THE DISCRETE LOG IS VERY DISCREET

AVITAL SCHWARTZ, ADI SHAMIR
APPLIED MATH DEPT
THE WEIZMANN INSTITUTE OF SCIENCE
ISRAEL

THE GENERAL PROBLEM:

LET $f$ BE A ONE WAY FUNCTION:

$x \rightarrow f(x)$ \quad easy

$\rightarrow $ $x$ \quad difficult

HOWEVER, PARTIAL INFORMATION ABOUT $x$ MAY LEAK OUT. LET $B$ BE A PREDICATE:

$f(x) \rightarrow B(x)$

TYPICAL CLASSES OF BOOLEAN PREDICATES:

- PARTICULAR BITS
- RELATIONSHIP BETWEEN BITS
- ARBITRARY PREDICATES ON A SUBSET OF BIT

THE PROBLEM CONSIDERED IN THIS TALK:

$f_{g, n}(x) = g^x \pmod{n}$

WHERE:

- $n$ IS A BLUM INTEGER (WITH UNKNOWN FACTORIZATION):
  $n = p \cdot q$, $p, q$ PRIMES, $|p| = |q|$, $p = q = 3 \pmod{4}$.

- $g$ IS A QUADRATIC RESIDUE MODULO $n$, WHICH IS EITHER RANDOMLY CHOSEN OR GUARANTEED TO HAVE A HIGH ORDER

WE WANT TO PROVE:

FOR ANY BOOLEAN PREDICATE $B$ OF THE RIGHT HALF OF THE BINARY REPRESENTATION OF $x$,

$B(x)$ CANNOT BE APPROXIMATED WITH A NON-NEGligible ADVANTAGE FOR RANDOMLY CHOSEN INPUTS $g, n, g^x \pmod{n}$

UNDER THE SOLE ASSUMPTION THAT POLYSIZE CIRCUITS CANNOT FACTOR A NON-NEGligible FRACTION OF BLUM INTEGERS.
THE NEW IDEA:

FOR A PRIME MODULUS \( n \):
\[
g^n = g^1 \pmod{n}
\]

FOR A COMPOSITE MODULUS \( n \):
\[
g^n = g^{n - \phi(n)} = g^{n - (p-1)(q-1)}
\]
\[
= g^{n - pq + p + q - 1} = g^{p + q - 1} \pmod{n}
\]

(UNLESS THE ORDER OF \( g \) IS EXTREMELY SMALL)

LET \( S = p + q - 1 \). THEN:

- \( S \) IS SMALL (HALF SIZE NUMBER).
- \( g^S \pmod{n} \) IS EASY TO COMPUTE.
- KNOWLEDGE OF \( S \) IMPLIES FACTORING.

THE MIDDLE BIT OF THE DISCRETE LOG CANNOT BE COMPUTED.

\[
S = \begin{array}{c}
\vdots \\
\hline \\
\vdots
\end{array}
\]

THE BINARY REPRESENTATION OF THE DISCRETE LOG OF \( g^m \pmod{n} \).

THIS REPRESENTATION CAN BE MOVED LEFT \( i \) BITS BY CONSIDERING THE DISCRETE LOG OF \( g^{\frac{m}{2}} \pmod{n} \), PROVIDED THAT \( i < \frac{\log_2 m}{2} \).

BY VIEWING THE MIDDLE BIT THROUGHOUT THE LEFT-SHIFT PROCESS, WE CAN GET ALL OF \( S \) AND THUS

THE SOLUTION:

- FIRST, MOVE \( S \) ALL THE WAY TO THE LEFT:
\[
\frac{m}{2} \cdot S
\]

- RANDOMIZE SUFFICIENTLY MANY TIMES TO FIND THE LSB OF \( S \) WITH OVERWHELMING PROBABILITY
- ZERO THIS BIT IN \( S \), AND REPEAT WITH FEWER LEFT SHIFTS
The right hand side bits cannot be approximated:

- We can find all the bits of $s$ to the right of the window.
- To shift $s$ to the right of its original position, we have to extract square roots.
- But this is difficult when the factorization of $m$ is unknown.

The solution:

We randomize the choice of $g$:

- $g_0$ is random.
- $g_{i+1} = g_i^2 \pmod{m}$
- $g = g_k$ for $k \approx \frac{m}{2}$

Squaring modulo Blum integers is a permutation over quadratic residues, so $g$ is random.

Now $\sqrt{g^s} \pmod{m}$ can be computed as $g^s_{k-1}^k$, its square root is $g^s_{k-1}$, etc.

The left hand side bits cannot be approximated:

We cannot prevent the effects of the carries on the window values:

We guess and zero the $O(\log \log m)$ bits in the shifted $s$ which are to the right of the window.

For each bit position and each guessed value of the $O(\log \log m)$ bits we get some prediction of one bit in $s$.

We can chain these predictions from right to left in a unique way:

```
ASSUMPTION
0 ← 11010
1 ← 01101
1 ← 10110
0 ← 11011
```

Conclusion: 011011010

We get $O(\log m)$ possible bit strings.
OVERALL METHOD FOR LEFT HAND SIDE WINDOWS:

\[ e^{\log n} \quad (1-e)\log n \]

Divide \( S \) into \( \frac{1}{e} \) pieces. Get \( O(\log n) \) possible values for each piece. Combine into \( O(\log n^{\frac{1}{e}}) \) possible values. Use exponentiation to pick the correct value.

Open problem: How secure are the bits between \( \log n \) and \( e\log n \)?

SIMULTANEOUS BIT SECURITY:

To prove that the right half of the bits are simultaneously secure, it suffices to show that none of them can be approximated even when given all the bits to their right.

Assume this is not the case:

Given all these bits

We can approximate the \( i \)-th bit for \( i \leq n \).

BUT IN OUR PROOF TECHNIQUE THE RIGHT HAND BITS ARE ACTUALLY KNOWN:

\[
\begin{align*}
L: S & \\
R & \\
\end{align*}
\]

So we could use the advantage to compute \( S \) and thus factor \( n \) with high probability.