CS489/689 Privacy, Cryptography, Network and Data Security

Winter 2023, Tuesday/Thursday 8:30-9:50am

Last Class: Padding Attack and MAC/Encrypt

- Learn activities were due today
- Responses will be used to finish the content
- If the content will not fit within the remainder of the crypto section Thursday I will record a lecture and release it with slides on Learn next week

Today: DLP, El Gamal, …

$$h = g^x \text{ , find } x$$

It's supposed to be hard to find x

I bet we can use that

But don't forget about me

Discrete Logarithm Problem

The Discrete Logarithm Problem

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

$$h = g^x$$

(Here **G** is a multiplicative group of prime order q)

Solutions to the Discrete Logarithm Problem?

If there's one solution, there are infinitely many (thank you Fermat's little theorem)

Fermat's Little Theorem (Recall attack Naive RSA)

Theorem: Let *p* be a prime number and let *a* be any integer.

Then:

$$a^{p-1} \equiv \begin{cases} 1 \pmod{p} & \text{if p does not divide a} \\ 0 \pmod{p} & \text{if p does divide a, } p|a \end{cases}$$

How to solve DLP in cyclic groups of prime order?

- Is the group cyclic, finite, and abelian?

Baby-step/Giant-step algorithms!!!

How to solve DLP in cyclic groups of prime order?

- Is the group cyclic, finite, and abelian?

How to solve DLP in cyclic groups of prime order?

- Is the group cyclic, finite, and abelian?

Baby-step/Giant-step algorithms!!!

Ohhhhhh. Divide and conquer since the bottleneck is solving DLP in the cyclic subgroups of prime order.

For **generic groups**, the complexity of the Baby-step/giant-step algorithm dominates the time required.

How to solve DLP in cycli… roup… f prim… order?

- Is the group cyclic, finite, a…

Ob…

…iant-step
…!!!

…tleneck is
…e order.

NOTE: for any actual group there may be specialized algorithms which work faster.

For **ge… group…** …e co… …ty of the Baby-step/giant-step algorithm domina… the time required.

Baby-Step/Giant-Step Algorithm? Notation.

- A public cyclic group G = <g> which has prime order p
- $h \in G$, goal: find x (mod p) such that $h = g^x$

- Divide and conquer?

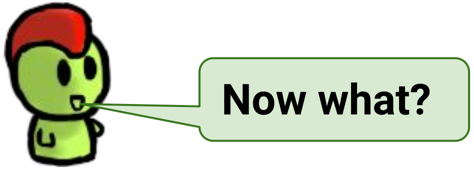
$$x = x_0 + x_1 * \lceil sqrt(p) \rceil$$

Baby-Step/Giant-Step Algorithm? Notation.

- A public cyclic group G = <g> which has prime order p
- $h \in G$, goal: find x (mod p) such that $h = g^x$

- Divide and conquer?

$$x = x_0 + x_1 * \lceil sqrt(p) \rceil$$

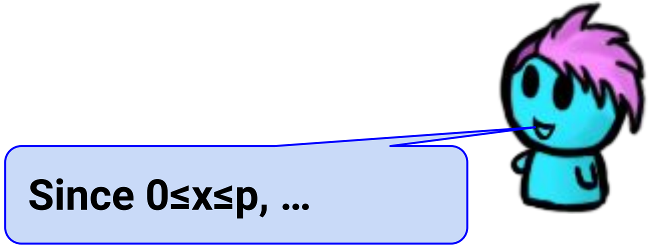


Baby-step/Giant-Step Algorithm

1. $x = x_0 + x_1 * \lceil sqrt(p) \rceil$

Baby-step/Giant-Step Algorithm

1. $x = x_0 + x_1 * \lceil \text{sqrt}(p) \rceil$
2. $0 \leq x_0, x_1 < \lceil \text{sqrt}(p) \rceil$
3.

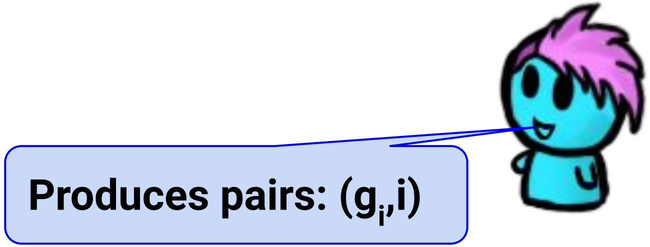


Baby-step/Giant-Step Algorithm

1. $x = x_0 + x_1 * \lceil sqrt(p) \rceil$
2. $0 \leq x_0, x_1 < \lceil sqrt(p) \rceil$
3. Baby-step: $g_i \leftarrow g^i$ for $0 \leq i < \lceil sqrt(p) \rceil$

Baby-step/Giant-Step Algorithm

1. $x = x_0 + x_1 * \lceil sqrt(p) \rceil$
2. $0 \leq x_0, x_1 < \lceil sqrt(p) \rceil$
3. Baby-step: $g_i \leftarrow g^i$ for $0 \leq i < \lceil sqrt(p) \rceil$



Baby-step/Giant-Step Algorithm

1. $x = x_0 + x_1 * \lceil \mathrm{sqrt}(p) \rceil$
2. $0 \le x_0, x_1 < \lceil \mathrm{sqrt}(p) \rceil$
3. Baby-step: $g_i \longleftarrow g^i$ for $0 \le i < \lceil \mathrm{sqrt}(p) \rceil$
4. Giant-step: $h_j \longleftarrow h * g^{-j \lceil \mathrm{sqrt}(p) \rceil}$, for $0 \le j < \lceil \mathrm{sqrt}(p) \rceil$
5.

Produces pairs: (h_j, j)

Baby-step/Giant-Step Algorithm

1. $x = x_0 + x_1 * \lceil sqrt(p) \rceil$
2. $0 \le x_0, x_1 < \lceil sqrt(p) \rceil$
3. Baby-step: $g_i \leftarrow g^i$ for $0 \le i < \lceil sqrt(p) \rceil$
4. Giant-step: $h_j \leftarrow h * g^{-j \lceil sqrt(p) \rceil}$, for $0 \le j < \lceil sqrt(p) \rceil$
5. Try to find a batch between baby-step and giant-step

Overall time and space $O(sqrt(p))$

Baby-step/Giant-Step Algo m

1. $x = x_0 + x_1 * \lceil sqrt(p) \rceil$
2. $0 \leq x_0, x_1 <$ ⌈sqrt
3.
4. Gian sqrt(p)⌉
5. Try to baby-step and giant-step

Note: For DLP in group G to be "difficult enough" (e.g., 2^{128} operations), needs prime order subgroup of size greater than 2^{256}

Overall time a space $O(sqrt(p))$

DLP Example, $182 = 64^x \pmod{607}$

- Note: the subgroup of order 101 in $\boldsymbol{F}_{607}$, generated by g=64

i		*i*	64^i (mod 607)
0		6	
1		7	
2		8	
3		9	
4		10	
5		-	

Baby-step: $g_i \leftarrow g^i$ for $0 \le i < \lceil sqrt(p) \rceil$

g = 64
⌈sqrt(p)⌉ = 10

j		*j*	18 … 4^{-11*j} (mod 607)
		6	
		7	
		8	
		9	
4		10	
5		-	

DLP Example, $182 = 64^x \pmod{607}$

i		i	64^i (mod 607)
0	1	6	330
1	64	7	482
2	454	8	498
3	527	9	308
4	343	10	288
5	100	-	

			^{1*j} (mod 607)
3		9	
4		10	
5		-	

DLP Example, $182 = 64^x \pmod{607}$

i		i	$64^i \pmod{607}$
0	1	6	330
1	64	7	482
2	454	8	498
3	527	9	308
4	343	10	288
5	100	-	

j		j	$182 * 64^{-11*j} \pmod{607}$
0	182	6	60
1	143	7	394
2	69	8	483
3	271	9	76
4	343	10	580
5	573	-	

DLP Example, $182 = 64^x \pmod{607}$

i		*i*	$64^i \pmod{607}$
0	1	6	330
1	64	7	482
2	454	8	498
3	527	9	308
4	**343**	10	288
5	100	-	

j		*j*	$182 \cdot 64^{-11 \cdot j} \pmod{607}$
0	182	6	60
1	143	7	394
2	69	8	483
3	271	9	76
4	**343**	10	580
5	573	-	

DLP Example, $182 = 64^x (\text{mod } 607)$

i		*i*	64^i (mod 607)
0	1	6	330
1	64	7	482
2	454	8	498
3	527	9	308
4	**343**	10	288
5	100		

j		*j*	$182 * 64^{-11*j}$ (mod 607)
0	182	6	60
1	143	7	394
2	69	8	483
3	271	9	76
4	**343**	10	580

Match when **i=4** and **j=4**.

DLP Example, $182 = 64^x \pmod{607}$

i		*i*	$64^i \pmod{607}$
0	1	6	330
1	64	7	482
2	454	8	498
3	527	9	308
4	**343**	10	288
5	10[illegible]		

j		*j*	$182 * 64^{-11*j} \pmod{607}$
0	182	6	60
1	143	7	394
2	69	8	483
3	271	9	76
4	**343**	10	580

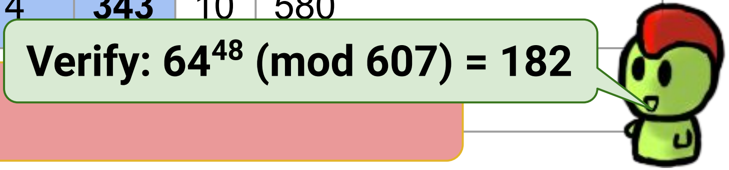
So: $x = 4 + 11*4 = 48$.

DLP Example, $182 = 64^x \pmod{607}$

i		i	$64^i \pmod{607}$
0	1	6	330
1	64	7	482
2	454	8	498
3	527	9	308
4	**343**	10	288
5	10[illegible]		

j		j	$182 * 64^{-11*j} \pmod{607}$
0	182	6	60
1	143	7	394
2	69	8	483
3	271	9	76
4	**343**	10	580

So: $x = 4 + 11*4 = 48$.



Act.

The value x

Q: Consider, $h = g^x$ and that x has been chosen such that the base-2 representation has few non-zeros.

Act.

The value x

Q: Consider, $h = g^x$ and $x \in Z_{31}{}^*$ has been chosen such that the base-2 representation has few non-zeros. Let g = 3 and h = 11. Each Y_b is length five with 2 bits of value 1.

Recall,

Giant: $\mathbf{g_i \leftarrow g^i}$

Baby: $\mathbf{h_j \leftarrow h * g^{-j \lceil sqrt(31) \rceil}}$

Giant-Step

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

Baby-Step

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

Act.

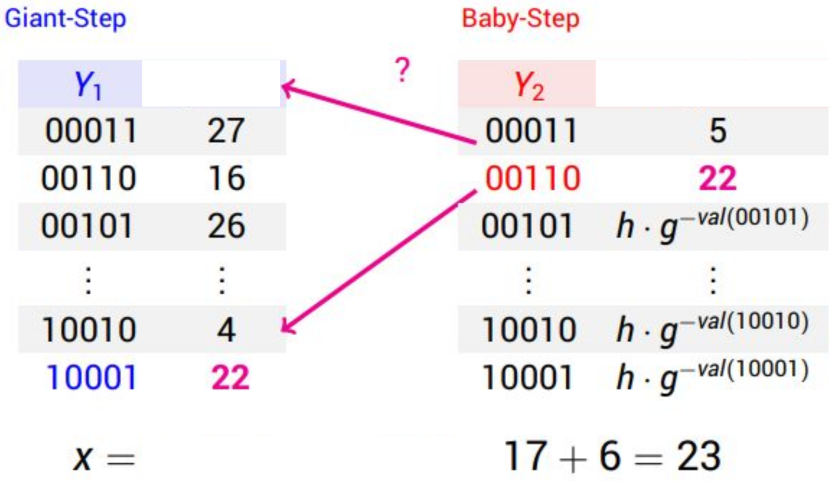
The value x

Q: Consider, $h = g^x$ and $x \in Z_{31}$* has been chosen such that the base-2 representation has few non-zeros. Let g = 3 and h = 11. Each Y_b is length five with 2 bits of value 1.

Recall,

Giant: $\mathbf{g_i} \leftarrow \mathbf{g^i}$

Baby: $\mathbf{h_j} \leftarrow \mathbf{h * g^{-j \lceil sqrt(31) \rceil}}$



Submit a match and four other rows.

Thursday: More Cryptography…

FAQ: Groups/Math Definitions

A Group:

- A set with an operation on its elements which
  - Is closed
  - Has an identity
  - Is associative, and
  - Every element has an inverse
- Commutative groups are called **abelian**

Groups with properties

- A cyclic group of prime order cannot be broken down into smaller groups
- Cyclic subgroups are generated by a generator g raised to a series of powers (the group consists of all its integer powers)

Mini Proof of Fermat's Little Theorem

- If $p|a$, then every power of *a* is divisible *p*.

Mini Proof of Fermat's Little Theorem

- If p|a, then every power of *a* is divisible *p*. **So we can skip it.**

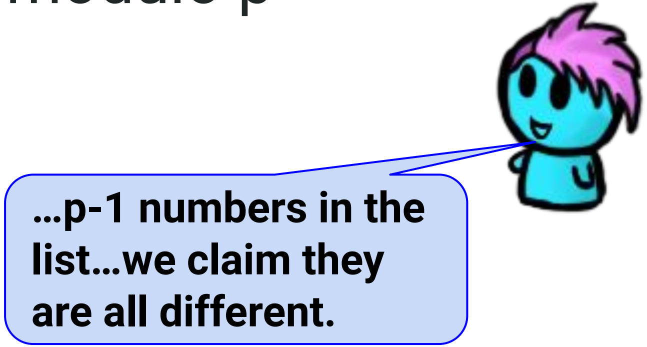
Mini Proof of Fermat's Little Theorem

- If p|a, then every power of *a* is divisible *p*. **So we can skip it.**
- So what about when *p* doesn't divide *a*?
- a, 2a, 3a, …, (p-1)a reduced modulo p

...p-1 numbers in the list...we claim they are all different.

Mini Proof of Fermat's Little Theorem

- If p|a, then every power of *a* is divisible *p*. **So we can skip it.**
- So what about when *p* doesn't divide *a*?
- a, 2a, 3a, ..., (p-1)a reduced modulo p



Mini Proof of Fermat's Little Theorem

- If p|a, then every power of *a* is divisible *p*. **So we can skip it.**
- So what about when *p* doesn't divide *a*?
- a, 2a, 3a, …, (p-1)a reduced modulo p

…p-1 numbers in the list…we claim they are all different.

Could you explain why?

Mini Proof of Fermat's Little Theorem

- When *p* doesn't divide *a*?
- a, 2a, 3a, …, (p-1)a reduced modulo p
- Consider *ja* mod p and *ka* mod *p*

Mini Proof of Fermat's Little Theorem

- When *p* doesn't divide *a*?
- a, 2a, 3a, …, (p-1)a reduced modulo p
- Consider *ja* mod p and *ka* mod *p*
- Then, *ja*≡*ka* (mod p), and, (j-k)a≡0 (mod p)

Mini Proof of Fermat's Little Theorem

- When *p* doesn't divide *a*?
- a, 2a, 3a, …, (p-1)a reduced modulo p
- Consider *ja* mod p and *ka* mod *p*
- Then, *ja*≡*ka* (mod p), and, (j-k)a≡0 (mod p)

1) Suppose they are the same

2) Thus p|(j-k)a

Mini Proof of Fermat's Little Theorem

- When *p* doesn't divide *a*?
- a, 2a, 3a, …, (p-1)a reduced modulo p
- Consider *ja* mod p and *ka* mod *p*
- Then, $ja \equiv ka \pmod{p}$, and, $(j-k)a \equiv 0 \pmod{p}$
- Etc…
-

1) Suppose they are the same

2) Thus p|(j-k)a