Format Preserving Encryption

Andreas B. Kidmose and Tine Jarman Topsøe

DTU Compute, Technical University of Denmark
Bygning 324, DK-2800 Kgs. Lyngby, Denmark
{abki,tjto}@dtu.dk

Abstract. Format Preserving Encryption (FPE) encrypts data into an output, that has the same format as the input. For example encrypting a credit card number into a credit card number. FPE is in particular applicable when encrypting existing databases with fixed record formats.
In this survey, we give an overview of FPE, with the latest developments in the area. We give a description of some general techniques for FPE. In particular, the two NIST recommended FPE schemes, FF1 and FF3, are described and an overview is given of their security and the latest published attacks.

Keywords: Format Preserving Encryption, Feistel Network

1 Introduction

Format Preserving Encryption (FPE) is, as the name indicates, a scheme that encrypts data in such a way that the output is in the same format as the input. A format here refers to a finite set of characters, along with a length. The input could for example be 16-digit decimal numbers, such as credit card numbers (CCN), or 9-digit decimal numbers, such as social security numbers (SSN).
More formally, let the FPE mapping be defined as $E : K \times M \to M$, where K is the secret key space and M is an arbitrary domain.

Today most standard block ciphers encrypts binary strings with fixed block length n. That is, a block cipher maps $E : K \times \{0,1\}^n \to \{0,1\}^n$, and typical block lengths are $n = 64$ bits or 128 bits. Hence, a standard block cipher is actually a FPE scheme, however limited to binary strings of certain lengths.

Usually, encrypting any message with a standard block cipher works fine; encrypting a message m (with arbitrary format) using a block cipher of a larger binary domain $\{0,1\}^n$, often includes encoding m to a binary string and using a padding rule to obtain a n-bit binary input string.
As standard block ciphers clearly handles the most useful cases in practice, why do we need FPE schemes for *arbitrary* domains?
Suppose we want to encrypt fields in an existing database with a certain record format. This could for example be 16-digit CCNs. Encrypting the CCNs with a standard block cipher means changing the record format to a larger and binary field. However, changing the record format in the database might be inconvenient and costly. Therefore,

using an encryption scheme that preserves the format, will add security to the database without having to reconstruct it.

So although standard block ciphers encrypt blocks of a fixed format, there are many other applications that requires encrypting data in other formats. Therefore dedicated FPE schemes for arbitrary domains have been designed.
One can of course design an FPE scheme from scratch, but it is difficult and it takes a long time to get it acknowledged by the cryptographic community. Therefore, it is more practical to build FPE schemes based on standardized block ciphers or hash functions, such as the block cipher AES [DR02] or the hash function SHA-3 [BJPA11].

The purpose of this survey, is to give an overview of FPE, with the latest developments in the area. We will in particular have focus on the two schemes, FF1 and FF3, recommended by the National Institute of Standards and Technology (NIST) in 2016 [NIS16].

In section 2, a brief historic outline of FPE is given. In section 3 a short description of general techniques for achieving FPE is given, followed by a description of the two NIST recommendations, FF1 and FF3, in section [ref]... At last,...

2 Brief History

Brightwell and Smith [BS97] introduced the first FPE scheme in 1997, and the scheme was proposed as a solution to data warehouse security. They called it *Data-type preserving encryption*. They suggested to build their scheme on well-known primitives like DES[ref] and IDEA[ref].

In 2002, Black and Rogaway [BR02] wrote a paper on FPE to draw attention to the area, and presented three general solutions to FPE, together with their analysis.

The name *format preserving encryption* was due to Terrence Spies from Voltage Security who wrote a paper [Spi08] on the matter in 2008.

In 2010, Bellare, Rogaway and Spies [BRS10] published a scheme for FPE named FFX. In the same year, Bries, Peyrin and Stern [BPS10] also published a scheme for FPE named BPS. The two proposed schemes were very similar, but both independently submitted to the NIST standardization process.

NIST specified three methods for FPE in Draft NIST SP 800-38G [NIS13], which was released for public comment in 2013. The methods where specified by the abbreviations FF1, FF2 and FF3, but submitted to NIST under the names FFX[radix], VAES3 and BPS-BC, respectively. The second scheme FF2 was submitted to NIST by Vance [ref] in 2011. However, FF2 was later withdrawn due to a comment from the National Security Agency (NSA), which resulted in the attack by Dworkin and Perlner [DP15] in 2015.

Finally in 2016, NIST published two recommendations for FPE in NIST SP 800-38G [NIS16], namely FF1 and FF3. Both FPE schemes are Feistel constructions, with slightly different round functions, but both derived from the standardized 128-bit block

cipher AES.

Even though FPE schemes date many years back, it is first in the recent years that the demand has increased and created an active area of research.

3 General Techniques

Designing FPE schemes for arbitrary domains are not straight forward, and various of techniques have been suggested in the last two decades.

Black and Rogaway suggested in a paper from 2002 [BR02] three different and fairly simple techniques for FPE, called *Prefix Cipher*, *Cycle-Walking Cipher* and *Generalized-Feistel Cipher*. The *shuffling cards* technique has also proven to be effective for FPE, where e.g. the *Thorp Shuffle* has provable good bounds. More dedicated techniques has also been suggested such as the VIL by Bellare and Rogaway [BR99].

A short description of the above mentioned techniques will be described in the following.

3.1 Prefix Cipher

A simple way of creating an FPE algorithm on the domain $M = \{m_0, m_1, m_2, \ldots, m_{N-1}\}$ is to assign a weight w_i to each element m_i for $0 \leq i < N$, and then use the relative ordering of the weights to determine the permutation. For example, sort the weights in ascending order, say $w_2 < w_{N-1} < w_0 < \cdots < w_1$, such that $E(m_2) = m_0$, $E(m_{N-1}) = m_1$, $E(m_0) = m_2$, ... , $E(m_1) = m_{N-1}$.

The weights can be assigned the elements by applying an already existing block cipher. For example, let M be the domain and let the weights be defined by applying the AES block cipher with a 128-bit secret key K on each element, such that $w_0 = AES_K(m_0), w_1 = AES_K(m_1), \ldots, w_{N-1} = AES_K(m_{N-1})$.

This technique was called *Prefix Cipher* by Black and Rogaway [BR02]. The technique is as secure as the block cipher used. Hence, using AES-128 as the applied block cipher offers the technique a 128-bit security.

Notice that the technique requires N calls to the applied block cipher in the initialization step, and requires storing a table of sensitive data. In other words, in order to encrypt one element you need encrypt all the elements. The technique is therefore only suitable for small domains.

3.2 Cycle Walking

Another technique suggested by Black and Rogaway [BR02] is the *Cycle-Walking Cipher*.

Let M be the domain, and suppose we know a FPE scheme that works on a superset of M. Denote this FPE scheme by $\tilde{E}_K(\cdot)$ working on $T \supseteq M$. Then construct a cipher $E_K(\cdot)$ on the set M, by computing $t = \tilde{E}_K(m)$ and iterating until $t \in M$.

For example, let $M = \{0, 1, \ldots, 10^9 - 1\}$ be the set of all numbers up to 9-digits, and let $\tilde{E}_K(\cdot)$ be a 30-bit block cipher on the set $T = \{0, 1, \ldots, 2^{30} - 1\}$, such that T is a strict superset of M, $T \supset M$. Suppose we want to encrypt the 9-digit number $m = 123456789$, then we compute $t_1 = \tilde{E}_K(123456789) = 1012345678$, which is a 10-digit number in the set T, but clearly $t_1 \notin M$. We then simply re-encrypt by computing $t_2 = \tilde{E}_K(t_1) = \tilde{E}_K(1012345678) = 123456$, and since $t_2 \in M$ we output 123456 as $E_K(123456789)$. In this example the expected number of repetitions is 1.074, which does not seem too bad.

Clearly the expected number of repetitions cannot be too large, so encrypting 9-digit numbers with a 128-bit block cipher is surely a bad idea. Also notice that the duration is not deterministic.

Black and Rogaway [BR02] show that $E_K(\cdot)$ is a well defined permutation on M, and they further proof that the security of the cipher relies on the underlying FPE scheme.

3.3 Generalized-Feistel Cipher

The Feistel network is a well known method for constructing block ciphers. The general idea is to split the input into two parts, input one part into a round function and add the output to the other part, and finally swap the two parts. This is repeated for a number of times. In Figure 1, a two-round Feistel Network is shown.

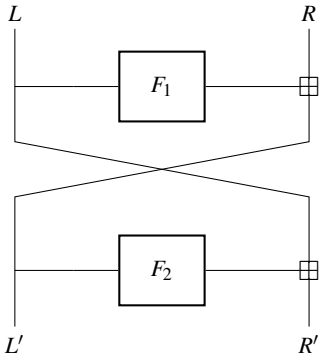

Fig. 1: A two-round Feistel Network

Black and Rogaway provided a security bound on the Feistel network in [BR02]. Later, in particular Jacques Patarin has led the way in providing security proofs [Pat03, Pat04] and generic attacks [Pat01].
FPE schemes built on Feistel networks has become a natural choice due to its history. Feistel networks have indeed been extensively studied by the cryptographic community for a long time, and is an acknowledge method for constructing block ciphers.

Furthermore, whereas Prefix Cipher is only suitable for small domains, and Cycle-Walking is only suitable for domains slightly smaller than the size of the used block cipher, Feistel networks have the advantage that they can be used for a large variety of domain sizes.

More details on the security of Feistel networks will be given in section 5.

3.4 Shuffling cards

One can also build a FPE scheme based on the idea of shuffling of cards. One uses a key as input to a card shuffling algorithm.

Morris, Rogaway and Stegers provided a security proof [MRS09] for the Thorp Shuffle. The Thorp shuffle works like this. Let N be the number of cards to shuffle, and divide the cards in two equal piles. Let a fair coin decide whether to first drop the bottom card of the right or left pile, and then drop the card from the other pile. Continue until the two piles are empty. This was one 'Thorp round', repeat for as a suitable number of rounds.
The special thing about the Thorp shuffle is that it is actually equivalent to a maximally unbalanced Feistel network for $N = 2^n$. A maximally unbalanced Feistel network, means that the round function takes $n-1$ bits as input and outputs a single bit. It takes n Feistel rounds to obtain one 'Thorp round'.

Morris, Rogaway and Stegers [MRS09] proved strong security bounds for the Thorp shuffle. However, the technique is only suitable for small domains due to the high number of Feistel rounds required. For example using a block cipher (like AES) as the round function, means the number of rounds, becomes the number of block cipher calls.

In general, shuffling cards technique are only suitable for small domain sizes.

3.5 VIL

A block cipher can be consider an FPE-scheme on $\{0,1\}^n$. With a mode of operation the input, of length $m > n$, will be padded and the output will be a multiple of n which is larger than the input length. Bellare and Rogaway [BR99] proposed VIL as a mode of operation that will produce a ciphertext of exactly the same length as the plaintext.

First the message M is split in two parts, M_p of length $m-n$ and M_s of length n, suchg that $M = M_p || M_s$. Then M_p is padded to ensure the last block is complete. A tag

is then produced using DMAC so that $\sigma = \texttt{DMAC}_{K1,K2}(M_p \| pad(M_p) \| M_s)$. The tag is used as the IV for counter mode to encrypt M_p using a different key, $K3$, which thne produces the ciphertext C_p. The fulle ciphertext is then $C = \sigma \| C_p$. From σ and C_p we can then recover M_p which we can then use to recover M_p.

As described here VIL only works for messages which are longer than the block length. Bellare and Rogaway proposed to use a Feistel Network when the message length was smaller, without providing any details. An advantage of this scheme is that we only need to perform two block cipher calls per block, whereas for many other schemes we need several block cipher calls per round of a Feistel Network with at least 8 rounds. Notice that this is not a complete FPE scheme as the only format it supports is $\{0,1\}^{\geq n}$, and therefore only suitable for large domains.

4 Schemes

So far, we have only described more general techniques for FPE. In this section we will take a closer look into more concrete FPE schemes. We will present the two FPE schemes recommended by NIST [NIS16], FF1 and FF3. We will also present two other schemes, FNR [DF14] and DTP [Mat09], which are proposed by the industry.

4.1 FF1 and FF3

FF1, and FF3, are the two NIST recommended FPE schemes from 2016 [NIS16]. The two schemes are very similar. Both schemes builds on the Feistel construction described in section 3.3. The schemes uses slightly different round functions, but both uses AES-128 as the underlying block cipher, and offers a 128-bit security.

Further, both schemes are tweakable, meaning that the ciphers takes both a key, K, and a tweak, T, as input. Both the key and tweak selects a permutation. Hence, two equal inputs with different tweaks will (most likely) encrypt into two different outputs. Whereas the key is always secret, usually the tweak is non-secret. The use of tweaks is especially important in FPE schemes - for small domains, using tweaks can prevent attacks such as dictionary attacks.
In FF1 the tweak is concatenated with the round number and parts of the input, whereas in FF3 the tweak is XORed to the round number. This difference actually led to a practical attack on FF3 [DV17], but an easy fixed was also proposed. More on the security of the two schemes is given in section 5.

For both schemes, the following two requirements should be met:

- $radix \in [2, \ldots, 2^{16}]$
- $N^2 = radix^{minlen} \geq 100$

Where $radix$ is the base, N^2 is the domain size, and $[minlen, maxlen]$ is the range of the length for the input. The requirement $N^2 = radix^{minlen} \geq 100$, is due to a generic Meet-in-the-Middle attack on the Feistel structure by Patarin [Pat01]. However, in [NIS16]

they actually do recommend that $radix^{minlen}$ to be at least a million.

There are additional requirements, one regarding the range of the length for the input and another on the length of the tweak. Here, the two schemes differs:

FF1:	FF3:
- $2 \leq minlen \leq maxlen < 2^{32}$	- $2 \leq minlen \leq maxlen < 2\lfloor \log_{radix}(2^{96}) \rfloor$
- $len(T) \in [0, maxTlen]$	- $len(T) = 64$ bits.

The parameters, the base, *radix*, the range of length for the input strings, $[minlen, maxlen]$, and for FF1, the maximum length of the tweak, $maxTlen$, are specified prior to the use of the encryption and decryption algorithms.

Notice that FF1, compared to FF3, offers greater length of input, and flexibility in the length of the tweak. This also implies that one can choose to have no tweak in FF1. However tweaks are recommended, especially for small domains.

Finally, 10 rounds are specified for FF1, whereas only 8 rounds for FF3. FF3 is faster than FF1, but mainly due to the fewer number of rounds.

More details on the two schemes are given in [NIS16].

4.2 FNR

FNR is a scheme proposed by Cisco [DF14] for encrypting short arbitrary-lenght bit-strings. It is based on the Naor-Reingold construction [NR99], which is an extension to the classical Feistel Network in which the first and last rounds are replaced by *pair-wise independent permutations* π and π^{-1} respectively. FNR uses 7 rounds for the Feistel Network and the two halves differ by at most one bit in length.

FNR uses an $n \times n$ invertible binary matrix, A, and a binary vector, B, of length n to construct $\pi(x) = (x \times A) \oplus B$ and the inverse permutation is then $\pi^{-1}(x) = (x \oplus B) \times A^{-1}$. The round function is simply AES in ECB mode with the intput of a round constant, the tweak, and one half of the state. The halves are combined using $\oplus$.

4.3 DTP

DTP was designed by Protegrity Corp [Mat09]. It works as a stream cipher in that encrypting $x = x_1 x_2 \ldots x_n$ we get the ciphertext $c_i = x_i + z_i \mod d$ where $c_i, x_i \in \mathbb{Z}_d$ and $z_i \in \mathbb{Z}_D$. If $d \leq 256$ then $D = 256$ otherwise $D = 2^{16}$. The z_i's are generated with a round function which is a block cipher like AES. The input to the first round is the tweak, r characters are extracted as the z_i's. For the next round the input is the remaining z_i's and the characters that were encrypted in the previous round.

4.4 DFF

FF2 was withdrawn from the final standard (NIST SP 800-38G) due to comment from the NSA[1] which resulted in the attack by Dworkin and Perlner [DP15]. DFF [VB14] was proposed to fix the issue and it is still under consideration.[2]

DFF uses a 10-round almost-balanced Feistel Network, where the round function is a block cipher. The key used in the block cipher is generated by encrypting the tweak using the master key. An offset is also generated from the tweak and the master key. The offset is XORed to the input of the round function before encryption with the block cipher. The difference between DFF and FF2 is the introduction of the offset which makes the round function dependent on the tweak.

The idea of this construction is that recovery of the key does not compromise the master key and thus encryptions using a different tweak are safe. This is refered to as *delegation*. The concept is related to identity-based encrytion, where we see the tweak as the identity. Bellare and Hoang [BH17] considers the case of identity-based format-preserving encryption and provides analysis of DFF.

5 Cryptanalysis - Security

In this section we will look at the security of FPE schemes. As many FPE schemes are built on Feistel Networks we will start by considering the security of generic Feistel Networks. This is followed by an outline of the dedicated attacks on FF3, and the two schemes from the industry FNR and DTP. At last, the security of FPE against quantum computers will briefly be touched.

We have to make the distinction between different types of attacks:

- *Distinguishing attacks*: The attacker can distinguish the encryption function from a random permutation.
- *Message-recovery attacks*: The attacker recovers the plaintext corresponding to a specifc ciphertext.
- *Key-recovery attacks*: The attacker recovers the secret key.

The strongest attack is the key-recovery attack as a key-recovery attack immediately gives the other attacks.

5.1 Generic Feistel Networks

Feistel Networks provide a flexible way to construct block ciphers for variable block sizes. They are therefore a popular choice for FPE schemes. Feistel Networks have been extensively studied by the cryptographic community since they were introduced. These results provide a good starting point when analyzing an FPE schemes.

[1] https://csrc.nist.gov/News/2014/Explanation-of-Changes-to-Draft-SP-800-38G

[2] https://csrc.nist.gov/projects/block-cipher-techniques/bcm/current-modes [7/6/2018]

Luby and Rackoff [LR88] showed that 3 rounds is enough for a Feistel Network to be secure against chosen plaintext attacks and 4 rounds is secure against chosen ciphertext attacks as well. Specifically they show that the attacker cannot distinguish the encryption function from a random permutation when the number of queries is much smaller than 2^n, where n is the block size. The proof assumes randomly chosen independent round functions, and as such do not apply directly to a specific scheme as the round functions are rarely completely random and independent.

Patarin improved the Luby-Rackoff bound of how many queries are needed to distinguish a Feistel Network from a random permutation for 5 or more rounds [Pat91].

In [Pat01] Patarin shows an attack on a 5-round Feistel Network, and considers the case where the attacker tries to distinguish a 6-round Feistel permutation generator from a random permutation generator[3]. However these results are not applicable when we just want to attack a specific instantiation and the data complexity is also prohibitive.

This is just a brief recap of some of Patarin's results on Feistel Networks, and we refer to [NPV17] for more details and results.

Most of these results assume that the Feistel Network is balanced, which is not always the case for FF1 or FF3. Another general assumption is that the branches are combined with XOR rather than the addition operation used in FF1 and FF3.

Hoang and Rogaway [HR10] provide security proofs for the case of generalized Feistel Networks, both unbalanced and networks with several branches. Different group operations are also considered as the combining operation. These results are therefore more relevant to the case of FPE as both FF1 and FF3 are captured by the generalized Feistel Networks.

Bellare et al. [BHT16] gave a message-recovery attack on Feistel-Based FPE schemes, which is feasible for FF1 and FF3 if the domain is small. Specifically they consider balanced Feistel Networks on the domaim $\{0,1\}^{2n}$, and the attack is feasible if $n \leq 4$. A caveat is that the data requirement is larger than the domain size, however they argue that the attack is still interesting as they only make a few queries for each tweak.

Hoang et al. [HTT18] extends the attack of Bellare et al. [BHT16] to the case where there are multiple targets, e.g. when the attacker wants to compromise parts of a database instead of just a single entry. In doing so they are able to improve on the attack by removing the assumption that there is a known correlation between the known plaintexts and the target plaintexts. The attack works especially well for FF3 due to the way odd length texts are handled.

5.2 FF3

The specification of FF3 allows the domain to have just 100 elements. Durak and Vaudenay [DV17] used this and the poor domain seperation to break the FF3 scheme, however, it was easily fixed. In their follow-up work they show that the fixed version also do

[3] This would be the case if we ask for encryptions with different keys.

not offer 128-bit security and extends the attack to even larger domains by considering the round-functions as black-boxes and then recovering them [DV18]. This only works for small domains, that is, with less than 17^2 elements, or the time complexity exceeds 2^{128}.

There are a few easy ways to fix the FF3 scheme. The attack only works for small domains, thus increasing the lower bound on the domain size would make the attack less efficient than exhaustive search. However the domain size is often decided by the application and thus increasing the lower bound is not really an option. Another way to fix it is to increase the number of rounds. According to Durak and Vaudenay [DV18] 12 rounds would be sufficient to provide 128-bit security even for domains of just 100 elements.

Due to these attacks NIST recommends not using FF3 as a general-purpose FPE[4].

5.3 FNR

FNR uses the Naor-Reingold construction and thus it is not obvious how to apply the message recovery attacks on Feistel Networks to this scheme. That problem was solved by Hoang et al. [HTT18]. The attack aims to recovery a number of plaintexts from a list of targets $Z_1,\dots,Z_p$. To achieve this they get a list of know plaintext $X_1,\dots X_t$. The attacker gets the encryptions for each of q tweaks $T_1,\dots,T_q$, that is the enctyptions of X_j are then $C_{1,j}\dots,C_{q,j}$, and the encryptions of target Z_k are $C'_{1,k}\dots,C'_{q,k}$

The attack then works as follows. Consider the distribution of $C_{i,j}\oplus C'_{i,k}$ for $1\leq i\leq q$, i.e., the XOR of the encyptions of X_j and Z_k under all tweaks. If $\pi(X_j)$ and $\pi(Z_k)$ have different right halves, then the distribution is non-singular and we can't recover the plaintext. However if the right halves are the same the distribution is singular and the plaintext can be recover since the most likely value will be $X_j\oplus Z_k$. We refer to [HTT18] for the details of why this is true.

5.4 DTP

Hoang et al. [HTT18] also presents a chiphertext only message recovery attack on DTP. Recall that $c_i = x_i + z_i \mod d$ where $c_i, x_i \in \mathbb{Z}_d$ and $z_i \in \mathbb{Z}_D$. The attack exploits that $z_i \mod d$ is not uniformly distributed if d does not divide D. The probability that $z_i = a$ is $\frac{\lceil D/d\rceil}{D}$ if $a < D \mod d$ and $\frac{\lfloor D/d\rfloor}{D}$ otherwise.

For a fixed target, x_i, we can exploit this imbalance by asking for the encryption of the message multiple times. Since a fresh tweak is used each time, z_i will be uniformly distributed on $\mathbb{Z}_D$. By inspecting the histogram of the ciphertexts we will find a block of $D \mod d$ message which have a higher value, the first of which corresponds to $z_i \mod d = 0$, hence this position is the value of the message.

5.5 Quantum computers

If we consider the security of the system for medium- to long-term we also have to consider the possibility of an adversary with a quantum computer. In the case where

[4] https://csrc.nist.gov/News/2017/Recent-Cryptanalysis-of-FF3

the adversary can make queries to the cryptographic function in a superposition the Luby-Rackoff bound is no longer valid due to Simon's algorithm [KM10, KLLN16]. These results apply to the standard Feistel Network in the group $(\mathbb{Z}/2)^n$. Alagic and Russell [AR17] showed that replacing the group with $(\mathbb{Z}/2^n)$, i.e. replacing XOR with modular addition, makes the system secure even against a quantum attacker. FF1 and FF3 already use modular addition, hence it makes difference here, but it is worth considering for other schemes.

Scheme	References
FF3 [NIS16]	[BHT16], [DV17], [DV18] [HTT18]
FF1 [NIS16]	[BHT16], [DV17], [DV18] [HTT18]
FF2	[DP15]
FNR	[HTT18]
DTP	[HTT18]

Table 1: Overview

6 Discussion

As we saw in Section 5 constructing a good general-purpose FPE scheme is hard. The main challenge is the domain size which can be very small. The domain is determined by the application and therefore it is not a feasible solution to just put a lower limit on the size.

A common construction is to use a Feistel Network with a complex round function like AES. The complexity of the round function means that the number of rounds must be kept low for performance reasons. The problem is that the complex round function does not add any security to small domains. If we consider a balanced Feistel Network with domain size N^2 and modular addition mod N as the combining operation, then for a fix key and tweak, the round function can be considered as a look-up table of size N. An attacker then just have to recover r of these functions to have an equivalent description of the cipherwith that particular key and tweak.

Increasing the number of rounds for small domains is a way to increase the security. This is the approach taken by the designers of FFX in the parameter collections they present in the appendix of [BRS10]. However this is not very efficient. If efficiency is a concern, a dedicated design for small domains with simpler round functions and more rounds is probably preferable.

7 List of Publications

7.1 Theory

– [BR99]: Provides a mode for encrypting bitstrings of arbitrary length such that the ciphertext has the same length, i.e., an FPE scheme on $\{0,1\}^*$. The method works

Outline	Reference
FF2 (a.k.a. VAES3) invokes Feistel rounds with different subkeys for every tweak. Weakness in subkey generation lead to chosen-plaintext attack.	[DP15]
New message-recovery attack on Feistel-based FPE constructions on small domains. Full recovery of one-byte messages using 2^{32} data for FF3, and 2^{40} data for FF1. Even though the total number of data needed exceeds the codebook, it is only a few data per tweak. Increasing the number of rounds on small domains will prevent the attack.	[BHT16]
Exploits bad domain separation of FF3, to create a practical slide attack with time complexity $O(N^5)$, where N^2 is the domain size. Suggests an easy fix on the tweak space to prevent the attack on FF3. Also, provided a generic Feistel network attack, which showed that FF1 and FF3 cannot provide 128-bit security when the domain branch is 7 and between 7 and 10, respectively.	[DV17]
Generic round-function-recovery attack for Feistel networks on small domains. Shows that FF1 and fixed version of FF3 from [DV17] cannot provide 128-bit security when the domain branch is smaller than or equal 11 and 17, respectively. They show that increasing the number of rounds on small domains will prevent the attack.	[DV18]
Multiple-targets attack on Feistel-based FPE constructions on small domains. Improvement of the attack [BHT16]. They also descover a new weakness to FF3 - how FF3 handles odd length domains leads to a substantial speed-up in their attacks. More rounds could prevent the attacks. Also, they show a message-recovery attack on FNR constructions on small domains, and a strong ciphertext-only attack on a variant of the DTP construction on even large domains.	[HTT18]

Table 2: Overview of existing attacks on FPE schemes

Attacked Schemes	Attack Type	Notes	Time complexity	Data complexity	Reference
FF2	Key-recovery	-	-	-	[DP15]
Generic Feistel	Message-recovery	$N \geq 4$	$O(N^{r-2}\log N)$	$O(N^{r-2}\log N)$	[BHT16]
FF3	Round-function-recovery	$r = 8$	$O(N^5)$	$O(N^{11/6})$	[DV17]
Generic Feistel	Round-function-recovery	$r \geq 5$	$O(N^{(r-5)N+\sqrt{N}+3})$	$O(N^{\frac{3}{2}})$	[DV17]
Generic Feistel	Round-function-recovery	$r \geq 6$	$N^{O\left(N^{1-\frac{1}{r-2}}\right)}$	$O(N^{2-\frac{1}{r-2}})$	[DV18]
Generic Feistel	Multi-target message-recovery	$N \geq 8$	$O(\log(N)^{1.5}N^{r-2} + N^{r-2}\log(N)p)$	$O(\log(N)^{1.5}N^{r-2} + N^{r-3}\log(N)p)$	[HTT18]

Table 3: Overview of existing attacks on FPE schemes, where r is the number of rounds, p the number of target-messages, and N^2 is the domain size.

as long as the message is longer than the block size of the underlying blockcipher. For smaller messages they suggest using a Feistel Network.
- [BR02]: Studies encryption on $[0, k-1]$. Their methods work for small and large domains but there is a gap where the methods are not efficient and/or secure. The gap is from about 2^{30} to 2^{60}.
- [BRRS09]: Contains a rigorous definition of FPE. Two appoaches are analyzed, *rank-then-encipher* and *cycle-walking*, as solutions.
- [MRS09]: Fixes the problem of intermediate domains of [BR02] by using the Thorp shuffle.
- [BH17]: Introduces identity-based format-preserving encryption, where different keys are used for different identities such that a compromised key does not compromise the security of other indentities.

7.2 Ciphers and Modes

- [NIS16]: Defines the standards FF1 and FF3. FF2 withdrawn after comments from the NSA.
- [BRS10]: The mode on which FF1 is based.
- [BPS10]: The mode on which FF3 is based.
- [VB14]: An update to FF2 which should prevent the attack.
- [WRB15]: A practical solution for general formats, if the formats are not too large. Solution has been patented [WR17].

7.3 Feistel Networks

Most schemes are based on Feistel networks since they are easy to scale. We will here give an overview of the results related to Feistel Netwoks, we refer to [NPV17] for the details.

- [LR88]: Proves that a Balanced Feistel Network is secure against chosen plaintext attacks with 3 rounds and against adaptively chosen plaintext/ciphertext attacks with 4 rounds.
- [ZMI89]: Investigates Feistel Networks where the input is split in more than two parts also called Generalized Feistel Networks.
- [Pat91]: Improves the Luby-Rackoff bound for 5- and 6-round Feistel Networks.
- [SK96]: Examines Unbalanced Feistel Networks, that is Feistel Networks where the two parts do not have the same length.
- [Pat03]: Considers the case where the number of queries, m, an attacker can make is $m \ll 2^{n(1-\varepsilon)}$ for every $\varepsilon > 0$ compared to $m \ll 2^{n/2}$ fro the Luby-Rackoff bound. Shows that 7 rounds are enough for security against adaptively chosen plaintext attacks, and 10 rounds is enough for security against adaptively chosen plaintext and ciphertext attacks.
- [Pat04]: Changes the bound on queries to $m \ll 2^n$ and improves the number of rounds to 5 and 6 for adaptively chosen plaintext and adaptively chosen plaintext and ciphertext attacks respectively.
- [HR10]: Considers the security of Generalized Feistel Networks.

7.4 Cryptanalysis

Much cryptanalysis focuses on the general problem of reconstructing the round functions of an FN with secret round functions. These attacks also apply to many FPE. Reconstructing the secret round function might be easier than finding the secret key for a small domain. Attacks based on Patarin's Mirror Theory, and in general attacks by Patarin, could also be interesting to consider.

- [BLP15]: Provides function recovery attacks for $\oplus$-FN for 5, 6, and 7 rounds and for 5 rounds against $\boxplus$-FN.
- [DDKS15]: Combines MITM and Dissection attacks to get better memory complexities for attacks on FN without affecting the time and data complexities.
- [DP15]: Attack on FF2 which is why it was withdrawn.
- [DV17]: Presents cryptanalysis of the FF3 standard on small domains. The weakness is easy to fix as the authors point out.
- [DV18] (ACNS 2018): Improves on the results of [DV17]. Impacts both FF1 and FF3 on small domains. FF1 and FF3 cannot offer 128-bit security when the domain size of a branch is smaller than or equal to 11 and 17 respectively. This attack also applies to the fixed version of FF3 from [DV17]
- [HTT18](CRYPTO 2018): Provides message recovery attacks on Feistel based FPEs like FF1 and FF2, FNR, and DTP.

References

AR17. Gorjan Alagic and Alexander Russell. Quantum-Secure Symmetric-Key Cryptography Based on Hidden Shifts. In *EUROCRYPT (3)*, volume 10212 of *Lecture Notes in Computer Science*, pages 65–93, 2017.

BH17. Mihir Bellare and Viet Tung Hoang. Identity-Based Format-Preserving Encryption. In *CCS*, pages 1515–1532. ACM, 2017.

BHT16. Mihir Bellare, Viet Tung Hoang, and Stefano Tessaro. Message-Recovery Attacks on Feistel-Based Format Preserving Encryption. In *ACM Conference on Computer and Communications Security*, pages 444–455. ACM, 2016.

BJPA11. G. Bertoni, J.Daemen, M. Peeters, and G. Van Assche. The KECCAK SHA-3 submission, 2011. `https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf` [accessed 11/06/2018].

BLP15. Alex Biryukov, Gaëtan Leurent, and Léo Perrin. Cryptanalysis of Feistel Networks with Secret Round Functions. In *SAC*, volume 9566 of *Lecture Notes in Computer Science*, pages 102–121. Springer, 2015.

BPS10. Eric Brier, Thomas Peyrin, and Jacques Stern. BPS: a Format-Preserving Encryption Proposal, April 2010. `https://pdfs.semanticscholar.org/0be5/d4c77e333d78ddab5c4bf55d15649a660771.pdf` [accesed 10/04/2018].

BR99. Mihir Bellare and Phillip Rogaway. On the Construction of Variable-Input-Length Ciphers. In *FSE*, volume 1636 of *Lecture Notes in Computer Science*, pages 231–244. Springer, 1999.

BR02. John Black and Phillip Rogaway. Ciphers with Arbitrary Finite Domains. In *CT-RSA*, volume 2271 of *Lecture Notes in Computer Science*, pages 114–130. Springer, 2002.

BRRS09. Mihir Bellare, Thomas Ristenpart, Phillip Rogaway, and Till Stegers. Format-Preserving Encryption. In *Selected Areas in Cryptography*, volume 5867 of *Lecture Notes in Computer Science*, pages 295–312. Springer, 2009.

BRS10. Mihir Bellare, Phillip Rogaway, and Terence Spies. The FFX Mode of Operation for Format-Preserving Encryption, February 2010. https://csrc.nist.gov/csrc/media/projects/block-cipher-techniques/documents/bcm/proposed-modes/ffx/ffx-spec.pdf [accesed 10/04/2018].

BS97. Micheal Brightwell and Harry E. Smith. Using Datatype-Preserving Encryption To Enhance Data Warehouse Security, 1997. https://csrc.nist.gov/csrc/media/publications/conference-paper/1997/10/10/proceedings-of-the-20th-nissc-1997/documents/141.pdf [accessed 22/05/2018].

DDKS15. Itai Dinur, Orr Dunkelman, Nathan Keller, and Adi Shamir. New Attacks on Feistel Structures with Improved Memory Complexities. In *CRYPTO (1)*, volume 9215 of *Lecture Notes in Computer Science*, pages 433–454. Springer, 2015.

DF14. Sashank Dara and Scott R. Fluhrer. FNR: Arbitrary Length Small Domain Block Cipher Proposal. In *SPACE*, volume 8804 of *Lecture Notes in Computer Science*, pages 146–154. Springer, 2014.

DP15. Morris Dworkin and Ray A. Perlner. Analysis of VAES3 (FF2). *IACR Cryptology ePrint Archive*, 2015:306, 2015. http://eprint.iacr.org/2015/306.

DR02. Joan Daemen and Vincent Rijmen. *The design of Rijndael: AES — the Advanced Encryption Standard*. Springer-Verlag, 2002.

DV17. F. Betül Durak and Serge Vaudenay. Breaking the FF3 format-preserving encryption standard over small domains. In *CRYPTO (2)*, volume 10402 of *Lecture Notes in Computer Science*, pages 679–707. Springer, 2017.

DV18. F. Betül Durak and Serge Vaudenay. Generic Round-Function Recovery for Feistel Networks over Small Domains. *IACR Cryptology ePrint Archive*, 2018:108, 2018. http://eprint.iacr.org/2018/108.

HR10. Viet Tung Hoang and Phillip Rogaway. On Generalized Feistel Networks. In *CRYPTO*, volume 6223 of *Lecture Notes in Computer Science*, pages 613–630. Springer, 2010.

HTT18. Viet Tung Hoang, Stefano Tessaro, and Ni Trieu. The Curse of Small Domains: New attacks on Format-Preserving Encryption. *IACR Cryptology ePrint Archive*, 2018:556, 2018. http://eprint.iacr.org/2018/556.

KLLN16. Marc Kaplan, Gaëtan Leurent, Anthony Leverrier, and María Naya-Plasencia. Breaking Symmetric Cryptosystems Using Quantum Period Finding. In *CRYPTO (2)*, volume 9815 of *Lecture Notes in Computer Science*, pages 207–237. Springer, 2016.

KM10. Hidenori Kuwakado and Masakatu Morii. Quantum distinguisher between the 3-round Feistel cipher and the random permutation. In *ISIT*, pages 2682–2685. IEEE, 2010.

LR88. Michael Luby and Charles Rackoff. How to Construct Pseudorandom Permutations from Pseudorandom Functions. *SIAM J. Comput.*, 17(2):373–386, 1988.

Mat09. Ulf T. Mattsson. FORMAT CONTROLLING ENCRYPTION USING DATATYPE PRESERVING ENCRYPTION. Cryptology ePrint Archive, Report 2009/257, 2009. https://eprint.iacr.org/2009/257.

MRS09. Ben Morris, Phillip Rogaway, and Till Stegers. How to Encipher Messages on a Small Domain. In *CRYPTO*, volume 5677 of *Lecture Notes in Computer Science*, pages 286–302. Springer, 2009.

NIS13. NIST. NIST Special Publication 800-38G Draft, Recommendation for Block Cipher Modes of Operation: Methods for Format-Preserving Encryption, July 2013. http://www.gocs.de/pages/fachberichte/archiv/144-sp800_38g_draft.pdf [accessed 08/06/2018].

NIS16. NIST. NIST Special Publication 800-38G, Recommendation for Block Cipher Modes of Operation: Methods for Format-Preserving Encryption, March 2016. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-38G.pdf [accessed 10/04/2018].

NPV17. Valérie Nachef, Jacques Patarin, and Emmanuel Volte. *Feistel Ciphers - Security Proofs and Cryptanalysis*. Springer, 2017.

NR99. Moni Naor and Omer Reingold. On the Construction of Pseudorandom Permutations: Luby-Rackoff Revisited. *J. Cryptology*, 12(1):29–66, 1999.

Pat91. Jacques Patarin. New Results on Pseudorandom Permutation Generators Based on the DES Scheme. In *CRYPTO*, volume 576 of *Lecture Notes in Computer Science*, pages 301–312. Springer, 1991.

Pat01. Jacques Patarin. Generic Attacks on Feistel Schemes. In *ASIACRYPT*, volume 2248 of *Lecture Notes in Computer Science*, pages 222–238. Springer, 2001.

Pat03. Jacques Patarin. Luby-Rackoff: 7 Rounds Are Enough for $2^{n(1-\varepsilon)}$Security. In *CRYPTO*, volume 2729 of *Lecture Notes in Computer Science*, pages 513–529. Springer, 2003.

Pat04. Jacques Patarin. Security of Random Feistel Schemes with 5 or More Rounds. In *CRYPTO*, volume 3152 of *Lecture Notes in Computer Science*, pages 106–122. Springer, 2004.

SK96. Bruce Schneier and John Kelsey. Unbalanced Feistel Networks and Block Cipher Design. In *FSE*, volume 1039 of *Lecture Notes in Computer Science*, pages 121–144. Springer, 1996.

Spi08. Terence Spies. Format Preserving Encryption, 2008. https://www.voltage.com/wp-content/uploads/Voltage-Security-WhitePaper-Format-Preserving-Encryption.pdf [accessed 23/05/2018].

VB14. Joachim Vance and Mihir Bellare. An Extension of the FF2 FPE Scheme, July 2014. https://csrc.nist.gov/CSRC/media/Projects/Block-Cipher-Techniques/documents/BCM/proposed-modes/dff/dff-ff2-fpe-scheme-update.pdf[accessed 7/06/2018].

WR17. Mor Weiss and Boris Rozenberg. Complex Format-Preserving Encryption Scheme, April 25th 2017. Patent No.: US9634838B2.

WRB15. Mor Weiss, Boris Rozenberg, and Muhammad Barham. Practical Solutions For Format-Preserving Encryption. *CoRR*, abs/1506.04113, 2015. http://arxiv.org/abs/1506.04113.

ZMI89. Yuliang Zheng, Tsutomu Matsumoto, and Hideki Imai. On the Construction of Block Ciphers Provably Secure and Not Relying on Any Unproved Hypotheses. In *CRYPTO*, volume 435 of *Lecture Notes in Computer Science*, pages 461–480. Springer, 1989.