Anonym function evaluation

From dining cryptographers to socialist millionaires

Holczer Tamás

Introduction and outline

- Alternative to mixnets: DC-nets
- Importance of OR
- Anonym communication without infrastructure

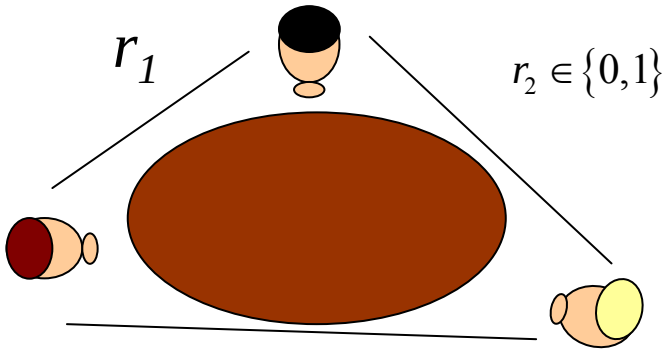
- **How to send boolean OR anonymously from a group?**

- Dining Cryptographers Problem: Unconditional sender and recipient Untraceability
- Socialist millionaires
- Anonym veto protocol
- Bibliography

Dining cryptographers (Chaum)

- Basic example of anonymous communication
- Offers sender and recipient anonymity
- Three cryptographers, who paid? Them or NSA?

1. Everyone picks random r_i
2. Pass to right
3. Difference
4. Add message if any
5. Broadcast sum (xor)
6. Total sum = message



$s_2 = r_2 + r_3 + m_2$

Everyone can be the recipient
No one knows the sender

$$S = s_1 + s_2 + s_3 = m_1 + m_2 + m_3$$

DC: Generalization, problems, properties

- N>3 nodes
- Every node must share a secret bit with every other node

$$S_i = \sum_j s_{i,j} + m_i$$

- Every node shares a key with the two neighbours (ring)
  - Two attacker can divide the anonymity set to two
- In general, attackers can divide the graph of users into smaller anomity sets
- Collision (more than 1 sender) leads to ambiguous result
- Problems:
  - key management, new users
  - result can inverted maliciously
- Non-interactive, unconditional

Socialist millionaries' protocol (Brandt) – Building blocks

- Similar to „Who is richer? My wealth is a secret!" (Yao 82)
- El Gamal Encryption (brief reminder):
  - p,q large primes
  - $p-1=kq$ for some k
  - private key: $x \in \mathcal{Z}_q$
  - public key: $y=g^x$
  - encryption: $(a,b)=(my^r,g^r)$
  - decryption: $a/b^x=my^r/g^{rx}=my^r/y^r=m$
- Homomorphism of El Gamal Encryption:
  - same key, product of two messages
  - $(a_1a_2,b_1b_2)=(m_1m_2y^{r1+r2}, g^{r1+r2})$
  - $a_1a_2/b_1{}^xb_2{}^x= m_1m_2y^{r1+r2}/ g^{xr1+xr2}= m_1m_2y^{r1+r2}/y^{r1+r2}=m_1m_2$
- El Gamal encryption is semantical secure if DDH problem is intractable (DDH: knowing g^a,g^b ,*hard to distinguish between* g^{ab} *and* g^c)

Socialist millionaries' protocol (Brandt) – Building blocks

- Distributed key generation:
  - x_i chosen at random by each participant
  - $y_i=g^{xi}$ is broadcast with proof of knowledge of x_i (later)
  - $y=\Pi y_i$ is the public key
  - $x=\Sigma x_i$ is the private key
- Distributed decryption:
  - *(a,b)* cyphertext
  - b^{xi} is broadcast with proof of equality of logarithm of b^{xi} *and* y_i
  - $m=a/\ \Pi\ b^{xi}$
- Random exponentiation:
  - M_i *random number*
  - $(a^{Mi},\ b^{Mi})$ *is broadcast* with proof of equality of logarithm of a^{Mi} *and* b^{Mi}
  - $(a^M,\ b^M)=\ \Pi\ (a^{Mi},\ b^{Mi})$

Socialist millionaries' protocol (Brandt) – Building blocks

- Proof of knowledge of discrete logarithm (interactive form):
  - Alice and Bob know v and g, but only Alice knows x, so that $v = g^x$.
  - A→B: $a = g^z$, z random value
  - B→A: c random value
  - A→B: $r = (z+cx) \bmod q$
  - B checks: $av^c = g^z (g^x)^c = g^{(z+cx)} = g^r$
- Proof of knowledge of discrete logarithm (non-interactive form):
  - Alice and Bob know v and g, but only Alice knows x, so that $v = g^x$.
  - $e=H(g,r,v)$
  - $r=g^k$ $s=k-xe$ proof: (r,s)
  - verification: $g^s v^e = g^{k-xe} g^{xe} = g^k = r$
- Proof of equality of two discrete logarithms (interactive form):
  - Alice and Bob know v,w, $g1$, and $g2$, but only Alice knows x, so that $v = g1^x$ and $w = g2^x$
  - A→B: $a = g1^z$ and $b = g2^z$, z random value
  - B→A: c random value
  - A→B: $r = (z+cx) \bmod q$
  - B checks: $av^c = g1^z (g1^x)^c = g1^{(z+cx)} = g1^r$ and $aw^c = g2^z (g2^x)^c = g2^{(z+cx)} = g2^r$

Socialist millionaries' protocol (Brandt) – Building blocks

- Proof that an encrypted value is one out of two values:
  - *m∈(1,z)*, without revealing *M*

1. If $m = 1$, Alice chooses r_1, d_1, w at random and sends (α, β), $a_1 = g^{r_1}\beta^{d_1}$, $b_1 = y^{r_1}\left(\frac{\alpha}{z}\right)^{d_1}$ and $a_2 = g^w$, $b_2 = y^w$ to Bob.
   If $m = z$, Alice chooses r_2, d_2, w at random and sends (α, β), $a_1 = g^w$, $b_1 = y^w$, $a_2 = g^{r_2}\beta^{d_2}$, and $b_2 = y^{r_2}\alpha^{d_2}$ to Bob.
2. Bob chooses a challenge c at random and sends it to Alice.
3. If $m = 1$, Alice sends d_1, $d_2 = c - d_1 \mod q$, r_1, and $r_2 = w - rd_2 \mod q$ to Bob.
   If $m = z$, Alice sends $d_1 = c - d_2 \mod q$, d_2, $r_1 = w - rd_1 \mod q$, and r_2 to Bob.
4. Bob checks that $c = d_1 + d_2 \mod q$, $a_1 = g^{r_1}\beta^{d_1}$, $b_1 = y^{r_1}\left(\frac{\alpha}{z}\right)^{d_1}$, $a_2 = g^{r_2}\beta^{d_2}$, and $b_2 = y^{r_2}\alpha^{d_2}$.

Socialist millionaries' protocol (Brandt) – The veto protocol

- *1. round: Distributed key generation*
- *2.* round: $E(b_i)$
  - $d_i=1$ for agree
  - $d_i=Y$ for veto (Y publicly known constant)
  - $D(E(d_i)) \in (1,Y)$, can be proven
  - compute $\Pi_i(E(d_i))$
- 3. round: $\Pi_i(E(d_i))^{Mi}$
  - compute $\Pi_i(E(d_i))^M$ M random value, not known to anyone
- *4. round:* $d=D_{joint}(\Pi_i(E(d_i))^M)$,
- No veto: $d=1$
- Veto: d random value

Anonym veto protocol (Hao, Zielinski)

- 1. round
  - x_i private key computed
  - g^{xi} is broadcast (with knowledge proof of x_i)
  - compute g^{yi}:

$$g^{y_i} = \frac{\prod_{j=1}^{i-1} g^{x_j}}{\prod_{j=i+1}^{N} g^{x_j}}$$

- 2. round
  - broadcast (w kp):

$$g^{c_i y_i} = \begin{cases} g^{r_i y_i} \text{ if veto, } r_i \text{ random} \\ g^{x_i y_i} \text{ if no veto} \end{cases}$$

  - compute:

$$D = \prod_{i=1}^{N} g^{c_i y_i}$$

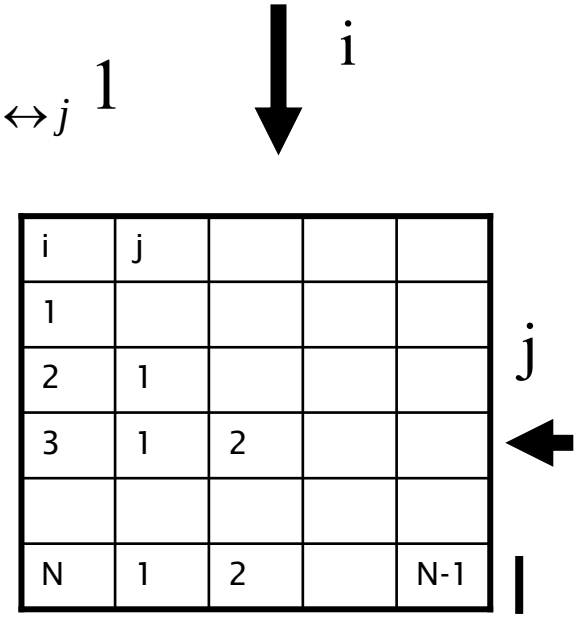
Anonym veto protocol (Hao, Zielinski)

- If no veto: D=1

$$D = \prod_{i=1}^{N} g^{c_i y_i} = \prod_{i=1}^{N} g^{x_i y_i} = \prod_{i=1}^{N} \left(\frac{\prod_{j=1}^{i-1} g^{x_j}}{\prod_{j=i+1}^{N} g^{x_j}} \right)^{x_i} =$$

$$\frac{\prod_{i=1}^{N} \prod_{j=1}^{i-1} g^{x_j x_i}}{\prod_{i=1}^{N} \prod_{j=i+1}^{N} g^{x_j x_i}} = \frac{\prod_{j=1}^{N-1} \prod_{i=j+1}^{N} g^{x_j x_i}}{\prod_{i=1}^{N} \prod_{j=i+1}^{N} g^{x_j x_i}} = \frac{\prod_{j=1}^{N} \prod_{i=j+1}^{N} g^{x_j x_i}}{\prod_{i=1}^{N} \prod_{j=i+1}^{N} g^{x_j x_i}} =_{i \leftrightarrow j} 1$$

- If veto: D=random



Comparison

Author	Year	Round	Security	Total traffic	Total comp
Chaum	‘88	2	uncond	$O(n^2)$	$O(n^2)$
Brandt	‘05	4	DDH	O(n)	O(n)
Hao Zielinski	‘06	2	DDH	O(n)	O(n)

Bibliography

- D. Chaum. **The dining cryptographers problem: *unconditional sender and recipient untraceability***, Journal of Cryptology, Volume 1 , Issue 1 (1988)
- F. Brandt. **Efficient cryptographic protocol design based on distributed El Gamal encryption.** In *Proceedings of the 8th International Conference on Information Security and Cryptology (ICISC)*, volume 3935 of *Lecture Notes in Computer Science (LNCS)*, pages 32-47. Springer-Verlag, 2005.
- F. Hao, P. Zielinski, "**A 2-round anonymous veto protocol**," 14th International Workshop on Security Protocols, Cambridge (2006)