ZAMA

BALANCED NON-ADJACENT FORMS

ASIACRYPT 2021 • December 6–10, 2021

Marc Joye

THE CLOUD NEEDS BETTER DATA SECURITY

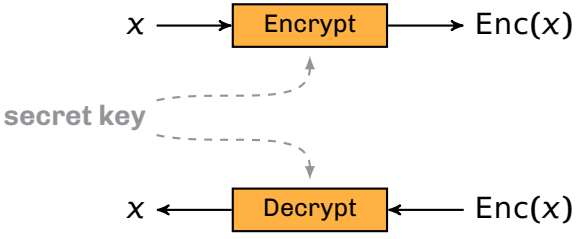
Even the best companies sometimes make mistakes

Research

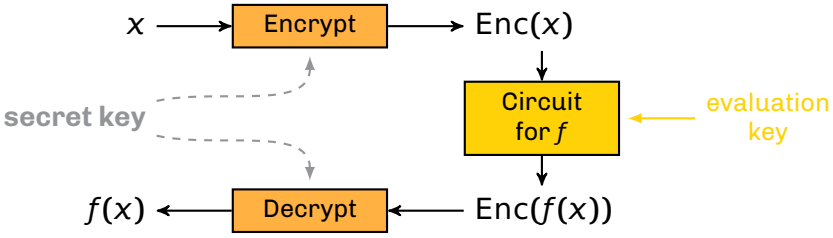
ChaosDB: How we hacked thousands of Azure customers' databases

August 26, 2021
Nir Ohfeld and Sagi Tzadik

FULLY HOMOMORPHIC ENCRYPTION



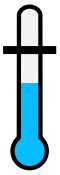
FULLY HOMOMORPHIC ENCRYPTION



Remark: Any private-key FHE scheme can easily be turned into a public-key FHE scheme

CONTROLLING THE NOISE

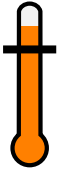
Enc(x)



✔

valid ciphertext

Enc(x)



✖

incorrect decryption

Noise accumulates over time

ILLUSTRATION

Suppose we want to compute $c \leftarrow \mathsf{Enc}(k \cdot x)$ for some $k \in \mathbb{Z}$, $|k| < B^n$, where Enc is a homomorphic encryption scheme

1. **First approach**
   - obtain $\mathsf{Enc}(x)$
   - compute $c \leftarrow k \cdot \mathsf{Enc}(x)$

ILLUSTRATION

Suppose we want to compute $c \leftarrow \mathsf{Enc}(k \cdot x)$ for some $k \in \mathbb{Z}$, $|k| < B^n$, where Enc is a homomorphic encryption scheme

1. **First approach**
   - obtain $\mathsf{Enc}(x)$
   - compute $c \leftarrow k \cdot \mathsf{Enc}(x)$

2. **Second approach**
   - write $k = \sum_{i=0}^{n} k'_i B^i$
   - obtain pre-computed ciphertexts $\mathsf{Enc}(B^i x)$
   - compute $c \leftarrow \sum_{i=0}^{n} k'_i \cdot \mathsf{Enc}(B^i x)$

ILLUSTRATION

Suppose we want to compute $c \leftarrow \mathsf{Enc}(k \cdot x)$ for some $k \in \mathbb{Z}$, $|k| < B^n$, where Enc is a homomorphic encryption scheme

1. **First approach**
   - obtain $\mathsf{Enc}(x)$
   - compute $c \leftarrow k \cdot \mathsf{Enc}(x)$

   Bound on noise variance: $\mathbb{E}[k^2]\,\sigma^2$

2. **Second approach**
   - write $k = \sum_{i=0}^{n} k'_i B^i$
   - obtain pre-computed ciphertexts $\mathsf{Enc}(B^i x)$
   - compute $c \leftarrow \sum_{i=0}^{n} k'_i \cdot \mathsf{Enc}(B^i x)$

   Bound on noise variance: $\left(\sum_{i=0}^{n} \mathbb{E}[k'^2_i]\right)\sigma^2$

BALANCED NON-ADJACENT FORMS

Problem statement

Given an integer k and a radix B, decompose

$$k = \sum_{i=0}^{n} k'_i B^i \quad \text{with } k'_i \in \{-(B-1), \ldots, B-1\}$$

such that the Euclidean weight $\sum_{i=0}^{n} {k'_i}^2$ is minimal

Notes:

- $B = 2 \rightsquigarrow$ regular NAFs
- B odd $\rightsquigarrow$ balanced forms; i.e., $k'_i \in \{-(B-1)/2, \ldots, (B-1)/2\}$

A USEFUL OBSERVATION

- Let $k_j \in \{-B/2, \ldots, B/2\}$

$$(\ldots, k'_{j+1}, k'_j) \equiv (\ldots, k'_{j+1} + \text{sign}(k'_j), -k'_j)$$

Example

- $B = 4$
  - $10 = (2, 2)_4$
  - $10 = (1, -2, 2)_4$
  - $10 = (1, -1, -2)_4$ [minimal form]
- Heuristic $(2, *)$ or $(-2, *) \implies$ always flip

BNAF RECODING

Input: Integer $k \neq 0$
Output: BNAF$(k) \leftarrow (k'_n, \ldots, k'_0)$ with $k'_i \in \{-\lfloor \frac{B}{2} \rfloor, \ldots, \lfloor \frac{B}{2} \rfloor\}$

$K \leftarrow k$; $i \leftarrow 0$
while $(K \neq 0)$ **do**
  $k'_i \leftarrow K \bmod B$; $K \leftarrow (K - k'_i)/B$
  if $(k'_i > \lfloor \frac{B}{2} \rfloor) \vee \left((k'_i = \lceil \frac{B}{2} \rceil) \wedge ((K \bmod B) \geq \lfloor \frac{B}{2} \rfloor)\right)$ **then**
    $k'_i \leftarrow k'_i - B$; $K \leftarrow K + 1$
  end if
  $i \leftarrow i + 1$
end while

return $(k'_{i-1}, \ldots, k'_0)$

BNAF RECODING

Input: Integer $k \neq 0$
Output: BNAF$(k) \leftarrow (k'_n, \ldots, k'_0)$ with $k'_i \in \{-\lfloor \frac{B}{2} \rfloor, \ldots, \lfloor \frac{B}{2} \rfloor\}$

$K \leftarrow k$; $i \leftarrow 0$
while $(K \neq 0)$ **do**
  $k'_i \leftarrow K \bmod B$; $K \leftarrow (K - k'_i)/B$
  if $(k'_i > \lfloor \frac{B}{2} \rfloor) \vee ((k'_i = \lceil \frac{B}{2} \rceil) \wedge ((K \bmod B) \geq \lfloor \frac{B}{2} \rfloor))$ **then**
    $k'_i \leftarrow k'_i - B$; $K \leftarrow K + 1$
  end if
  $i \leftarrow i + 1$
end while

return $(k'_{i-1}, \ldots, k'_0)$

❶ ***B* odd**
⇝ balanced forms

BNAF RECODING

Input: Integer $k \neq 0$
Output: BNAF$(k) \leftarrow (k'_n, \ldots, k'_0)$ with $k'_i \in \{-\lfloor \frac{B}{2} \rfloor, \ldots, \lfloor \frac{B}{2} \rfloor\}$

$K \leftarrow k$; $i \leftarrow 0$
while $(K \neq 0)$ **do**
  $k'_i \leftarrow K \bmod B$; $K \leftarrow (K - k'_i)/B$
  if $(k'_i > \lfloor \frac{B}{2} \rfloor) \vee \left((k'_i = \lceil \frac{B}{2} \rceil) \wedge ((K \bmod B) \geq \lfloor \frac{B}{2} \rfloor)\right)$ **then**
    $k'_i \leftarrow k'_i - B$; $K \leftarrow K + 1$
  end if
  $i \leftarrow i + 1$
end while

return $(k'_{i-1}, \ldots, k'_0)$

❷ $B = 2$
⇝ regular NAFs

BNAF RECODING

Input: Integer $k \neq 0$
Output: BNAF$(k) \leftarrow (k'_n, \ldots, k'_0)$ with $k'_i \in \{-\lfloor \frac{B}{2} \rfloor, \ldots, \lfloor \frac{B}{2} \rfloor\}$

$K \leftarrow k$; $i \leftarrow 0$
while $(K \neq 0)$ **do**
  $k'_i \leftarrow K \bmod B$; $K \leftarrow (K - k'_i)/B$
  if $(k'_i > \lfloor \frac{B}{2} \rfloor) \vee ((k'_i = \lceil \frac{B}{2} \rceil) \wedge ((K \bmod B) \geq \lfloor \frac{B}{2} \rfloor))$ **then**
    $k'_i \leftarrow k'_i - B$; $K \leftarrow K + 1$
  end if
  $i \leftarrow i + 1$
end while

return $(k'_{i-1}, \ldots, k'_0)$

❸ ***B* even**
⇝ BNAFs

BNAF RECODING

Input: Integer $k \neq 0$
Output: BNAF$(k) \leftarrow (k'_n, \ldots, k'_0)$ with $k'_i \in \{-\lfloor \frac{B}{2} \rfloor, \ldots, \lfloor \frac{B}{2} \rfloor\}$

$K \leftarrow k$; $i \leftarrow 0$
while $(K \neq 0)$ **do**
  $k'_i \leftarrow K \bmod B$; $K \leftarrow (K - k'_i)/B$
  if $(k'_i > \lfloor \frac{B}{2} \rfloor) \vee \left((k'_i = \lceil \frac{B}{2} \rceil) \wedge ((K \bmod B) \geq \lfloor \frac{B}{2} \rfloor)\right)$ **then**
    $k'_i \leftarrow k'_i - B$; $K \leftarrow K + 1$
  end if
  $i \leftarrow i + 1$
end while

return $(k'_{i-1}, \ldots, k'_0)$

MAIN RESULT

Theorem (Existence & Uniqueness)

Every integer has a unique BNAF

Theorem (Optimality)

BNAF has minimal Euclidean weight among M-R representations

DIGIT DISTRIBUTION

Proposition (Uniformly random BNAF)

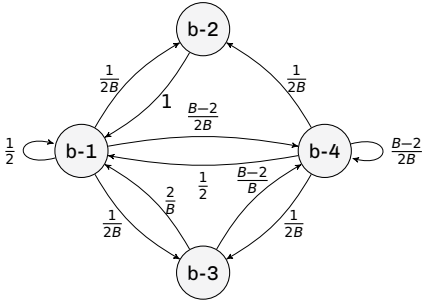
1. *B even:*

$$\Pr[k'_i = d] = \begin{cases} \frac{B+2}{B+1}\frac{1}{B} & \text{if } d = 0 \\ \frac{1}{2(B+1)} & \text{if } d \in \{-\lfloor \frac{B}{2} \rfloor, \lfloor \frac{B}{2} \rfloor\} \\ \frac{1}{B} & \text{otherwise} \end{cases}$$

2. *B odd:* $\Pr[k'_i = d] = \frac{1}{B}$

Corollary

- $\mathbb{E}[k'_i] = 0$
- $\mathrm{Var}(k'_i) = \begin{cases} \frac{1}{12}(B^2-1) & \text{if } B \text{ is odd} \\ \frac{1}{12}\frac{(B+2)(B^2-B+1)}{B+1} & \text{if } B \text{ is even} \end{cases}$



(Exact distribution for *n*-digit integers also given)

EXTENSION

- BNAF recoding naturally extends to modular integers

$$\text{BNAF} \pmod{B^n} = \begin{cases} \text{BNAF}(k \bmod B^n) & \text{if } k \bmod B^n \leq \lfloor \frac{B^n}{2} \rfloor \\ \text{BNAF}\big((k \bmod B^n) - B^n\big) & \text{otherwise} \end{cases}$$

- BNAF modulo B^n exists and is unique
  - …unless B is even and leading recoded digit is $\pm B/2$, in which case, there are two BNAFs: $(\pm k'_{n-1}, k'_{n-2}, \ldots, k'_0)$
- BNAF modulo B^n has minimal Euclidean weight

SOME APPLICATIONS

Gadget decomposition Using 'gadget' vector $\vec{g} = (1, B, \ldots, B^{\ell-1})$, one can define inverse transformation g^{-1} such that, for any scalar k, $g^{-1}(k) \cdot \vec{g}^{\intercal} = k$ and $g^{-1}(k)$ is small

$$g^{-1}(k) = (k_0, \ldots, k_{\ell-1}) \text{ and } k \equiv \sum_{i=0}^{\ell-1} k_i B^i \pmod{q}$$

$\rightsquigarrow$ select the BNAF decomposition for g^{-1}

SOME APPLICATIONS

Gadget decomposition Using 'gadget' vector $\vec{g} = (1, B, \ldots, B^{\ell-1})$, one can define inverse transformation g^{-1} such that, for any scalar k, $g^{-1}(k) \cdot \vec{g}^{\intercal} = k$ and $g^{-1}(k)$ is small

$$g^{-1}(k) = (k_0, \ldots, k_{\ell-1}) \text{ and } k \equiv \sum_{i=0}^{\ell-1} k_i B^i \pmod{q}$$

⇝ select the BNAF decomposition for g^{-1}

Key switching Key switching enables converting ciphertexts into ciphertexts under another key in different parameter sets using key-switching keys $ksk[j] \leftarrow \widehat{LWE}_{\vec{s}'}(s_j)$ ⇝ BNAFs limit the noise

SOME APPLICATIONS

Gadget decomposition Using 'gadget' vector $\vec{g} = (1, B, \ldots, B^{\ell-1})$, one can define inverse transformation g^{-1} such that, for any scalar k, $g^{-1}(k) \cdot \vec{g}^{\intercal} = k$ and $g^{-1}(k)$ is small

$$g^{-1}(k) = (k_0, \ldots, k_{\ell-1}) \text{ and } k \equiv \sum_{i=0}^{\ell-1} k_i B^i \pmod{q}$$

⇝ select the BNAF decomposition for g^{-1}

Key switching Key switching enables converting ciphertexts into ciphertexts under another key in different parameter sets using key-switching keys $ksk[j] \leftarrow \widehat{LWE}_{\vec{s}'}(s_j)$ ⇝ BNAFs limit the noise

Fast Fourier transform Balanced representations (and thus BNAFs) tend to reduce the convolution errors attendant to floating-point arithmetic

SUMMARY

- BNAF: New minimal M-R form for integers
  - existence and uniqueness*
  - optimality w.r.t. Euclidean weight
  - digit distribution
  - cryptographic applications
- For more information, check out the paper at

  `https://ia.cr/2021/1161`