Syndrome Based Collision Resistant Hashing

Matthieu Finiasz

ENSTA ParisTech

Outline

- Description of the FSB Hash Function
- Classic Attacks
- Recent Attacks
- Proposed Improvements and Parameters

Description of the FSB Hash Function

- FSB is based on the Merkle-Damgård construction
  - we only need to define a compression function.
- For security reasons, the internal state has to be larger than the output:
  - we add a final compression function.
- The compression function relies on a binary matrix $\mathcal{H}$
  - the output is the XOR of columns of $\mathcal{H}$,
  - security is related to the Syndrome Decoding problem.

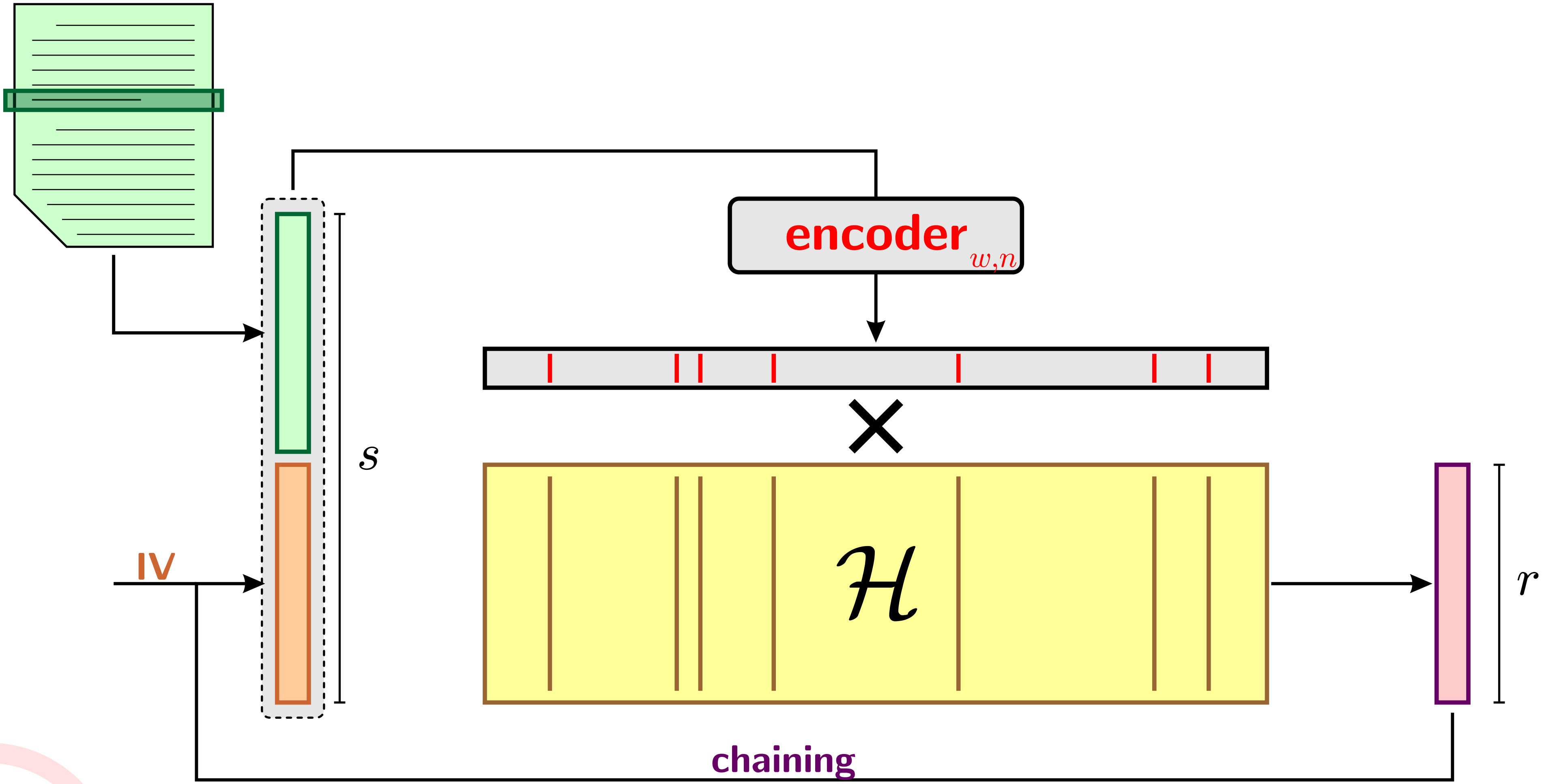
Compression Function

- The compression function has several parameters:
  - $r \times n$, the size of matrix $\mathcal{H}$,
  - w, the number of columns to XOR.

- The compression function takes s input bits and outputs an r-bit syndrome.
  - the s bits are converted to a binary word of weight w and length n using a constant weight encoder.
  - this binary word is multiplied by $\mathcal{H}$ to obtain the output syndrome.

- The value of s depends on the encoder choice.

Compression Function



- This compression function seems very simple. Why should it be secure?

- If $\mathcal{H}$ is seen as the parity check matrix of a binary code :
  - Inversion requires to find a word of low weight having a given syndrome
    $\rightarrow$ exactly the Syndrome Decoding (SD) problem.
  - Collision requires to find a word of twice this low weight with null syndrome
    $\rightarrow$ again, the SD problem.

FSB Specification

- To completely specify FSB, we need to define:
  - the structure of $\mathcal{H}$,
  - the constant weight encoder,
  - the final compression function
    $\longrightarrow$ not the scope of this presentation,
  - the parameters n, w and r
    $\longrightarrow$ will depend on the target security.

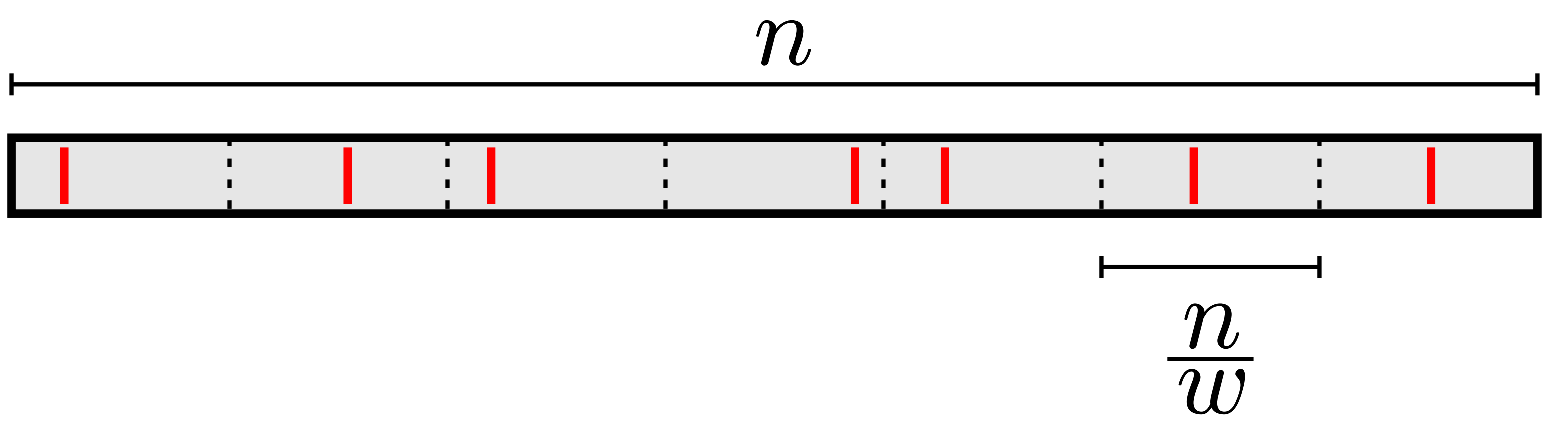
The Original FSB

[Augot-Finiasz-Sendrier - Mycrypt 2005]

In this original version the choices are as follows.

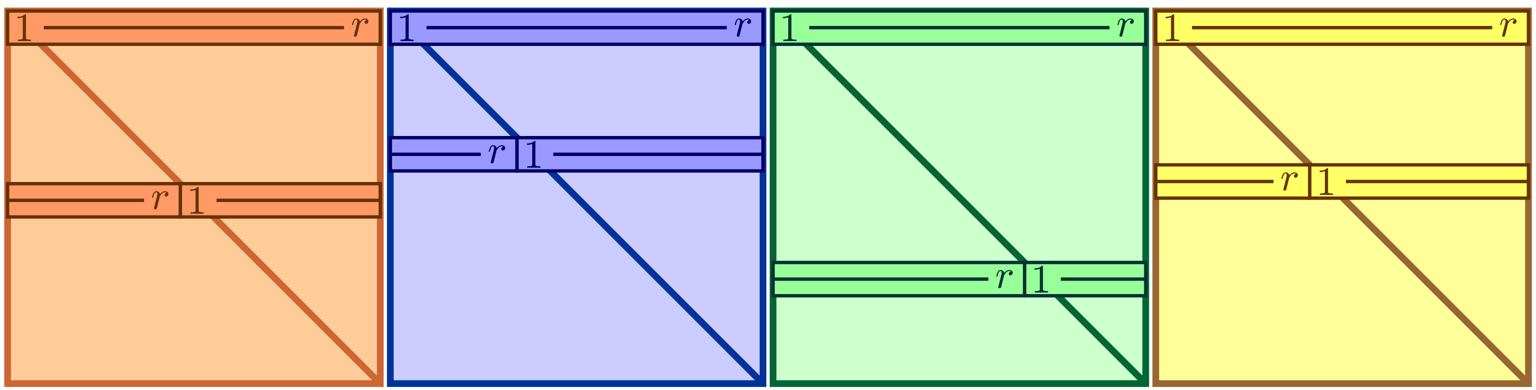
- $\mathcal{H}$ is a random binary matrix
  → FSB has a large description.

- The constant weight encoder uses regular words
  - we assumed that no attack can take advantage of this.



This new version uses a structured $\mathcal{H}$.

- $\mathcal{H}$ is Quasi-Cyclic.
  - its first line describes it completely.

$\mathcal{H} =$ 

- Regular words are still used.

Classical Attacks

Collision Search

- Finding a collision on FSB requires to:
  - find two words of weight w with identical syndrome,
  - find a word of weight $\leq 2w$ with null syndrome.

- Two main algorithms solve this coding theory problem:
  - Decoding algorithm: using the Canteaut-Chabaud algorithm (or the Bernstein-Lange-Peters variant),
    → efficient for a single solution
  - Birthday paradox: using Wagner's generalized birthday technique.
    → efficient for a large number of solutions.

- This attack has a cost of $2^{\frac{r}{a+1}}$ where the maximum possible a depends on the parameters of FSB.

- This will be the reference attack for FSB
  - parameters will be chosen so that no other attack performs better.

- If $s > r$ (that is, the compression function compresses):
  - $a = 3$ is always possible,
  - a security of $2^{\frac{r}{2}}$ against collision is impossible.
    $\longrightarrow$ This is why we need a final compression function.

Choice of the Constant Weight Encoder

- The choice of the encoder is a tradeoff between:
  - the bit efficiency: the number of input bits s,
  - the speed efficiency: the cost of this encoder.

- Two extreme solutions:
  - one to one encoder: all words of weight w are mapped
    → largest possible $s = \log_2 \binom{n}{w}$.
  - regular encoder: uses regular words
    → $s = w \times \log_2 \frac{n}{w}$, but no computation are required.

- Larger s requires less compression rounds, but regular words are still, by far, the fastest solution.

Choice of the Constant Weight Encoder

- Concerning security:
  - Could regular words be a weakness?
  - No, a collision on regular words is also a collision for the one to one encoder.

    $\longrightarrow$ the one to one encoder is the weakest encoder.
- Can another encoder be more secure?
  - Probably, but we have no proof...

We now evaluate security considering the one to one encoder, but use regular words in practice.

Recent Attacks

Linearization Attack
[Saarinen - Indocrypt 2007]

- This attack works for large values of w, say $w = \frac{r}{2}$
  - we look for a null XOR of $2w$ columns of r bits,
  - one chooses $2w$ pairs of columns $h_i^0$and h_i^1.
  - let $\mathcal{H}'$ the matrix with columns $h_i' = h_i^1 - h_i^0$.

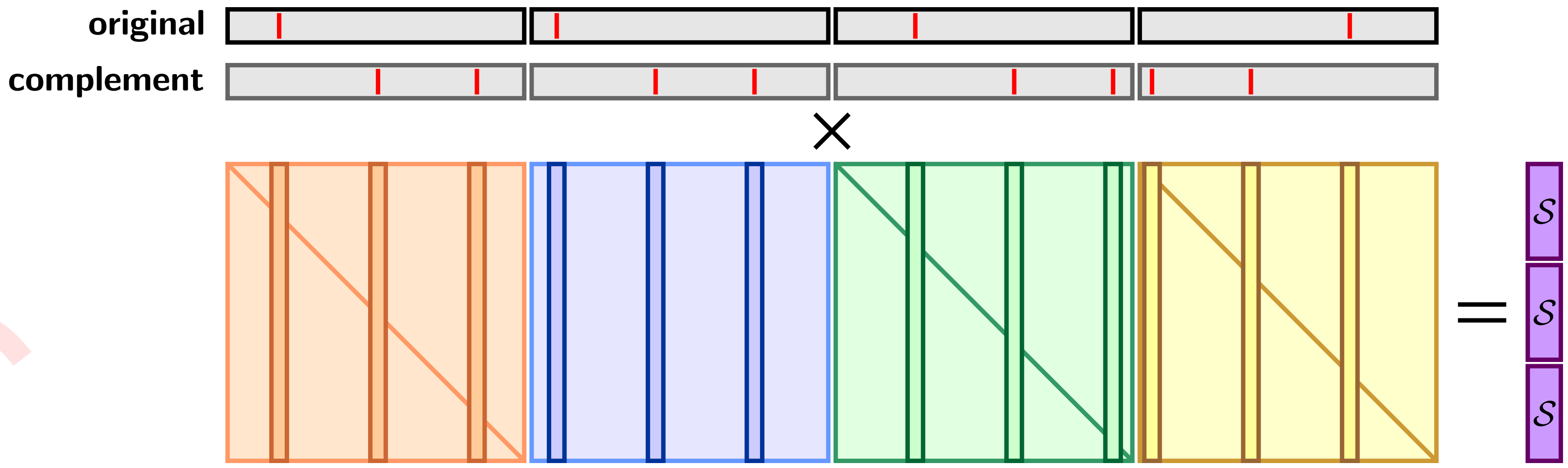
    → a collision is a vector B such that:

$$\mathcal{H}' \times B = \sum h_i^0.$$

- For $w \geq \frac{r}{2}$, collisions are found in polynomial time.
  - for $\frac{r}{4} \leq w \leq \frac{r}{2}$ a variation of this attack still applies.

All proposed parameters must verify $w < \frac{r}{4}$.

- This attack only applies when $\mathcal{H}$ is quasi-cyclic and when the block size r is divisible by some p.
- One chooses inputs to obtain $p-$repeating syndromes:
  - $\frac{2w}{p}$ columns are chosen freely,
  - for each column, $p-1$ other columns with the same index $\mathrm{mod}\ \frac{r}{p}$ are chosen in the same block.



- This attack only applies when $\mathcal{H}$ is quasi-cyclic and when the block size r is divisible by some p.
- One chooses inputs to obtain $p-$repeating syndromes:
  - $\frac{2w}{p}$ columns are chosen freely,
  - for each column, $p-1$ other columns with the same index $\mod \frac{r}{p}$ are chosen in the same block.
- Now Wagner's attack can apply to $2w' = \frac{2w}{p}$ and $r' = \frac{r}{p}$.
  → this improves the complexity of the attack a lot.

If a quasi-cyclic matrix is to be used, r must be prime.

IV Weakness

[Fouque-Leurent - CT-RSA 2008]

- Originally, the IV bits and message bits are not mixed:
  - r bits are used to compute a syndrome, $s - r$ another, and both are XORed.
  - If a collision is found using only the $s - r$ last input bits, it is IV-independent.

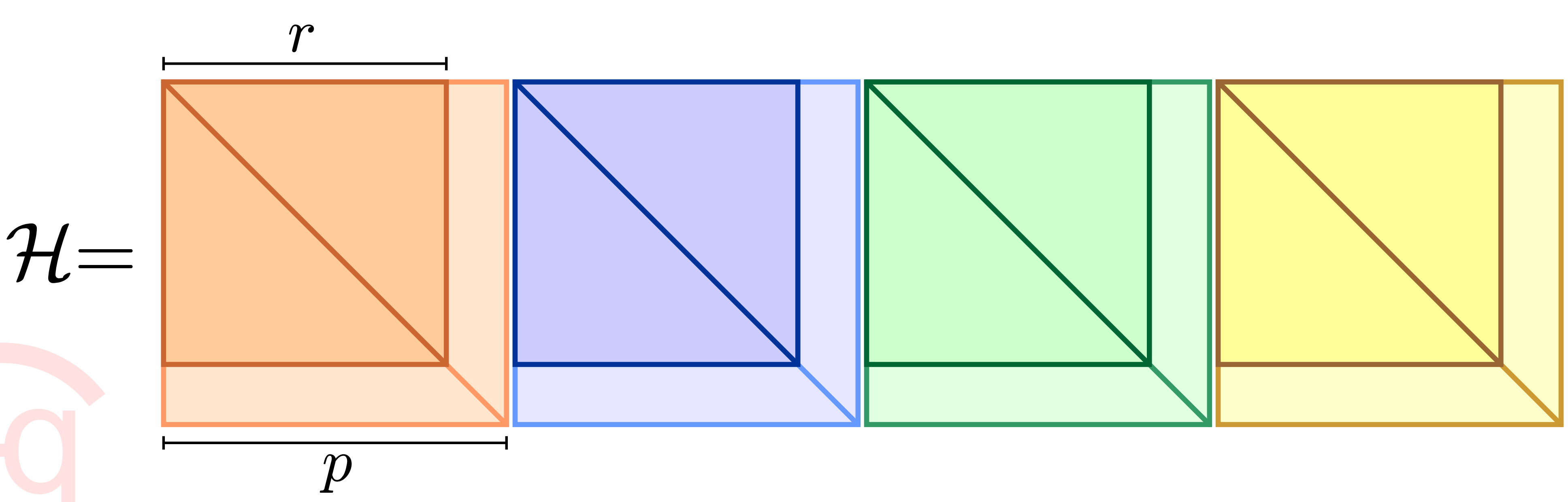
- This makes using FSB impossible for some applications.

The input should be “mixed” so that each position depends on both the IV and the message.

Proposed Improvements and Parameters

Using a Truncated Quasi-Cyclic Matrix

- Quasi-cyclic matrices are necessary, and r being a power of 2 helps implementation
  - we need to avoid quasi-cyclic divisibility attacks.
- We could use a quasi-cyclic matrix of cyclicity p and truncate it to r lines.



Using a Truncated Quasi-Cyclic Matrix

- Quasi-cyclic matrices are necessary, and r being a power of 2 helps implementation
  - ▷ we need to avoid quasi-cyclic divisibility attacks.
- We could use a quasi-cyclic matrix of cyclicity p and truncate it to r lines.
- We use p prime such that 2 is a generator of $GF(p)$.
  - ▷ such quasi-cyclic codes have good properties,
  - ▷ p close to r to keep these properties.
  - → $(r,p) \in \{(512,523),(768,773),(1024,1061)\ldots\}$

Input Bits Interleaving

- To address the IV weakness, input bits have to be mixed:
  - a simple interleaving should be enough,
  - each position is defined by $\log_2 \frac{n}{w}$ bits
    → $\frac{r}{s} \log_2 \frac{n}{w}$ from the IV, $\frac{s-r}{s} \log_2 \frac{n}{w}$ from the message
- Depending on the value of r, w and n this interleaving might have to be irregular to obtain integers
  - interleaving should not slow down hashing a lot.

Previously Proposed Parameters

- Original version:
  - Short Hash: security of $2^{72.2}$ as the gain from regular words is no longer taken into account,
  - Fast Hash: security of $2^{59.9}$ due to linearization attacks,
  - Intermediate Hash: security still above 2^{80}.

- Quasi-Cyclic version:
  - all parameters used powers of 2 for r

    $\longrightarrow$ all broken with the divisibility attack...

- We select $r = 512$, thus $\log_2 \binom{n}{w} \leq 1688$ to be secure.
- $w = 128$ is the maximum to avoid linearization attacks which gives $n = 2^{18}$.

- The truncated quasi-cyclic matrix uses $p = 523$,
- Each of the w positions is coded with 11 bits
  → 4 from the IV, 7 from the message.

- Matrix $\mathcal{H}$ has a description of ~ 32kB.

- We select $r = 768$, thus $\log_2 \binom{n}{w} \leq 2048$ to be secure.
- $w = 192$ is the maximum to avoid linearization attacks, we choose $n = 3 \times 2^{14}$.

- The truncated quasi-cyclic matrix uses $p = 773$,
- Each of the w positions is coded with 8 bits
  → 4 from the IV, 4 from the message.

- Matrix $\mathcal{H}$ has a description of ~ 6kB.

Conclusion

- Taking into account all newly proposed attacks we were able to:
  - precisely evaluate which parameters remain secure,
  - propose new optimizations of FSB,
  - propose new/improved parameters.

- Some work remains:
  - precisely evaluate the requirements for the final compression function,
  - select a (provably) secure final compression function.