1 Reverse Homework Review

1.1 Collision-Resistant Hash Functions from Discrete Logarithms

Modular Exponentiation modular p with g as the generator, $< g > = \mathbb{Z}_p^*$ and $|p| \approx 1024$ bits,

$\text{exp}_{p,g} = \mathbb{Z}_{p-1} \rightarrow \mathbb{Z}_p^*$

$x \rightarrow g^x \mod p$

And the other way:

$x \leftarrow h \epsilon \mathbb{Z}_p^*$ where $x$ s.t. $g^x \mod p = h$ and $g \epsilon \mathbb{Z}_p^*$

Suppose $< h > = < g > = \mathbb{Z}_p^*$, $g \neq h$

multi - $\text{exp}_{g,h,p} : \mathbb{Z}_{p-1} \times \mathbb{Z}_{p-1} \rightarrow \mathbb{Z}_p^*$

$(u, v) \rightarrow g^u h^v \mod p$

Check:

1. Efficient? $\sqrt{}$
2. Compresses? $\sqrt{}$
3. Collision Resistant? $\sqrt{}$ yes, show by reduction.

Proof

Suppose attacker can break CRHF

$\rightarrow (u_1, v_1) \neq (u_2, v_2)$ such that $g^{u_1} h^{v_1} \equiv g^{u_2} h^{v_2} \mod p$

This is difficult to show, so we will do the converse.

\[
g, h, p \\
\downarrow \\
\underline{\text{A}_{\text{CRHF}}} \\
\downarrow \\
(u_1, v_1) \neq (u_2, v_2) \text{ s.t. } g^{u_1} h^{v_1} \equiv g^{u_2} h^{v_2} \mod p \\
\downarrow \\
x \text{ s.t. } g^x = h \mod p
\]
We want $x$ such that $g^x = h \mod p$

We call $A_{CRHF}$ on $(g, h, p)$ and obtain $(u_1, v_1) \neq (u_2, v_2)$ s.t. $g^{u_1}h^{v_1} \equiv g^{u_2}h^{v_2} \mod p$ because $(h^{v_1})^{-1} \mod p$ and $(g^{u_1})^{-1} \mod p$ exist.

$g^{u_1}h^{v_1} \equiv g^{u_2}h^{v_2} \mod p$

Suppose $\gcd(v_2 - v_1, p - 1) = 1$

$\implies \exists (v_2 - v_1)^{-1} \mod (p - 1)$, we will call $(v_2 - v_1)^{-1} \varphi$

$(g^{(u_1-u_2)} \varphi = (h^{(v_2-v_1)} \varphi) \mod p$

$(v_2 - v_1) \varphi \equiv 1 \mod (p - 1) \iff (v_2 - v_1) \varphi = 1 + k(p - 1)$

$(v_2 - v_1) \varphi \equiv 0 \mod (p - 1) \quad p - 1 | ((v_2 - v_1) \varphi - 1)$

$(v_2 - v_1) \varphi - 1 = kp$

So $g^{(u_1-u_2) \varphi} = h^{(v_2-v_1) \varphi} \mod p = h^1h^{k(p-1)} = h(h^{p-1})^k = h$

Therefore, $x = (u_1 - u_2) \varphi$

And finally, choose strong prime $p$ and $g, h$ such that $g = g^2$ and $h = h^2$.

### 1.2 Cyclic Groups

Recall $\mathbb{Z}_p^*$ is cyclic when $p$ is prime. Why is $\mathbb{Z}_n^*$ where $n$ is the product of two primes, i.e. $n = pq$, not cyclic?

Consider $n = 15 = 3 * 5$

- $\mathbb{Z}_3^* \cong \mathbb{Z}_2^* * \mathbb{Z}_5^*$
- $\mathbb{Z}_5^* = 1, 2, 3, 4$ with generator $\langle 2 \rangle$
- $\mathbb{Z}_2^* = 1, 2, 3, 4$ with generator $\langle 2 \rangle$, where $2^1 = 2, 2^2 = 4, 2^3 = 3, 2^4 = 1$, i.e. $2 \rightarrow 4 \rightarrow 3 \rightarrow 1$.
- $\mathbb{Z}_{15}^* = \{1, 2, 4, 7, 8, 11, 13, 14\} = \{1, 2, 4, 7, 7, 4, 2, -1\}$

Check for generators:

1. $1 \rightarrow 1$
2. $2 \rightarrow 4 \rightarrow 8 \rightarrow 1$
3. $4 \rightarrow 1$
4. $7 \rightarrow 4 \rightarrow 13 = -2 \rightarrow 1 = -14$

Therefore, no generators and not cyclic.

Consider $\mathbb{Z}_{15}$ as a cross product, $\mathbb{Z}_{15} \rightarrow \mathbb{Z}_3^* \times \mathbb{Z}_5^*$.

In order for $\mathbb{Z}_n^*$ to be cyclic, there must exist $a^{\varphi(n)} = 1 \mod n$. So $\mathbb{Z}_n^*$ cyclic $\iff a^{\varphi(n)} = 1 \mod n$ and $a^{l} \neq 1 \mod n$ where $l < \varphi(n)$. $\varphi(n) = (p - 1)(q - 1) > 2 \left(\frac{p-1}{2}\right) + \left(\frac{q-1}{2}\right)$ is even.

$lcm(p - 1, q - 1) = \lambda \leq \left(\frac{(p-1)(q-1)}{2}\right)$ called the grammatical function of $n$

$$a^\lambda = 1 \mod n$$

$\lambda < \varphi(n)$

Now consider $a \rightarrow (a_p, a_q)$ where $a_p = a \mod p$ and $a_q = a \mod q$.

WTS: $(a_p^\lambda, a_q^\lambda) = (1, 1)$

If $\lambda = lcm(p - 1, q - 1) = kq(p - 1) = kp(q - 1)$

$(a_p^\lambda, a_q^\lambda) = (1, 1)$ iff $a_p^\lambda = 1(p^\lambda \land a_p^\lambda) = 1(q)$

L11-2
\[ a_p^k = a_p^{k(p-1)} = (a_p^{p-1})^k = 1^k q = 1(p) \]

Therefore, no generator and not cyclic.

Note: \( \text{lcm}(a, b) = \frac{ab}{\text{gcd}(a,b)} \)

## 2 Chinese Remainder Theorem

Let \( n = pq \). We know \((\mathbb{Z}_n, +) \cong (\mathbb{Z}_p, +) \times (\mathbb{Z}_q, +)\) is a group. Is \((\mathbb{Z}_n, \cdot) \cong (\mathbb{Z}_p, \cdot) \times (\mathbb{Z}_q, \cdot)\) a group?

Let \( \theta \) be a bijection such that

\[ \theta : \mathbb{Z}_n \rightarrow \mathbb{Z}_p \times \mathbb{Z}_q \]

\[ a \rightarrow a_p x a_q \]

Let \( \delta \) be the inverse such that

\[ \delta : \mathbb{Z}_p \times \mathbb{Z}_q \rightarrow \mathbb{Z}_n \]

\[ (a_p x a_q) \rightarrow a = b a_p + d a_q (mod n) \]

such that \( b = 1 \) modp or 0 modq and \( d = 0 \) modp or 1 modq.

Then \( b a_p + d a_q (mod p) = 1 a_p + 0 a_q = a_p \) if \( b \) has the property. And vice versa, \( b a_p + d a_q (mod p) = 0 a_p + 1 a_q = a_q \) if \( d \) has the property.

So now how do you get \( b \) to be as we want it?

Let \( \bar{p} = p^{-1} mod q \) and \( \bar{q} = q^{-1} mod p \). \( b = q \bar{q} \) and \( d = p \bar{p} \).

Note: we don’t need “mod n” in the above equations because both \( b \) and \( d \) are less than \( pq = n \).

Now, by construction \( q \bar{q} mod p = 1 \) and \( p \bar{p} mod q = 1 \). And therefore efficient. It remains to show there exists an isomorphism, but we’re not going to.

## 3 Public Key Encryption

### 3.1 Shamir’s No-Key Protocol

Let \( p \) be prime, \(< g > = \mathbb{Z}_p^* \) and \( m \in \mathbb{Z}_p^* \)

<table>
<thead>
<tr>
<th>ALICE</th>
<th>BOB</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. pick ( e \in \mathbb{Z}<em>{p-1} ). Suppose ( (e, p-1) = 1 ). then ( \exists de \in \mathbb{Z}</em>{p-1} ) s.t. ( ed = 1 (mod p) ) ( \rightarrow m^e mod p \rightarrow )</td>
<td>2. pick ( f \in \mathbb{Z}<em>{p-1} ). Suppose ( (f, p-1) = 1 ) ( \exists ce \in \mathbb{Z}</em>{p-1} ) s.t. ( fc = 1 (mod p) ) ( \leftarrow y_B \leftarrow )</td>
</tr>
<tr>
<td>( m^{ef} mod p = (m^e)^f = (m^1)^f mod p \rightarrow m \rightarrow ) Compute ( m^{efdc} )</td>
<td></td>
</tr>
</tbody>
</table>

\[ m^{efdc} = m (mod p) \]

In general, \( a^m mod p = a^{u mod (p-1)} mod p \), so \( (m^e)^f c = (m^1)^1 = m \).

Recall definition of remainder, \( u = \lfloor \frac{u}{p-1} \rfloor (p-1) + u mod (p-1) \)
So, $a^u = a^{\lfloor \frac{u - 1}{p - 1} \rfloor (p - 1)} + u \mod (p - 1) = (a^{p - 1})^{\lfloor \frac{u - 1}{p - 1} \rfloor} \ast a^{u \mod (p - 1)} = 1^{\lfloor \frac{u - 1}{p - 1} \rfloor} \ast a^{u \mod (p - 1)} = a^{u \mod (p - 1)}$.

And the same thing holds true where the exponent is the order of the group, i.e. $a^u \mod p = a^{u \mod \phi(n)} \mod n$.

### 3.2 Pohlig-Hellman Cryptosystem

**definition:** SK=$(e, f)$ such that $eeZ_{p-1}$, $d = e^{-1} \mod (p - 1)$,

- $Enc((e, d), m) = m^e \mod p$,
- $Dec((e, d), c) = c^d \mod p$,
- $(m^e)^d \mod p = m^{ed \mod (p - 1)} \mod p = m$

This is as good as OTP. However, note that Shamir’s protocol is not as good as OTP.

This is not RSA because $p-1$ is not prime and everything is secret.

$en$ does not imply $d$ where $\phi(n) = (p - 1)(q - 1)$, but $en \Rightarrow d$ where $\phi(p) = p - 1$. This means that from $\phi(n)$ we can find out its factorization and vice versa.

In protocol, we want to get rid of the assumption "suppose $(e, p-1)=1"$ so the order of the group is prime. Recall LaGrange Theorem: $H \subseteq G$, and $G$ is a group, then $ord(H) \mid ord(G)$. This leads us to the fact that if $p$ is prime suchthat $|p|=1,024$ bits and there exists $q$ prime where $|q|=160$ such that $p = hq + 1$ for some $h$, and therefore, $p - 1 = hq$.

$Z_p^*$ has order $hq$

$\Rightarrow q \mid ord(Z_p^*)$

$\Rightarrow$ there exists $G \subset Z_p^*$ such that the order is $q$, i.e. $|G| = q$

How do you find such a subgroup?

Let $\bar{g}$ be the generator of $Z_p^*$, i.e. $Z_p^* = \bar{g}, \bar{g}^2, ... , \bar{g}^{p-1}$ and $\bar{g} = \bar{g}^{h \mod p}$.

So $G = \langle g \rangle = \{g, g^2, ..., g^q \} = \{g^h, g^{2h}, ..., g^{hq} \}$

The first thing we want to do is show $g^q = 1 \mod p$.

$1 = (g^{hq})^q = g^{hq} = g^{p-1} = 1 \mod p$

Next we want to show $q$ is the smallest value such that you get 1 where $q$ is the first power so $g^q = 1 \mod p$

$g^r \not= 1 \mod p$ for $1 \leq r \leq q$

By construction, $g^q = 1 \Rightarrow g^{hr} = 1$ and so $hr < hq$. But this is a contradiction to the assumption that $\bar{g}$ is a generator

Finally if you factor $h$ out of the order, you are left with order $q$, which is what we wanted.
3.3 Contraind Modular Exponentiation

Given \( h,q,g,p \) where \( p, q \) are primes, \( p = hq + 1 \), \( |p| = 1024 \) and \( |q| = 160 \).

\[
\exp_{hqg} : \mathbb{Z}_q \mapsto \mathbb{Z}_p^* \\
x \mapsto g^x \mod p
\]

Then the image of \( \exp_{hqg} \) is \( G \).

If \( a \in G \subset \mathbb{Z}_p^* \) and \( |G| = q \) where \( q \) is prime, then \( a^n = a^{\text{mod}\ q} = a^{\text{mod}\ (p-1)} = a^{\text{mod}\ hq} \).

The running time to compute \( a^x \mod p \) is \( O(\|x\| \times |p|^2) \) where \( O \) is the order. To get the same security as AES, you can use \( p = 2^{15} \) and \( q = 266 \), but this would be a lot slower.

Now to redo/update Shamir’s protocol

**ALICE**  
1. pick \( e \in \mathbb{Z}_{p-1} \).  
   then \( \exists d \in \mathbb{Z}_{p-1} \) s.t. \( ed = 1 \mod p \)  
   \[ \rightarrow m^e \mod p \rightarrow \]

**BOB**  
2. pick \( f \in \mathbb{Z}_{p-1} \).  
   then \( \exists c \in \mathbb{Z}_{p-1} \) s.t. \( fc = 1 \mod p \)  
   \[ \leftarrow y_B \leftarrow \]

\[ \rightarrow m^{efd} \mod p = (m^e)^f = (m^1)^f \mod p - 1 = m \rightarrow \]

Compute \( m^{efdc} \)

The only difference between this and the previous picture is the assumption that \( e \) and \( p-1 \) are relatively prime has been removed. This has no effect on how the protocol works; it works the same.

3.4 Strong Primes

The usage of string primes doesn’t get you efficiency.

**Proof**  
Left \( p \in \mathbb{Z}_p^* \) be a strong prime such that \( p = 2p' + 1 \), which implies \( p' = \frac{p-1}{2} \). (Note, here \( p' = q \)). Let \( G \subset \mathbb{Z}_p^* \) and \( |G| = p' = q = \frac{|\mathbb{Z}_p^*|}{2} \)
< g > = G and < \bar{g} > = Z_p^* \{4 \}
\bar{g} = \bar{G} \mod p
\bar{g}^p = \bar{g}^{2p'} = \bar{g}^{p-1} = 1
But r < p' so \bar{g}^r = \bar{g}^{2r} \neq 1

3.5 Quadratic Residues

\mathbb{Z}_p^* \mapsto \mathbb{Z}_p^*
\ h \mapsto h^2 \mod p

The resulting image is exactly 1/2 of the set, i.e. \(|G| = \frac{1}{2} |\mathbb{Z}_p^*|\). Also realized that because we are dealing with square roots, if you take a value in the domain and square it, you only get one value in the image. However, if you take a square and square root it (i.e. the other way/direction), you will get 2 values which account for the positive and negative square root. If a value is a quadratic residue modp, for example, it is denoted QR or also SQ_p.

3.6 Finding Square Roots mod p

x \in QR_p, \ \exists y \in \mathbb{Z}_p \ \text{s.t.} \ x = y^2 (\mod p)
\frac{p+1}{4} = \frac{2p+1+1}{4} = \frac{p'}{2} \ \text{is even} \ \bullet \in \mathbb{Z}

Claim: \ y = x^{\frac{p+1}{4}} (\mod p) \ \text{(check if} \ \frac{p+1}{4} \text{is an integer.)}
y^2 = x^{\left(\frac{p+1}{4}\right)^2} (\mod p)
y^2 = x^{\left(\frac{p+1}{2}\right)} (\mod p)
y^2 = x^{\frac{p-1}{2}} \cdot x = +1 \cdot x = x (\mod p)
x \in QR_p \iff \frac{x}{p} = +1 \iff x^{\frac{p-1}{2}} (\mod p) = 1

Example: Show if y \in QR_p, i.e. y^{\frac{p-1}{2}} = 1, then y^{\frac{p+1}{4}} is one of its square roots.
WTS: (y^{\frac{p+1}{4}})^2 = y (\mod p)
y^{\frac{p+1}{2}} = y^{\frac{p-1}{2}} = y^{\frac{p-1}{4}} y^2 = 1 \ast y = y
(Note: if y is not a square root, the above equation results in \neg y.)
3.7 Protocol

Let Alice and Bob share \( p = 2p' + 1 \) and \(<g> = G = QR_p\)

\[
\begin{array}{ll}
\text{ALICE} & \text{BOB} \\
1. x_A \leftarrow \mathbb{Z}_p^* & y_B \leftarrow \mathbb{Z}_p^* \\
y_A \leftarrow g^{x_A \mod p} & y_B \leftarrow g^{x_B \mod p} \\
y_A \rightarrow & y_B \\
2. x_B \leftarrow \mathbb{Z}_{p'} & y_A \leftarrow g^{x_A \mod p} \\
y_B \leftarrow g^{x_B \mod p} & y_B \leftarrow g^{x_B \mod p}
\end{array}
\]

For Alice:
\[
k_{AB} := (y_A)^{x_B} \mod p \\
= (g^{x_A})^{x_B} \mod p \\
= g^{x_A x_B} \mod p
\]

And for Bob:
\[
k_{AB} := (y_A)^{x_B} \mod p \\
= (g^{x_A})^{x_B} \mod p \\
= g^{x_A x_B} \mod p
\]

If the discrete log can be broken, the system is broken. And the protocol can be broken even if the discrete log is hard to compute, i.e. if the discrete log is easy to compute, it implies the entire algorithm/process is easy to calculate, however, any easy algorithm does not necessarily imply the discrete log is easy to calculate.