Computing Low-Weight Discrete Logarithms

Bailey Kacsmar
University of Waterloo

Sarah Plosker
Brandon University

Ryan Henry
Indiana University

Selected Areas in Cryptography (SAC) 2017

This material is based upon work supported by the National Science Foundation under Grant No. CNS-1565375.

Discrete Logarithm Problem (DLP)

Given $(g, h) \in \mathbb{G} \times \mathbb{G}$, find $x \in \mathbb{Z}_q^*$ such that:

$$h = g^x$$

(Here $\mathbb{G}$ is a multiplicative group of prime order q)

Low-Weight Discrete Logarithm Problem

Consider the radix-b representation of x in

$$h = g^{x}$$

$$\mathbf{x} = \mathbf{1010001001}$$

Weight: number of non-zero digits, t
Hamming-weight: weight t, in radix-b, $\mathbf{b} = \mathbf{2}$
Length: number of digits, m

Low-Weight Discrete Logarithm Problem

Consider the radix-b representation of x in

$$h = g^x$$

$$\mathbf{x} = \mathbf{1010001001}$$

Weight: number of non-zero digits, t

Hamming-weight: weight t, in radix-b, $\mathbf{b} = \mathbf{2}$

Length: number of digits, m

Low-Weight Discrete Logarithm Problem

Consider the radix-b representation of x in

$$h = g^x$$

$$\mathbf{x} = \mathbf{1010001001}$$

Weight: number of non-zero digits, t
Hamming-weight: weight t, in radix-b, $\mathbf{b} = \mathbf{2}$
Length: number of digits, m

Low-Weight Discrete Logarithm Problem

Consider the radix-b representation of x in

$$h = g^{x}$$

$$\mathbf{x} = \mathbf{1010001001}$$

Weight: number of non-zero digits, t
Hamming-weight: weight t, in radix-b, $\mathbf{b} = \mathbf{2}$
Length: number of digits, m

Solving the Low-Weight Discrete Logarithm Problem

Find $x \in \mathbb{Z}_q^*$ such that:

$$h = g^x \bmod q$$

Classically, go from, solving in say,

$$q \quad \rightarrow \quad \sqrt{q}$$

For us, go from solving in say,

$$\binom{m}{t} \quad \rightarrow \quad \binom{m/2}{t/2}$$

Outline

- Baby-Step, Giant-Step Algorithms
- Optimizations
- Generalizing to arbitrary bases ($b > 1$)
- Cryptanalytic Application

A Low-Hamming Weight Example in Finding x

Assume, $t = 4$, and $m = 10$, (i.e. $\binom{10}{4}$ possible x's)

Let x=1010001001

Y_1 Y_2

$val(Y_1) = 20 \cdot 2^5$ and $val(Y_2) = 9 \cdot 2^0$

$$x = val(Y_1) + val(Y_2)$$

A Low-Hamming Weight Example in Finding x

Assume, $t = 4$, and $m = 10$, (i.e. $\binom{10}{4}$ possible x's)

Let x=1010001001

Y_1 Y_2

$val(Y_1) = 20 \cdot 2^5$ and $val(Y_2) = 9 \cdot 2^0$

$x = val(Y_1) + val(Y_2)$

A Low-Hamming Weight Example in Finding x

Assume, $t = 4$, and $m = 10$, (i.e. $\binom{10}{4}$ possible x's)

Let $x = 10100 01001$

$Y_1 \quad Y_2$

$val(Y_1) = 20 \cdot 2^5$ and $val(Y_2) = 9 \cdot 2^0$

$$x = val(Y_1) + val(Y_2)$$

A Low-Hamming Weight Example in Finding x

Recall, $x = 1010001001$

$Y_1 \quad Y_2$

From, $g^x = h$,

$$g^{val(Y_1)+val(Y_2)} = h$$

$$h \cdot (g^{-1})^{val(Y_2)} = g^{val(Y_1)}$$

$$\log_g h = val(Y_1) + val(Y_2) \bmod q$$

So, search for Y_1 and Y_2 with weight $t/2$

A Low-Hamming Weight Example in Finding x

Recall, $x = \underbrace{10100}_{Y_1}\underbrace{01001}_{Y_2}$

From, $g^x = h$,

$$g^{val(Y_1)+val(Y_2)} = h$$

$$h \cdot (g^{-1})^{val(Y_2)} = g^{val(Y_1)}$$

$$\log_g h = val(Y_1) + val(Y_2) \bmod q$$

So, search for Y_1 and Y_2 with weight $t/2$

A Low-Hamming Weight Example in Finding x

Recall, $x = \underbrace{10100}_{Y_1}\underbrace{01001}_{Y_2}$

From, $g^x = h$,

$$g^{val(Y_1)+val(Y_2)} = h$$

$$h \cdot (g^{-1})^{val(Y_2)} = g^{val(Y_1)}$$

$$\log_g h = val(Y_1) + val(Y_2) \bmod q$$

So, search for Y_1 and Y_2 with weight $t/2$

Low-Hamming Weight - Basic Algorithm [1]

Assume $t = 4$ and $m = 5$

Giant-Step

Y_1	$g^{val(Y_1)}$
00011	$g^{val(00011)}$
00110	$g^{val(00110)}$
00101	$g^{val(00101)}$
$\vdots$	$\vdots$
10010	$g^{val(10010)}$
10001	$g^{val(10001)}$

Baby-Step

Y_2	$h \cdot g^{-val(Y_2)}$
00011	$h \cdot g^{-val(00011)}$
00110	$h \cdot g^{-val(10010)}$
00101	$h \cdot g^{-val(00101)}$
$\vdots$	$\vdots$
10010	$h \cdot g^{-val(10010)}$
10001	$h \cdot g^{-val(10001)}$

$$x = val(Y_1) + val(Y_2) \qquad \Theta\left(\binom{m}{t/2}\right)$$

[1] Due to Heiman (1992) and Odlyzko (1992)

Low-Hamming Weight - Basic Algorithm [1]

Assume $t = 4$ and $m = 5$

Giant-Step

Y_1	$g^{val(Y_1)}$
00011	$g^{val(00011)}$
00110	$g^{val(00110)}$
00101	$g^{val(00101)}$
$\vdots$	$\vdots$
10010	$g^{val(10010)}$
10001	$g^{val(10001)}$

?

Baby-Step

Y_2	$h \cdot g^{-val(Y_2)}$
00011	$h \cdot g^{-val(00011)}$
00110	$h \cdot g^{-val(10010)}$
00101	$h \cdot g^{-val(00101)}$
$\vdots$	$\vdots$
10010	$h \cdot g^{-val(10010)}$
10001	$h \cdot g^{-val(10001)}$

$$x = val(Y_1) + val(Y_2) \qquad \Theta\left(\binom{m}{t/2}\right)$$

[1] Due to Heiman (1992) and Odlyzko (1992)

Low-Hamming Weight - Basic Algorithm [1]

Assume $t = 4$ and $m = 5$

Giant-Step

Y_1	$g^{val(Y_1)}$
00011	$g^{val(00011)}$
00110	$g^{val(00110)}$
00101	$g^{val(00101)}$
$\vdots$	$\vdots$
10010	$g^{val(10010)}$
10001	$g^{val(10001)}$

?

?

Baby-Step

Y_2	$h \cdot g^{-val(Y_2)}$
00011	$h \cdot g^{-val(00011)}$
00110	$h \cdot g^{-val(10010)}$
00101	$h \cdot g^{-val(00101)}$
$\vdots$	$\vdots$
10010	$h \cdot g^{-val(10010)}$
10001	$h \cdot g^{-val(10001)}$

$$x = val(Y_1) + val(Y_2) \qquad \Theta\left(\binom{m}{t/2}\right)$$

[1] Due to Heiman (1992) and Odlyzko (1992)

Example: Low Hamming Weight Basic Algorithm

Assume $x \in \mathbb{Z}_{31}^{*}$, $t = 4, m = 5, g = 3$ and $h = 11$

Giant-Step

Y_1	$g^{val(Y_1)}$
00011	$g^{val(00011)}$
00110	$g^{val(00110)}$
00101	$g^{val(00101)}$
$\vdots$	$\vdots$
10010	$g^{val(10010)}$
10001	$g^{val(00101)}$

Baby-Step

Y_2	$h \cdot g^{-val(Y_2)}$
00011	$h \cdot g^{-val(00011)}$
00110	$h \cdot g^{-val(00110)}$
00101	$h \cdot g^{-val(00101)}$
$\vdots$	$\vdots$
10010	$h \cdot g^{-val(10010)}$
10001	$h \cdot g^{-val(10001)}$

$$x = val(Y_1) + val(Y_2) = 17 + 6 = 23$$

Example: Low Hamming Weight Basic Algorithm

Assume $x \in \mathbb{Z}_{31}^{*}$, $t = 4, m = 5, g = 3$ and $h = 11$

Giant-Step

Y_1	$g^{val(Y_1)}$
00011	27
00110	16
00101	26
$\vdots$	$\vdots$
10010	4
10001	22

Baby-Step

Y_2	$h \cdot g^{-val(Y_2)}$
00011	$h \cdot g^{-val(00011)}$
00110	$h \cdot g^{-val(00110)}$
00101	$h \cdot g^{-val(00101)}$
$\vdots$	$\vdots$
10010	$h \cdot g^{-val(10010)}$
10001	$h \cdot g^{-val(10001)}$

$$x = val(Y_1) + val(Y_2) = 17 + 6 = 23$$

Example: Low Hamming Weight Basic Algorithm

Assume $x \in \mathbb{Z}_{31}^{*}$, $t = 4$, $m = 5$, $g = 3$ and $h = 11$

Giant-Step

Y_1	$g^{val(Y_1)}$
00011	27
00110	16
00101	26
$\vdots$	$\vdots$
10010	4
10001	

?

Baby-Step

Y_2	$h \cdot g^{-val(Y_2)}$
00011	5
00110	$h \cdot g^{-val(00110)}$
00101	$h \cdot g^{-val(00101)}$
$\vdots$	$\vdots$
10010	$h \cdot g^{-val(10010)}$
10001	$h \cdot g^{-val(10001)}$

$$x = val(Y_1) + val(Y_2) = 17 + 6 = 23$$

Example: Low Hamming Weight Basic Algorithm

Assume $x \in \mathbb{Z}_{31}^*$, $t = 4, m = 5, g = 3$ and $h = 11$

Giant-Step

Y_1	$g^{val(Y_1)}$
00011	27
00110	16
00101	26
$\vdots$	$\vdots$
10010	4
10001	**22**

?

Baby-Step

Y_2	$h \cdot g^{-val(Y_2)}$
00011	5
00110	**22**
00101	$h \cdot g^{-val(00101)}$
$\vdots$	$\vdots$
10010	$h \cdot g^{-val(10010)}$
10001	$h \cdot g^{-val(10001)}$

$$x = val(Y_1) + val(Y_2) = 17 + 6 = 23$$

Optimization - Interleaving

Assume $t = 4, m = 5,$

Y_1	$g^{val(Y_1)}$
00011	$g^{val(00011)}$
00110	$g^{val(00110)}$
00101	$g^{val(00101)}$
$\vdots$	$\vdots$
10010	$g^{val(10010)}$
10001	$g^{val(10001)}$

Y_2	$h \cdot g^{-val(Y_2)}$
10001	$h \cdot g^{-val(10001)}$
10010	$h \cdot g^{-val(10010)}$
10100	$h \cdot g^{-val(10100)}$
$\vdots$	$\vdots$
00110	$h \cdot g^{-val(00110)}$
00011	$h \cdot g^{-val(00011)}$

$$x = val(Y_1) + val(Y_2) = 6 + 17 = 23$$

Optimization - Interleaving

Assume $x \in \mathbb{Z}_{31}^{*}$, $t = 4$, $m = 5$, $g = 3$, and $h = 11$

Y_1	$g^{val(Y_1)}$
00011	$g^{val(00011)}$
00110	$g^{val(00110)}$
00101	$g^{val(00101)}$
$\vdots$	$\vdots$
10010	$g^{val(10010)}$
10001	$g^{val(10001)}$

Y_2	$h \cdot g^{-val(Y_2)}$
10001	$h \cdot g^{-val(10001)}$
10010	$h \cdot g^{-val(10010)}$
10100	$h \cdot g^{-val(10100)}$
$\vdots$	$\vdots$
00110	$h \cdot g^{-val(00110)}$
00011	$h \cdot g^{-val(00011)}$

$$x = val(Y_1) + val(Y_2) = 6 + 17 = 23$$

Optimization - Interleaving

Assume $x \in \mathbb{Z}_{31}^{*}$, $t = 4$, $m = 5$, $g = 3$, and $h = 11$

Y_1	$g^{val(Y_1)}$
00011	27
00110	$g^{val(00110)}$
00101	$g^{val(00101)}$
$\vdots$	$\vdots$
10010	$g^{val(10010)}$
10001	$g^{val(10001)}$

?

Y_2	$h \cdot g^{-val(Y_2)}$
10001	$h \cdot g^{-val(10001)}$
10010	$h \cdot g^{-val(10010)}$
10100	$h \cdot g^{-val(10100)}$
$\vdots$	$\vdots$
00110	$h \cdot g^{-val(00110)}$
00011	$h \cdot g^{-val(00011)}$

$$x = val(Y_1) + val(Y_2) = 6 + 17 = 23$$

Optimization - Interleaving

Assume $x \in \mathbb{Z}_{31}^*$, $t = 4$, $m = 5$, $g = 3$, and $h = 11$

Y_1	$g^{val(Y_1)}$
00011	27
00110	$g^{val(00110)}$
00101	$g^{val(00101)}$
$\vdots$	$\vdots$
10010	$g^{val(10010)}$
10001	$g^{val(10001)}$

?

Y_2	$h \cdot g^{-val(Y_2)}$
10001	16
10010	$h \cdot g^{-val(10010)}$
10100	$h \cdot g^{-val(10100)}$
$\vdots$	$\vdots$
00110	$h \cdot g^{-val(00110)}$
00011	$h \cdot g^{-val(00011)}$

$$x = val(Y_1) + val(Y_2) = 6 + 17 = 23$$

Optimization - Interleaving

Assume $x \in \mathbb{Z}_{31}^*$, $t = 4$, $m = 5$, $g = 3$, and $h = 11$

Y_1	$g^{val(Y_1)}$
00011	27
00110	16
00101	$g^{val(00101)}$
$\vdots$	$\vdots$
10010	$g^{val(10010)}$
10001	$g^{val(10001)}$

?

?

Y_2	$h \cdot g^{-val(Y_2)}$
10001	16
10010	$h \cdot g^{-val(10010)}$
10100	$h \cdot g^{-val(10100)}$
$\vdots$	$\vdots$
00110	$h \cdot g^{-val(00110)}$
00011	$h \cdot g^{-val(00011)}$

$$x = val(Y_1) + val(Y_2) = 6 + 17 = 23$$

Interleaved Low-Hamming Weight Algorithm

In the worst case, a collision is found in

$$\binom{m}{t/2} \quad \longrightarrow \quad \binom{m - t/2}{t/2}$$

Heuristically, we can expect a collision after around

$$\binom{m/2}{t/2}$$

Recall Example: Low Hamming Weight

Giant-Step

Y_1	$g^{val(Y_1)}$
00011	27
00110	16
00101	26
$\vdots$	$\vdots$
10010	4
10001	22

Baby-Step

Y_2	$h \cdot g^{-val(Y_2)}$
00011	5
00110	22
00101	4
$\vdots$	$\vdots$
10010	26
10001	16

$$x = val(Y_1) + val(Y_2)$$

More then one pair of valid values Y_1, Y_2

Coppersmith's Deterministic Algorithm

Weight $t = 10$ and length $m = 22$

$$x = 1\,1\,0\,0\,1\,0\,1\,0\,1\,1\,1\,0\,1\,1\,0\,0\,0\,0\,0\,0\,1\,0$$

Baby-Step Giant-Step

Y_1: 5 ones, 6 zeroes and Y_2: 5 ones, 6 zeroes

If no collision is found, shift the set cyclically by one

Guaranteed that such (disjoint) subsets Y_1, Y_2 exist

$$m\binom{m/2}{t/2}$$

Coppersmith's Deterministic Algorithm

Weight $t = 10$ and length $m = 22$

$$x = 1\,1\,0\,0\,1\,0\,1\,0\,1\,1\,1\,0\,1\,1\,0\,0\,0\,0\,0\,0\,1\,0$$

Baby-Step Giant-Step

Y_1: 5 ones, 6 zeroes and Y_2: 5 ones, 6 zeroes

If no collision is found, shift the set cyclically by one

Guaranteed that such (disjoint) subsets Y_1, Y_2 exist

$$m\binom{m/2}{t/2}$$

Coppersmith's Deterministic Algorithm

Weight $t = 10$ and length $m = 22$

$$x = \; 1\,1\,0\,0\,1\,0\,1\,0\,1\,1\,1\,0\,1\,1\,0\,0\,0\,0\,0\,0\,1\,0$$

Baby-Step Giant-Step

Y_1: 5 ones, 6 zeroes and Y_2: 5 ones, 6 zeroes

If no collision is found, shift the set cyclically by one

Guaranteed that such (disjoint) subsets Y_1, Y_2 exist

$$m\binom{m/2}{t/2}$$

Coppersmith's Deterministic Algorithm

Weight $t = 10$ and length $m = 22$

$x =$ 1 1 0 0 1 0 1 0 1 1 1 0 1 1 0 0 0 0 0 0 1 0

Baby-Step Giant-Step

Y_1: 5 ones, 6 zeroes and Y_2: 5 ones, 6 zeroes

If no collision is found, shift the set cyclically by one

Guaranteed that such (disjoint) subsets Y_1, Y_2 exist

$$m\binom{m/2}{t/2}$$

Coppersmith's Deterministic Algorithm

Weight $t = 10$ and length $m = 22$

$x =$ 1 1 0 0 1 0 1 0 1 1 1 0 1 1 0 0 0 0 0 0 1 0

Baby-Step Giant-Step

Y_1: 5 ones, 6 zeroes and Y_2: 5 ones, 6 zeroes

If no collision is found, shift the set cyclically by one

Guaranteed that such (disjoint) subsets Y_1, Y_2 exist

$$m\binom{m/2}{t/2}$$

Coppersmith's Deterministic Algorithm

Weight $t = 10$ and length $m = 22$

$$x = \quad 1\,1\,0\,0\,1\,0\,1\,0\,1\,1\,1\,0\,1\,1\,0\,0\,0\,0\,0\,0\,1\,0$$

Baby-Step Giant-Step

Y_1: 5 ones, 6 zeroes and Y_2: 5 ones, 6 zeroes

If no collision is found, shift the set cyclically by one

Guaranteed that such (disjoint) subsets Y_1, Y_2 exist

$$m\binom{m/2}{t/2}$$

Optimization - Pascal's Lemma

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

Which means that,

$$\binom{m/2}{t/2} = \binom{m/2-1}{t/2-1} + \binom{m/2-1}{t/2}$$

Implications of Pascal's Lemma

Reduce all but the first iteration of the loop by using

$$\binom{m/2}{t/2} = \binom{m/2-1}{t/2-1} + \binom{m/2-1}{t/2}$$

Resulting in an improvement from

$$m\binom{m/2}{t/2} \quad \to \quad t\binom{m/2}{t/2}$$

Generalizing to Radix$-b$

Assume, $t = 6, m = 8, b = 4$

Search for Y_1, X_1 and Y_2, X_2 with weight $t/2$

For each Y_1, such as 1011

consider possible $X_2 = 1011$

Meaning, include an inner loop over each $X_i \in [b-1]^{t/2}$

Generalizing to Radix$-b$

Assume, $t = 6, m = 8, b = 4$

Search for Y_1, X_1 and Y_2, X_2 with weight $t/2$

For each Y_1, such as 1011

consider possible $X_2 = 2011$

Meaning, include an inner loop over each $X_i \in [b-1]^{t/2}$

Generalizing to Radix$-b$

Assume, $t = 6, m = 8, b = 4$

Search for Y_1, X_1 and Y_2, X_2 with weight $t/2$

For each Y_1, such as 1011

consider possible $X_2 =$ 2011

Meaning, include an inner loop over each $X_i \in [b-1]^{t/2}$

Generalizing to Radix$-b$

Assume, $t = 6, m = 8, b = 4$

Search for Y_1, X_1 and Y_2, X_2 with weight $t/2$

For each Y_1, such as 1011

consider possible $X_2 =$ 2012

Meaning, include an inner loop over each $X_i \in [b-1]^{t/2}$

Generalizing to Radix$-b$

Assume, $t = 6, m = 8, b = 4$

Search for Y_1, X_1 and Y_2, X_2 with weight $t/2$

For each Y_1, such as 1011

consider possible $X_2 =$ 2013

Meaning, include an inner loop over each $X_i \in [b-1]^{t/2}$

Impact of Generalization

Requires an inner loop over each $X_i \in [b-1]^{t/2}$

Adds an additional factor of $(b-1)^{t/2}$

Basic Hamming Weight to Basic Low-Weight

$$\binom{m}{t/2} \quad \rightarrow \quad (b-1)^{t/2}\binom{m}{t/2}$$

Algorithmic Complexity Highlights

Best Deterministic Radix-2 Algorithm [2]

- Evaluates $\big(t + o(1)\big)\binom{m/2}{t/2}$
- Stores $2\binom{m/2}{t/2}$

Best Deterministic Radix$-b$ Algorithm (See B.3)

- Evaluates $\big(t + o(1)\big)\binom{m/2}{t/2}(b-1)^{t/2}$
- Stores $2\binom{m/2}{t/2}(b-1)^{t/2}$

[2] Improves best known (due to Stinson, 2001) by a factor $c\sqrt{t}\lg m$

Application to VPAKE Protocol

Originally: Keifer and Manulis, ESORICS 2014

km14.png

Password Authenticated Key Exchange

- Interactive protocol between a client and a server
- Authenticates client and establishes shared secret key
- Security requires that the interaction reveals at most negligible information

Verifier Based Password Authenticated Key Exchange

- The VPAKE protocol is used to register a password with the server, which will store a ‘verifier’
- Does not reveal the password, and
- Enforces a given password policy

VPAKE Protocol

User maps her password *pw* to an integer using the mapping

$$\pi = \texttt{PWDtoINT}(s\,;pw) := \sum_{i=1}^{|pw|} s^i pw_i$$

where, $pw_i \in [0\,..\,93]$ and $s \geq 94$

User computes a fingerprint of *pw*
Produces a pedersen-like commitment such that:

Difficulty for recovering *pw* is equivalent to solving DLP

Attack from Best Deterministic Low-Weight Algorithm

For any password of up to $m = 12$ characters

Consider brute force

$$\sum_{m=0}^{12} 94^m \approx 2^{78.7}$$

guesses, as compared with

$$\sum_{t=0}^{12} t\binom{m/2}{t/2} 93^{t/2} \approx 2^{38.2}$$

group operations

Summary

- Minimized hidden constants in low-Hamming weight algorithms, improving best known complexity
- Generalization of optimized algorithms for Low-Hamming weight to Low-Weight for $b > 1$
- Demonstrated cryptanlaytic applications against several VPAKE protocols

Arbitrary values t and m

Thus far assumed that t is even so that $t/2$ is an integer, but this is not a necessary condition

Results still hold if for example,

$$(X_1, Y_1) \in [b-1]^{\lfloor t/2 \rfloor} \times \binom{[m]}{\lfloor t/2 \rfloor}, \text{ and}$$

$$(X_2, Y_2) \in [b-1]^{\lceil t/2 \rceil} \times \binom{[m]}{\lceil t/2 \rceil}$$

For further analysis, see Stinson (2002)

The Question of Change

Theorem 15. *Fix a radix $b > 1$ and an exponent x with radix-b density d. There exists a constant $k_0 \in \mathbb{R}$ (with $k_0 > 1$) such that, for all $k > k_0$, if the radix-b^k density of x is less than or equal to d, then a radix-b^k algorithm has lower cost than the corresponding radix-b algorithm.*

Radix-4: 11012013 has density 0.75

Radix-8: 50607 has density 0.6

Order q Unknown

Two notes on adapting for unknown q:

- Set m to be any *upper bound* on $\lceil \lg q \rceil$
- Omit modular reduction in the algorithm (see paper)