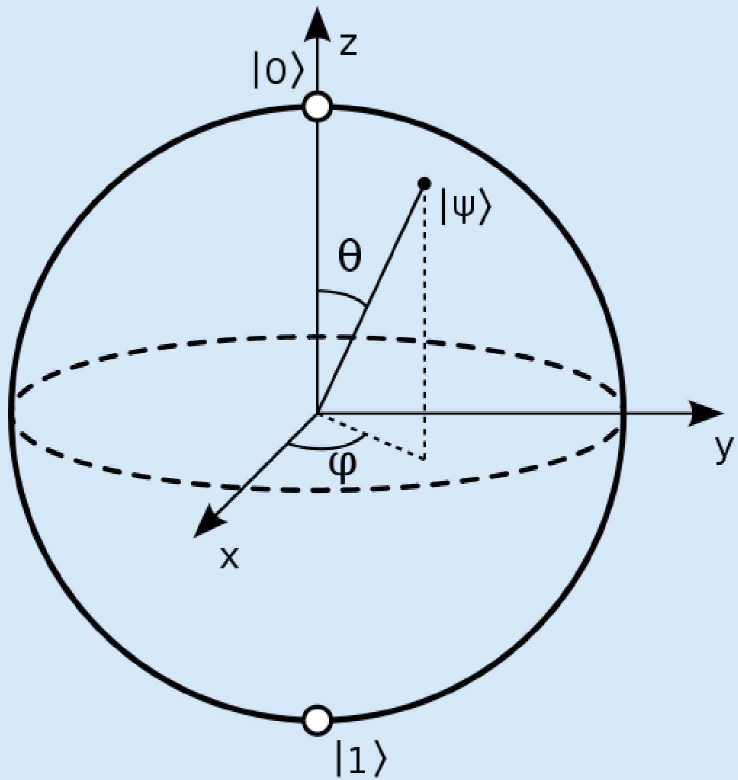
Quantum Computing for Classical Public-Key Cryptosystems

Lindsay Stewart
Mentor: Wim van Dam
Department of Computer Science
EUREKA Summer Internship
8/22/11

Quantum Computing



- Qubits vs. Bits
- "Quantum parallelism"
- Unitary transformations (rotations) only

Public-Key Cryptography

Security relies on hard mathematical problems defined by the public key which become easy when you know the private key

Example of Public-Key Cryptosystems:
RSA is based on the problem of factoring

Subset-Sum Problem

Given a set B of integers, what subset (if any) sums to an arbitrary integer S?

Example: { 319, 196, 250, 477, 200, 559 }, with the target S = 1605

Solution: 319 + 250 + 477 + 559 = 1605

Examples of cryptosystems based on the subset-sum problem:

Merkle-Hellman, Chor-Rivest

Is the cryptosystem secure?

1. Study public-key cryptosystems based on the Knapsack/Subset-Sum problem
2. Develop quantum algorithms to break such cryptosystems

Understanding Systems

Chor-Rivest Knapsack System
Benny Chor and Ronald L. Rivest
(1984)

→

Solved by
Serge Vaudenay
(1998)

Powerline System
Hendrik W. Lenstra Jr.
(1991)

→

?

Quick Finite Field Intro

FF(p^h) = { $a_0 + a_1x + a_2x^2 + \dots + a_{h-1}x^{h-1}$ }, where a_i are in **FF**(p), so if p is an integer, { 1, 2, 3, … , p-1 }

Multiplicative generator: an element g such that g^n produces every element of the finite field except the zero element.

Chor-Rivest Knapsack

Public Key:

- **FF**(p^h), where p is prime and $h \leq p$
- $\{c_0, \ldots, c_{p-1}\}$, where $c_i = \log_g(x+i) + d$

Private Key:

- random multiplicative generator g of **FF**(p^h)
- random integer $0 \leq \mathbf{d} \leq p^h-2$

Chor-Rivest Knapsack

Encryption:

For message *m* with weight *h*, $\mathbf{E(m) = \Sigma m_i c_i}$

Decryption:

Factor the expression:

$$g^s \bmod f(x) + f(x),$$

where $\mathbf{s = E(m) - hd}$

The roots of the linear factors contain the message.
Factoring polynomials is easy.

Powerline System

Public Key:

- **FF**(q) and **FF**(q^h), where $q = p^n$
- Random set $S = \{ 1, 2, 3, \dots , s \}$ where $s \le q$
- $\{c_1, \dots, c_s\}$, $c_i = (ux-u\pi(i))^k$ of **FF**(q^h)

Private Key:

- Random element $u \in$ **FF**(q^h)
- random integer $1 \le k \le q^h-1$, where $\gcd(k, q^h-1) = 1$
- Map $\pi: S \rightarrow$ **FF**(q)

Powerline System

Encryption:

For message m with weight h, $\mathbf{E(m) = \prod c_i^{m_i}}$ **of FF(q^h)**

Decryption:

Factor the expression:

$$\mathbf{E(m)^l u^{-h} + f(x)},$$
$$\text{where } \mathbf{kl = 1 \bmod (q^h-1)}$$

The roots of the linear factors contain the message.
Factoring polynomials is easy.

Conclusion

Chor-Rivest Knapsack System
Benny Chor and Ronald L. Rivest
(1984)

Solved by
Serge Vaudenay
(1998)

Powerline System
Hendrik W. Lenstra Jr.
(1991)

Discrete logarithm problem

Acknowledgments

Thank you ...
Professor Wim van Dam,
CNSI and EUREKA,
and Arica Lubin.

Made possible in part by
an NSF Career grant

Questions?