D., 2020, *"Stark friendly hash–survey and recommendation"*, Cryptology ePrint Archive.

[8] Albrecht M., Grassi L., Rechberger C., Roy A., Tiessen T., 2016, *"MiMC: Efficient encryption and cryptographic hashing with minimal multiplicative complexity"*, Advances in Cryptology–ASIACRYPT 2016: 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I, Springer, 191-219.

[9] Bonnetain X., 2019, *"Collisions on feistel-MiMC and univariate GMiMC"*.

[10] Aly A., Ashur T., Ben-Sasson E., Dhooghe S., Szepieniec A., 2019, *"Efficient Symmetric Primitives for Advanced Cryptographic Protocols (A Marvellous Contribution)."*

[11] Keller N., Rosemarin A., 2021, *"Mind the middle layer: The HADES design strategy revisited"*, Advances in Cryptology–EUROCRYPT 2021: 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17–21, 2021, Proceedings, Part II, Springer, 35-63.

[12] Eichlseder M., Grassi L., Lüftenegger R., Øygarden M., Rechberger C., Schofnegger M., Wang Q., 2020, *"An algebraic attack on ciphers with low-degree round functions: application to full MiMC"*, Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7–11, 2020, Proceedings, Part I 26, Springer, 477-506.

[13] Bardet M., Faugere J., Salvy B., 1963, *"On the complexity of Gröbner basis computation of semi-regular overdetermined algebraic equations"*, Proceedings of the International Conference on Polynomial System Solving, 71–74.

[14] Nyberg K., 1994, *"Differentially uniform mappings for cryptography"*, Advances in Cryptology—EUROCRYPT'93: Workshop on the Theory and Application of Cryptographic Techniques Lofthus, Norway, May 23–27, 1993 Proceedings 12, Springer, 55-64.