RSA: From Theory to Practice

CS5431

February 17, 2017
RSA
Finite Math

1 + 4 = 0
2 + 4 = 1
3 * 4 = 2
4 * 4 = 1
RSA

- Pick primes $p, q$
- Choose $e, d$ such that $ed = 1 \mod (p - 1)(q - 1)$
- $PK = (n, e)$
- $SK = (p, q, d)$

$$c = m^e \mod n$$

$$m = c^d \mod n$$
Chinese Remainder Algorithm

- Pick primes $p, q$
- Choose $e, d$ such that $ed = 1 \mod (p - 1)(q - 1)$
- $PK = (n, e)$
- $SK = (p, q, q^{-1} \mod p, d, d \mod p - 1, d \mod q - 1)$

\[ c = m^e \mod n \]

\[
\begin{align*}
m_1 &= c^d \mod p^{-1} \mod p, \\
m_2 &= c^d \mod q^{-1} \mod q, \\
h &= q^{-1}(m_1 - m_2) \mod p, \\
m &= m_2 + hq
\end{align*}
\]
Square-and-Multiply

res = 1;
while (exp > 0) {
    if (exp % 2 == 1){
        res = res * base % p;
    }
    base = base^2 % p;
    exp >> 1;
}
return res;
Beyond Theory…
Choosing Good Keys
Randomness Quality

```cpp
int getRandomNumber()
{
    return 4;  // chosen by fair dice roll.
    // guaranteed to be random.
}
```
Netscape
Debian Randomness

In the rush to clean up the Debian-OpenSSL fiasco, a number of other major security holes have been uncovered:

<table>
<thead>
<tr>
<th>Affected System</th>
<th>Security Problem</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fedora Core</td>
<td>Vulnerable to certain decoder rings</td>
</tr>
<tr>
<td>Xandros (EEE PC)</td>
<td>Gives root access if asked in stern voice</td>
</tr>
<tr>
<td>Gentoo</td>
<td>Vulnerable to flattery</td>
</tr>
<tr>
<td>OLPC OS</td>
<td>Vulnerable to Jeff Goldblum’s Powerbook</td>
</tr>
<tr>
<td>Slackware</td>
<td>Gives root access if user says Elvish word for “friend”</td>
</tr>
<tr>
<td>Ubuntu</td>
<td>Turns out distro is actually just Windows Vista with a few custom themes</td>
</tr>
</tbody>
</table>
The Internet of Keys
Padding

\[ m \quad 000 \quad r \]

\[ X_0 \quad X_1 \]
Side Channels

- Power
- Timing
- EM Radiation
- Acoustics
Blinding