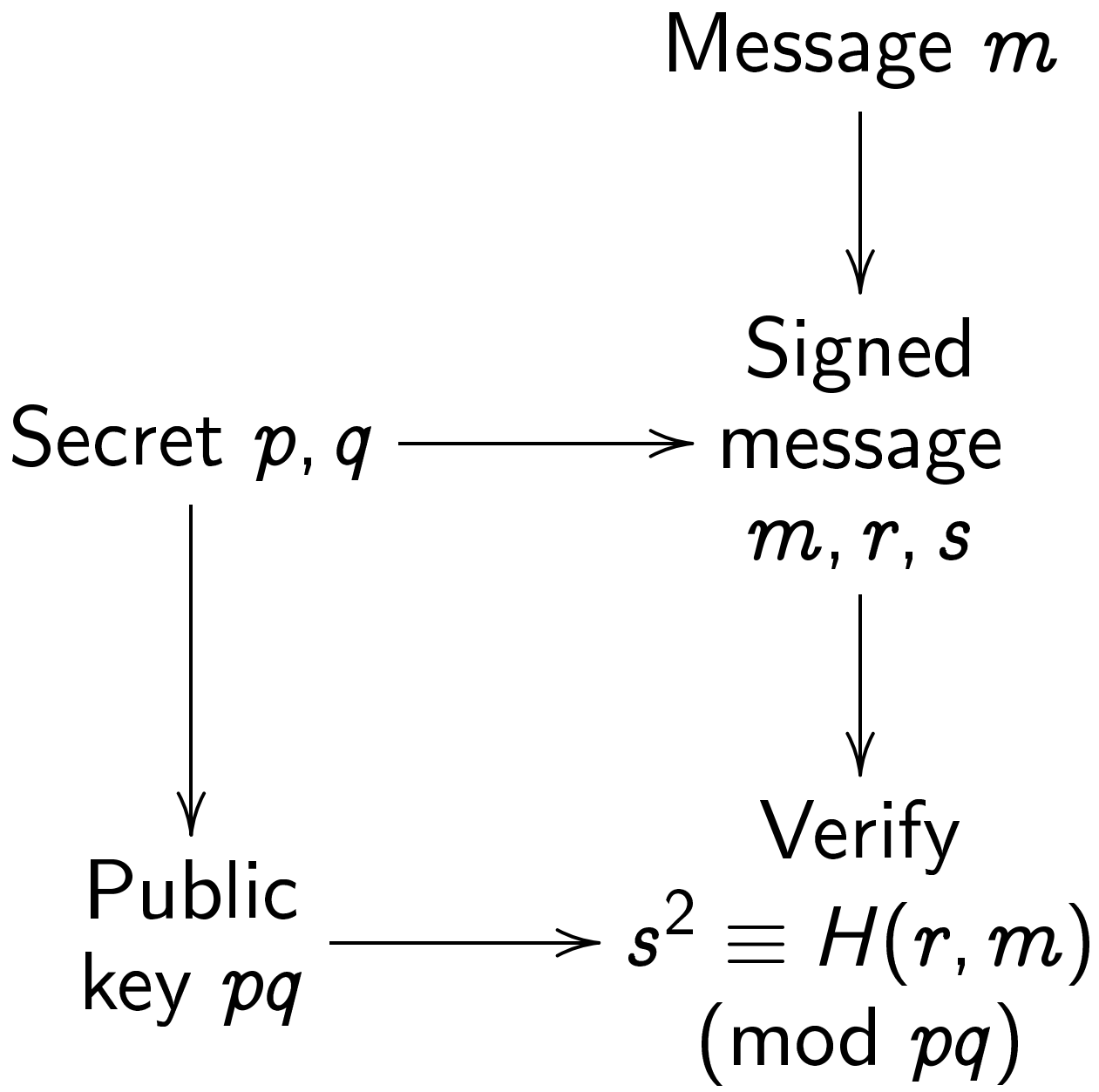
Compressing RSA keys and signatures

D. J. Bernstein

Thanks to:
University of Illinois at Chicago
NSF CCR–9983950
Alfred P. Sloan Foundation
Math Sciences Research Institute
University of California at Berkeley
American Institute of Mathematics

Rabin's public-key signature system



H is a public hash function.

Example: $p = 528763$, $q = 320687$.
Publish $pq = 169567420181$.

To sign $m =$ "Bid \$500 for a T30":
Choose random $r = 202008969701$.
Compute $H(r, m) = 93832038350$.
Use p, q to find $s = 108506016599$.
(May have to try several r's.)

Anyone can compute
$s^2 = 11773555638182463526801$,
$s^2 \bmod pq = 93832038350$,
$H(r, m) \bmod pq = 93832038350$.

Scale up, seems hard to break:
$p \approx 2^{768}$, $q \approx 2^{768}$, $pq \approx 2^{1536}$.

Public key pq has 1536 bits.
Signature r, s has 1600 bits
if randomizer r has 64 bits.
Key+signature: 3136 bits.

Verification: Square 1536-bit s;
subtract $H(r, m)$;
divide by 1536-bit pq.

Can use RSA instead of Rabin:
s^3 instead of s^2.

Application: DNS security

Client to `.com` server,
which has public key 72637729 . . .:
"Where is `www.aol.com`?"

`.com` server to client:
"Ask the `aol.com` server,
which has public key 86186124 . . .,
at IP address 152.163.159.232.
Signed, 1514147951 . . ."

Client to `aol.com` server:
"Where is `www.aol.com`?"

etc.

To prevent forgeries,
client checks that 1514147951 . . .
is a signature of “Ask . . . 232”
under public key 72637729 . . ..

Verification must be fast.

Also, need short keys and signatures:
have only 4096 bits in a DNS packet.

(Splitting data, keys, signatures
into multiple packets
would slow protocol down,
require more software changes,
and allow easy denial of service.)

Speeding up verification

(Bernstein 1997)

Expand signature to (r, s, t)
where $t = (s^2 - H(r, m))/pq$.

Randomized verification:
Choose random 128-bit prime v.
Reduce $H(r, m), pq, s, t$ modulo v
to obtain $\underline{h}, \underline{n}, \underline{s}, \underline{t}$.
Accept if v divides $\underline{s}^2 - \underline{h} - \underline{tn}$.

Much faster, but longer signatures!

Elliptic-curve signatures

Different signature system
using 224-bit elliptic curves:
seems hard to break;
224-bit keys;
448-bit signatures.
Key+signature: 672 bits.

But verification is much slower.

Can we obtain short key+signature
with reasonably fast verification?

Half-size RSA/Rabin keys

Every user finds p, q so that
$pq = 169567 \cdot 10^6 +$ 6-digit number.
Then transmit 6-digit number.

To find p, q:
Choose random prime p.
Compute $q = \lceil 169567000000/p \rceil$.
If q is not prime, try again.

e.g. after several tries: $p = 427243$;
$q = \lceil 169567000000/p \rceil = 396887$;
$pq = 169567192541$;
transmit 192541.

Scaled up to 1536-bit pq:
Reduce keys to 800 bits with
fast key generation;
or 768 bits with
fairly fast key generation.

(Save a few more bits with
much slower key generation.)

Published in an ISO standard
by Guillou, Quisquater 1991.
Reinvented and patented by
Vanstone, Zuccherato 1994.

More key compression

Can quickly find p, q so that $pq = 16956719 \cdot 10^4 + 4$ digits.

Start with random p_0, q_0 so that $p_0 q_0 = 169567 \cdot 10^6 + 6$ digits;
e.g. $p_0 = 435130$, $q_0 = 389694$,
$p_0 q_0 = 169567550220$.

Consider $p = p_0 + x$, $q = q_0 + y$ where x and y are small.

$(p_0 + x)(q_0 + y) - 169567195000$
$= 389694x + 435130y + xy + 355220.$

Use "lattice reduction"
to find small x, y with small
$389694x + 435130y + 355220$:
$x = 27$, $y = -25$.

Take $p = p_0 + 27 = 435157$,
$q = q_0 - 25 = 389669$;
then $pq = 169567193033$.
Stop if p, q are prime.

Scaled up to 1536-bit pq:
Reduce keys to 512 bits with
tolerably fast key generation.
(Coppersmith 2003)

In general, compress to 1/3 size.

Can do better: roughly 1/4 size.
(Elkies)

Signature compression

Can quickly find s
with $s^3 \equiv h \pmod{pq}$,
given h, pq, and
$\approx 2/3$ of the top bits of s.

Can quickly find s
with $s^2 \equiv h \pmod{pq}$,
given h, pq, and
$\approx 1/2$ of the top bits of s.

(Coppersmith 1996,
using lattice reduction)

Transmit $> 1/2$ of the top bits of s.
Recipient can recover all of s.

Very fast, given $2/3$ of the bits.
(Bernstein)

Better compression method
reduces to $1/2$ size
with very fast decompression.
(Bleichenbacher)

With $pq = 169567420181$,
compressing $s = 108506016599$:

Transmit 6-digit x so that
$|xs - ypq|$ is at most $\sqrt{pq}$.

To compute $x = 378877$:
$s - pq = -61061403582$;
$2s - pq = 47444613017$;
$3s - 2pq = -13616790565$;
$11s - 7pq = 6594241322$;
$25s - 16pq = -428307921$;
$\ldots$;
$378877s - 242443pq = 37140$.

Given $x = 378877$,
$pq = 169567420181$,
$H(r, m) = 93832038350$:

Square x, multiply by $H(r, m)$,
divide by pq:
$x^2 H(r, m) \bmod pq = 1379379600$.
Compute $\sqrt{1379379600} = 37140$.

Declare signature valid;
can reconstruct s if desired.