Hardness of DLP, DDHP, CDHP

Tanja Lange

Eindhoven University of Technology

2WF80: Introduction to Cryptology

Hardness of DLP and CDHP

- BSGS takes at most $2m + 2$ multiplications and 1 inversion.
  Note that each step is a *multiplication* not an *exponentiation*.
- BSGS also needs space for storing m group elements.

Hardness of DLP and CDHP

- BSGS takes at most $2m+2$ multiplications and 1 inversion. Note that each step is a *multiplication* not an *exponentiation*.
- BSGS also needs space for storing m group elements.
- BSGS does not use any properties of the group, so this works in any cyclic group (and powers of g are a cyclic group).
- This means that DLP in any group cannot be harder than $O(\sqrt{|G|})$.

Hardness of DLP and CDHP

- BSGS takes at most $2m + 2$ multiplications and 1 inversion. Note that each step is a *multiplication* not an *exponentiation*.
- BSGS also needs space for storing m group elements.
- BSGS does not use any properties of the group, so this works in any cyclic group (and powers of g are a cyclic group).
- This means that DLP in any group cannot be harder than $O(\sqrt{|G|})$.
- We like groups where this is also the best attack cost.
- Disclaimer: we can avoid storage cost using Pollard's rho method. No better attacks known for elliptic curves. See 2MC10 for both.

Hardness of DLP and CDHP

- BSGS takes at most $2m + 2$ multiplications and 1 inversion.
  Note that each step is a *multiplication* not an *exponentiation*.
- BSGS also needs space for storing m group elements.
- BSGS does not use any properties of the group, so this works in any cyclic group (and powers of g are a cyclic group).
- This means that DLP in any group cannot be harder than $O(\sqrt{|G|})$.
- We like groups where this is also the best attack cost.
- Disclaimer: we can avoid storage cost using Pollard's rho method.
  No better attacks known for elliptic curves. See 2MC10 for both.
- For $\mathbb{F}_p^*$ (and $\mathbb{F}_{p^n}^*$) stronger attacks exist.
  Index calculus attacks have similar attack complexity to factoring.
  So we want p with at least 3072 bits.

Hardness of DLP and CDHP

- BSGS takes at most $2m + 2$ multiplications and 1 inversion.
  Note that each step is a *multiplication* not an *exponentiation*.
- BSGS also needs space for storing m group elements.
- BSGS does not use any properties of the group, so this works in any cyclic group (and powers of g are a cyclic group).
- This means that DLP in any group cannot be harder than $O(\sqrt{|G|})$.
- We like groups where this is also the best attack cost.
- Disclaimer: we can avoid storage cost using Pollard's rho method.
  No better attacks known for elliptic curves. See 2MC10 for both.
- For $\mathbb{F}_p^*$ (and $\mathbb{F}_{p^n}^*$) stronger attacks exist.
  Index calculus attacks have similar attack complexity to factoring.
  So we want p with at least 3072 bits.
- CDHP: No attacks better than solving DLP known,
  but it could be easier.
- There is a proof that breaking CDH implies breaking DLP

Hardness of DLP and CDHP

- BSGS takes at most $2m + 2$ multiplications and 1 inversion.
  Note that each step is a *multiplication* not an *exponentiation*.
- BSGS also needs space for storing m group elements.
- BSGS does not use any properties of the group, so this works in any cyclic group (and powers of g are a cyclic group).
- This means that DLP in any group cannot be harder than $O(\sqrt{|G|})$.
- We like groups where this is also the best attack cost.
- Disclaimer: we can avoid storage cost using Pollard's rho method.
  No better attacks known for elliptic curves. See 2MC10 for both.
- For $\mathbb{F}_p^*$ (and $\mathbb{F}_{p^n}^*$) stronger attacks exist.
  Index calculus attacks have similar attack complexity to factoring.
  So we want p with at least 3072 bits.
- CDHP: No attacks better than solving DLP known,
  but it could be easier.
- There is a proof that breaking CDH implies breaking DLP
  – but requiring several CDH computations for one DLP.
  The "several" depends on the group and can be many.

Hardness of DDHP

Given $g, h_A = g^a, h_B = g^b$, and $d = g^c$ decide whether $g^c = g^{ab}$.
This is no harder than CDHP – but can be much easier.

Take $G = \mathbb{F}_p^*$, generated by g. Observe that g has order $p-1$.

We can check whether a (or b or c) is even (without knowing them) by computing

$$h_A^{(p-1)/2} = \begin{cases} 1 \\ p-1 \end{cases} \text{for } a = \begin{cases} 2a' \\ 2a'+1 \end{cases}$$

because $(g^{2a'})^{(p-1)/2} = g^{a'(p-1)} = (g^{p-1})^{a'} \equiv 1^{a'} = 1 \bmod p$ and $g^{(p-1)/2}$ is the unique number that give 1 when squared but is not 1.

Turn this into an attack:
If (at least) one of a and b is even, then also $ab \bmod p-1$ is even, because reduction modulo the even number $p-1$ does not change parity.
If a and b are odd then $ab \bmod p-1$ is odd.

If c is randomly chosen then this is detected with probability

$$3/4 \cdot 1/2 + 1/4 \cdot 1/2 = 1/2.$$

Example of DDH attack

Take $G = \mathbb{F}_{53}^*$, generated by $g = 2$. With $h_A = 33, h_B = 25$, $d = 3$.

$h_A^{(p-1)/2} = 33^{26} \equiv 52 \bmod 53$. Thus a is odd.

$d^{(p-1)/2} = 3^{52} \equiv 52 \bmod 53$. Thus c is odd;
and we do not have an answer, yet.

Example of DDH attack

Take $G = \mathbb{F}_{53}^*$, generated by $g = 2$. With $h_A = 33, h_B = 25$, $d = 3$.

$h_A^{(p-1)/2} = 33^{26} \equiv 52 \bmod 53$. Thus a is odd.

$d^{(p-1)/2} = 3^{52} \equiv 52 \bmod 53$. Thus c is odd;
and we do not have an answer, yet.

$h_B^{(p-1)/2} = 25^{26} \equiv 1 \bmod 53$. Thus b is even!
We learn that this is not a valid DDH triple.

Example of DDH attack

Take $G = \mathbb{F}_{53}^*$, generated by $g = 2$. With $h_A = 33, h_B = 25$, $d = 3$.

$h_A^{(p-1)/2} = 33^{26} \equiv 52 \bmod 53$. Thus a is odd.

$d^{(p-1)/2} = 3^{52} \equiv 52 \bmod 53$. Thus c is odd;
and we do not have an answer, yet.

$h_B^{(p-1)/2} = 25^{26} \equiv 1 \bmod 53$. Thus b is even!
We learn that this is not a valid DDH triple.

We have broken this DDHP with 3 exponentiations.