Cryptanalysis of SFLASH with Slightly Modified Parameters

Vivien Dubois, Pierre-Alain Fouque and Jacques Stern

Ecole normale supérieure, Paris

SFLASH is a multivariate signature scheme designed by Patarin-Goubin-Courtois in 2001

SFLASH is a multivariate signature scheme designed by Patarin-Goubin-Courtois in 2001

It is reputed for being very fast

SFLASH is a multivariate signature scheme designed by Patarin-Goubin-Courtois in 2001

It is reputed for being very fast

It is reputed for being very light, suitable for low-end smartcards

SFLASH is a multivariate signature scheme designed by Patarin-Goubin-Courtois in 2001

It is reputed for being very fast

It is reputed for being very light, suitable for low-end smartcards

It is recommended by the NESSIE European Consortium since 2003

Topic of the talk

We show that slight modifications of the parameters render the scheme insecure

Topic of the talk

We show that slight modifications of the parameters render the scheme insecure

More precisely...

- SFLASH is some instance of C^{*-} schemes [PGC98]
- All C^{*-} schemes are currently considered secure

Topic of the talk

We show that slight modifications of the parameters render the scheme insecure

More precisely...

- SFLASH is some instance of C^{*-} schemes [PGC98]
- All C^{*-} schemes are currently considered secure
- We show that a large class of C^{*-} schemes is insecure
- This class is defined by the non-coprimality of two parameters
- The attack does not apply to the parameters of SFLASH, but the choice of SFLASH parameters was not justified

Organisation of the talk

- A few basics about multivariate schemes
- Description of C^{*-} schemes
- Basic strategy for attacking C^{*-} schemes
- Description of the attack

Multivariate Schemes

- A family of asymmetric schemes
- Hard problems involve MQ polynomials over a finite field $\mathbb{F}_q$
- e.g. solving an MQ system is NP-hard and currently requires exponential time and memory on average

Multivariate Schemes

- A family of asymmetric schemes
- Hard problems involve MQ polynomials over a finite field $\mathbb{F}_q$
- e.g. solving an MQ system is NP-hard and currently requires exponential time and memory on average

The Generic Multivariate Construction

- Hiding an easily invertible function using linear transforms

$$\boldsymbol{P} = T \circ P \circ S$$

- Schemes differ from the type of easy function embedded

The C^* Scheme

C^* was proposed by [MI88] and broken by Patarin in 95

Short Description of C^*

- The internal function is a monomial over $\mathbb{F}_{q^n}$

$$P(x) = x^{1+q^\theta} = x.x^{q^\theta}$$

- $\mathbb{F}_{q^n}$ is a n-dimension vector space over $\mathbb{F}_q$, isomorphic to $(\mathbb{F}_q)^n$
- Since a q-powering is linear in $\mathbb{F}_{q^n}$, $P(x)$ is quadratic
- $P(x)$ is an n-tuple of mult. quad. polynomials $(p_1, \ldots, p_n)$

$$p_k(x_1, \ldots, x_n) = \alpha_{12}x_1x_2 + \alpha_{13}x_1x_3 + \ldots$$

- P can be inverted by raising to the inverse power of $1 + q^\theta$
- $\boldsymbol{P} = T \circ P \circ S$ is the public key

The attack by Patarin on C^*

- Any element x and $y = P(x)$ satisfy

$$y^{q^\theta - 1} = x^{(q^\theta+1)(q^\theta-1)} \qquad \Longrightarrow \qquad x.y^{q^\theta} - y.x^{q^{2\theta}} = 0$$

- Consequence : plain and cipher texts are bilinearly related
- These bilinear equations can be determined using pairs (x, y)
- Then, for any specified value y, x is solution of a system of linear equations

C^{*-} Schemes

C^{*-} schemes are C^* schemes with a truncated public key [PGC98]

Construction of a C^{*-} scheme

(n, θ, r) are the parameters of the scheme

1. Generate a C^* with parameters (n, θ) : $P(x) = x^{1+q^\theta}$
2. Remove the last r polynomials from the public key

$$T \circ P \circ S = \begin{cases} \boldsymbol{p}_1(x_1, \ldots, x_n) \\ \vdots \\ \vdots \\ \boldsymbol{p}_n(x_1, \ldots, x_n) \end{cases} \overset{\Pi}{\longmapsto} \begin{cases} \boldsymbol{p}_1(x_1, \ldots, x_n) \\ \vdots \\ \boldsymbol{p}_{n-r}(x_1, .., x_n) \end{cases} = \Pi \circ \boldsymbol{P}$$

Signing with a C^{*-} scheme

1. Append r random bits μ to the message m to be signed
2. Find a preimage σ of (m, μ) by $T \circ P \circ S$ using S, T
3. Such a preimage always exists since a C^* monomial is bijective
4. σ is a valid signature since $\Pi \circ \boldsymbol{P}(\sigma) = m$

Choosing Parameters

Parameters (n, θ) must define a bijective C^*

$$P(x) = x^{1+q^\theta}$$

- P is bijective when $\gcd(q^\theta + 1, q^n - 1) = 1$ (q even)
- This condition is equivalent to n/d odd where $d = \gcd(n, \theta)$

Choosing Parameters

Parameters (n, θ) must define a bijective C^*

$$P(x) = x^{1+q^\theta}$$

- P is bijective when $\gcd(q^\theta + 1, q^n - 1) = 1$ (q even)
- This condition is equivalent to n/d odd where $d = \gcd(n, \theta)$

$q^r \geq 2^{80}$ to avoid a possible recomposing attack from [PGC98]

Proposed Instantiations

The first version of SFLASH was a tweaked C^{*-} scheme

- S, T taken over $\mathbb{F}_2$ rather than $\mathbb{F}_q$ to make the key smaller
- This specificity could be exploited for an attack [GM02]

Proposed Instantiations

The first version of SFLASH was a tweaked C^{*-} scheme

- S, T taken over $\mathbb{F}_2$ rather than $\mathbb{F}_q$ to make the key smaller
- This specificity could be exploited for an attack [GM02]

Standard Instantiations

	q	n	θ	d	r	Length	PubKey Size
FLASH	2^8	29	11	1	11	296 bits	18 Ko
SFLASHv2 [NESSIE]	2^7	37	11	1	11	259 bits	15 Ko
SFLASHv3	2^7	67	33	1	11	469 bits	112 Ko

Basic Strategy of our Attack

Basic Strategy of our Attack

Important observation

- Consider a C^* public key $\boldsymbol{P} = T \circ P \circ S$

$$\begin{bmatrix} \boldsymbol{p}_1 \\ \vdots \\ \vdots \\ \boldsymbol{p}_n \end{bmatrix} = \begin{bmatrix} t_{11} & \dots & t_{1n} \\ \vdots & & \vdots \\ \vdots & & \vdots \\ t_{n1} & \dots & t_{nn} \end{bmatrix} \begin{bmatrix} (P \circ S)_1 \\ \vdots \\ \vdots \\ (P \circ S)_n \end{bmatrix}$$

Basic Strategy of our Attack

Important observation

- Consider a C^* public key $\boldsymbol{P} = T \circ P \circ S$

$$\begin{bmatrix} \boldsymbol{p}_1 \\ \vdots \\ \vdots \\ \boldsymbol{p}_n \end{bmatrix} = \begin{bmatrix} t_{11} & \dots & t_{1n} \\ \vdots & & \vdots \\ \vdots & & \vdots \\ t_{n1} & \dots & t_{nn} \end{bmatrix} \begin{bmatrix} (P \circ S)_1 \\ \vdots \\ \vdots \\ (P \circ S)_n \end{bmatrix}$$

- The C^{*-} public key $\Pi \circ \boldsymbol{P}$ consists of the $n - r$ first rows

$$\begin{bmatrix} \boldsymbol{p}_1 \\ \vdots \\ \boldsymbol{p}_{n-r} \end{bmatrix} = \begin{bmatrix} t_{11} & \dots & t_{1n} \\ \vdots & & \vdots \\ t_{n-r,1} & \dots & t_{n-r,n} \end{bmatrix} \begin{bmatrix} (P \circ S)_1 \\ \vdots \\ \vdots \\ (P \circ S)_n \end{bmatrix}$$

Important observation 2

- If we could regenerate r *new* linear combinations

$$\begin{bmatrix} \boldsymbol{p}'_1 \\ \vdots \\ \boldsymbol{p}'_r \end{bmatrix} = \begin{bmatrix} t'_{11} & \dots & t'_{1n} \\ \vdots & & \vdots \\ t'_{r1} & \dots & t'_{rn} \end{bmatrix} \begin{bmatrix} (P \circ S)_1 \\ \vdots \\ (P \circ S)_n \end{bmatrix}$$

Important observation 2

- If we could regenerate r *new* linear combinations

$$\begin{bmatrix} \boldsymbol{p}'_1 \\ \vdots \\ \boldsymbol{p}'_r \end{bmatrix} = \begin{bmatrix} t'_{11} & \dots & t'_{1n} \\ \vdots & & \vdots \\ t'_{r1} & \dots & t'_{rn} \end{bmatrix} \begin{bmatrix} (P \circ S)_1 \\ \vdots \\ (P \circ S)_n \end{bmatrix}$$

- then adding them to $\Pi \circ \boldsymbol{P}$ might complete a full C^* key :

$$\boldsymbol{P}' = \begin{bmatrix} \boldsymbol{p}_1 \\ \vdots \\ \boldsymbol{p}_{n-r} \\ \boldsymbol{p}'_1 \\ \vdots \\ \boldsymbol{p}'_r \end{bmatrix} = \begin{bmatrix} t_{11} & \dots & t_{1n} \\ \vdots & & \vdots \\ t_{n-r,1} & & t_{n-r,n} \\ t'_{11} & \dots & t'_{1n} \\ \vdots & & \vdots \\ t'_{r1} & \dots & t'_{rn} \end{bmatrix} \begin{bmatrix} (P \circ S)_1 \\ \vdots \\ \vdots \\ (P \circ S)_n \end{bmatrix}$$

Important observation 3

- This C^* public key $\boldsymbol{P}'$ coincides with the original one $\boldsymbol{P}$ on the first $n-r$ coordinates :

$$\Pi \circ \boldsymbol{P}' = \Pi \circ \boldsymbol{P}$$

Important observation 3

- This C^* public key $\boldsymbol{P'}$ coincides with the original one $\boldsymbol{P}$ on the first $n - r$ coordinates :

$$\Pi \circ \boldsymbol{P'} = \Pi \circ \boldsymbol{P}$$

- We can find preimages by $\Pi \circ \boldsymbol{P}$ by inverting $\boldsymbol{P'}$ using Patarin's attack

Important observation 3

- This C^* public key $\boldsymbol{P}'$ coincides with the original one $\boldsymbol{P}$ on the first $n-r$ coordinates :

$$\Pi \circ \boldsymbol{P}' = \Pi \circ \boldsymbol{P}$$

- We can find preimages by $\Pi \circ \boldsymbol{P}$ by inverting $\boldsymbol{P}'$ using Patarin's attack

Goal

Find a way to generate new linear combinations of the hidden function $P \circ S$

Basic Strategy 2

A recomposing attack through injection of *commuting* maps !

We look for pairs of linear maps (M, N) "commuting" with the internal function :

$$P \circ M = N \circ P$$

Basic Strategy 2

A recomposing attack through injection of *commuting* maps !

We look for pairs of linear maps (M, N) "commuting" with the internal function :

$$P \circ M = N \circ P$$

Then, composing $\Pi \circ \boldsymbol{P}$ with the conjugate of M

$$\boldsymbol{M} = S^{-1} \circ M \circ S$$

generates new linear combinations :

$$\begin{aligned}(\Pi \circ T \circ P \circ S) \circ \boldsymbol{M} &= \Pi \circ T \circ (P \circ M) \circ S \\ &= \Pi \circ T \circ (N \circ P) \circ S \\ &= (\Pi \circ T \circ N) \circ P \circ S\end{aligned}$$

In C^*, P is multiplicative and $M_\xi : x \mapsto \xi.x$ are commuting maps.

$$P \circ M_\xi = M_{P(\xi)} \circ P$$

In C^*, P is multiplicative and $M_\xi : x \mapsto \xi.x$ are commuting maps.

$$P \circ M_\xi = M_{P(\xi)} \circ P$$

Goal

Find a way to discover some maps $\boldsymbol{M}_\xi$ conjugates of M_ξ :

$$\boldsymbol{M}_\xi = S^{-1} \circ M_\xi \circ S$$

The Differential of C^*

FGS05 : Differential Cryptanalysis for Multivariate Schemes

The differential of a quadratic function P is :

$$DP(a, x) = P(a + x) - P(x) - P(a) + P(0)$$

- DP is bilinear and symmetric in (a, x)
- If $\boldsymbol{P} = T \circ P \circ S$ then $D\boldsymbol{P} = T \circ DP(S, S)$

The Differential of C^*

FGS05 : Differential Cryptanalysis for Multivariate Schemes

The differential of a quadratic function P is :

$$DP(a,x) = P(a+x) - P(x) - P(a) + P(0)$$

- DP is bilinear and symmetric in (a,x)
- If $\boldsymbol{P} = T \circ P \circ S$ then $D\boldsymbol{P} = T \circ DP(S,S)$

The differential of a C^* monomial

$$DP(a,x) = a^{q^\theta} x + a x^{q^\theta} = a^{q^\theta+1}\left(\frac{x}{a}\right) + a^{q^\theta+1}\left(\frac{x}{a}\right)^{q^\theta}$$

Letting $L(\xi) = \xi + \xi^{q^\theta}$, we have :

$$DP(a,x) = P(a).L\left(\frac{x}{a}\right)$$

The Differential of C^*

Notable Consequence

- For any element ξ in $\ker(L)$,

$$DP(a, \xi.a) = P(a).L\left(\frac{\xi.a}{a}\right) = P(a).L(\xi) = 0$$

The Differential of C^*

Notable Consequence

- For any element ξ in $\ker(L)$,

$$DP(a, \xi.a) = P(a).L\left(\frac{\xi.a}{a}\right) = P(a).L(\xi) = 0$$

- Therefore, the maps M_ξ with ξ in $\ker(L)$ are the solutions of the *linear* functional equation :

$$DP(a, M(a)) = 0$$

The Differential of C^*

Notable Consequence

- For any element ξ in $\ker(L)$,

$$DP(a, \xi.a) = P(a).L\left(\frac{\xi.a}{a}\right) = P(a).L(\xi) = 0$$

- Therefore, the maps M_ξ with ξ in $\ker(L)$ are the solutions of the *linear* functional equation :

$$DP(a, M(a)) = 0$$

- Considering the differential of this equation, these maps satisfy

$$DP(a, M(x)) + DP(M(a), x) = 0$$

M_ξ with ξ in $\ker(L)$ are the *skew-symmetric maps* w.r.t DP.

Skew-symmetric Maps w.r.t the Diff. of the C^* Monomial

The kernel of $L(\xi) = \xi + \xi^{q^\theta}$

- The non-zero elements of $\ker(L)$ satisfy : $\xi^{q^\theta - 1} = 1$
- There are $\gcd(q^\theta - 1, q^n - 1) = q^d - 1$ such elements
- Therefore, *ker*(L) is a linear subspace of dimension d

Skew-symmetric Maps w.r.t the Diff. of the C^* Monomial

The kernel of $L(\xi) = \xi + \xi^{q^\theta}$

- The non-zero elements of $\ker(L)$ satisfy : $\xi^{q^\theta - 1} = 1$
- There are $\gcd(q^\theta - 1, q^n - 1) = q^d - 1$ such elements
- Therefore, *$ker(L)$ is a linear subspace of dimension d*

Skew-symmetric Maps w.r.t the Diff. of the C^* Monomial

- These maps are **multiplications** M_ξ
- They are the solutions of the **linear** equation

$$DP(a, M(x)) + DP(M(a), x) = 0$$

- They form a subspace of dimension $d = \gcd(n, \theta)$.

Skew-symmetric Maps w.r.t the Diff. of the C^* Monomial

The kernel of $L(\xi) = \xi + \xi^{q^\theta}$

- The non-zero elements of $\ker(L)$ satisfy : $\xi^{q^\theta - 1} = 1$
- There are $\gcd(q^\theta - 1, q^n - 1) = q^d - 1$ such elements
- Therefore, *$ker(L)$ is a linear subspace of dimension d*

Skew-symmetric Maps w.r.t the Diff. of the C^* Monomial

- These maps are **multiplications** M_ξ
- They are the solutions of the **linear** equation

$$DP(a, M(x)) + DP(M(a), x) = 0$$

- They form a subspace of dimension $d = \gcd(n, \theta)$.
- This subspace is non-trivial when $d > 1$, since scalar multiples of the identity are useless.

Skew-symmetric Maps w.r.t the Diff. of the C^* Pub.Key

- They are the solutions of the linear equation :

$$D\boldsymbol{P}(\boldsymbol{M}(a), x) + D\boldsymbol{P}(a, \boldsymbol{M}(x)) = 0 \qquad (1)$$

where

$$D\boldsymbol{P} = T \circ DP(S, S)$$

- Therefore, those are :

$$\boldsymbol{M}_\xi = S^{-1} \circ M_\xi \circ S \qquad \text{where} \quad M_\xi(x) = \xi.x \text{ and } \xi \in \ker(L)$$

- Equation (1) : $\simeq n^3$ linear equations in n^2 unknowns over $\mathbb{F}_q$: ($\simeq n^2/2$ lin.indep (a, x) and n coord. of $D\boldsymbol{P}$)

Skew-symmetric Maps w.r.t the Diff. of the C^* Pub.Key

- They are the solutions of the linear equation :

$$D\boldsymbol{P}(\boldsymbol{M}(a), x) + D\boldsymbol{P}(a, \boldsymbol{M}(x)) = 0 \tag{1}$$

where

$$D\boldsymbol{P} = T \circ DP(S, S)$$

- Therefore, those are :

$$\boldsymbol{M}_\xi = S^{-1} \circ M_\xi \circ S \qquad \text{where} \quad M_\xi(x) = \xi.x \text{ and } \xi \in \ker(L)$$

- Equation (1) : $\simeq n^3$ linear equations in n^2 unknowns over $\mathbb{F}_q$: ($\simeq n^2/2$ lin.indep (a, x) and n coord. of $D\boldsymbol{P}$)

  We might not need all coordinates of $\boldsymbol{P}$ to recover the $\boldsymbol{M}_\xi$!

- If we are only given the first $n-r$ coordinates of $\boldsymbol{P}$:

$$\Pi \circ D\boldsymbol{P}(\boldsymbol{M}(a), x) + \Pi \circ D\boldsymbol{P}(a, \boldsymbol{M}(x)) = 0$$

gives $(n-r)n(n-1)/2$ linear equations in n^2 unknowns

- If we are only given the first $n - r$ coordinates of $\boldsymbol{P}$:

$$\Pi \circ D\boldsymbol{P}(\boldsymbol{M}(a), x) + \Pi \circ D\boldsymbol{P}(a, \boldsymbol{M}(x)) = 0$$

  gives $(n - r)n(n - 1)/2$ linear equations in n^2 unknowns
- The skew-symmetric maps $\boldsymbol{M}_\xi$ are solutions.

- If we are only given the first $n - r$ coordinates of $\boldsymbol{P}$:

$$\Pi \circ D\boldsymbol{P}(\boldsymbol{M}(a), x) + \Pi \circ D\boldsymbol{P}(a, \boldsymbol{M}(x)) = 0$$

gives $(n - r)n(n - 1)/2$ linear equations in n^2 unknowns

- The skew-symmetric maps $\boldsymbol{M}_\xi$ are solutions.
- We expect no other solutions when :

$$(n - r)\frac{n(n - 1)}{2} \geq n^2 - d$$

- If we are only given the first $n - r$ coordinates of $\boldsymbol{P}$:

$$\Pi \circ D\boldsymbol{P}(\boldsymbol{M}(a), x) + \Pi \circ D\boldsymbol{P}(a, \boldsymbol{M}(x)) = 0$$

  gives $(n - r)n(n - 1)/2$ linear equations in n^2 unknowns
- The skew-symmetric maps $\boldsymbol{M}_\xi$ are solutions.
- We expect no other solutions when :

$$(n - r)\frac{n(n-1)}{2} \geq n^2 - d$$

- Hence, heuristically, the $\boldsymbol{M}_\xi$ are the only solutions up to :

$$r^*_{max} = n - \left\lceil 2\frac{n^2 - d}{n(n-1)} \right\rceil = n - 3$$

- The actual value r_{max} is very close to the heuristical r^*_{max} :

n	36	36	38	39	39	40	42	42	44
θ	8	12	10	13	9	8	12	14	12
d	4	12	2	13	3	8	6	14	4
r^*_{max}	33	33	35	36	36	37	39	39	41
r_{max}	33	32	35	35	36	37	39	38	41

- The actual value r_{max} is very close to the heuristical r^*_{max} :

n	36	36	38	39	39	40	42	42	44
θ	8	12	10	13	9	8	12	14	12
d	4	12	2	13	3	8	6	14	4
r^*_{max}	33	33	35	36	36	37	39	39	41
r_{max}	33	32	35	35	36	37	39	38	41

The skew-symmetric maps can be recovered from as few as 3 or 4 coordinates of the public key !

Recovering a Full C^* Public Key

Using a single non-trivial $\boldsymbol{M}_\xi$, up to $r = n/2$

1. We complete $\Pi \circ \boldsymbol{P}$ using r coordinates of $\Pi \circ \boldsymbol{P} \circ \boldsymbol{M}_\xi$.

$$\left\{ \begin{array}{l} \Pi \circ \boldsymbol{P} \\ (\Pi \circ \boldsymbol{P} \circ \boldsymbol{M}_\xi)_{1 \to r} \end{array} \right. = \left[\begin{array}{l} \Pi \circ T \\ (\Pi \circ T \circ M_{P(\xi)})_{1 \to r} \end{array} \right] \circ P \circ S$$

2. We can check that this is a full C^* public key since Patarin's attack works again.

n	36	36	38	39	39	40	42	42	44
θ	8	12	10	13	9	8	12	14	12
d	4	12	2	13	3	8	6	14	4
r	11	11	11	12	12	12	13	13	13
$C^{*-} \mapsto C^*$	$57s$	$57s$	$94s$	$105s$	$90s$	$105s$	$141s$	$155s$	$155s$

Note : parameters are close to those of SFLASHv2, with the same $q = 2^7$.

Recovering a Full C^* Public Key

Using a whole basis of $\boldsymbol{M}_\xi$

Since we have $d(n-r)$ coordinates available, the overall bound is :

$$r \leq n\left(1 - \frac{1}{d}\right)$$

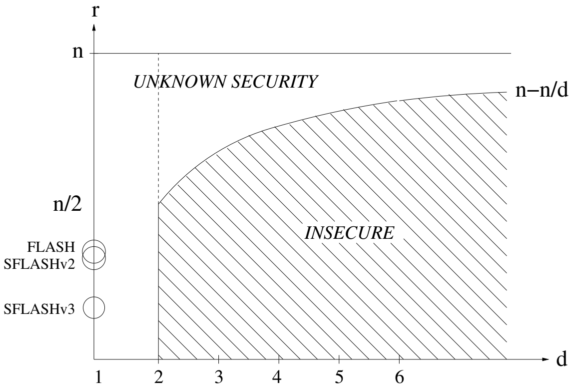
n	36	36	38	39	39	40	42	42	44
θ	8	12	10	13	9	8	12	14	12
d	4	12	2	13	3	8	6	14	4
r	27	32	19	35	26	35	35	38	33
$C^{*-} \mapsto C^*$	$65s$	$51s$	$112s$	$79s$	$107s$	$95s$	$134s$	$117s$	$202s$

Conclusion

- C^{*-} schemes with $d > 1$ are insecure up to $r = n(1 - \frac{1}{d})$

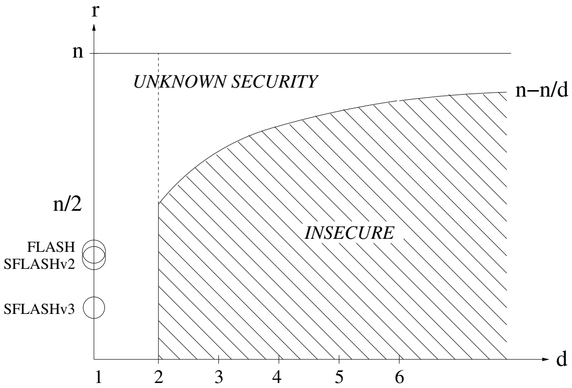
Conclusion

- C^{*-} schemes with $d > 1$ are insecure up to $r = n(1 - \frac{1}{d})$



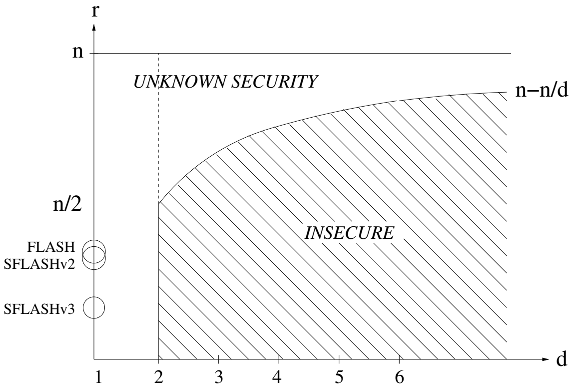
Conclusion

- C^{*-} schemes with $d > 1$ are insecure up to $r = n(1 - \frac{1}{d})$
- The attack does not apply to the case $d = 1$



Conclusion

- C^{*-} schemes with $d > 1$ are insecure up to $r = n(1 - \frac{1}{d})$
- The attack does not apply to the case $d = 1$ (but a different way to find multiplications breaks these cases : see Crypto07, joint work with Adi Shamir)



Thank you for your attention !

Questions ?