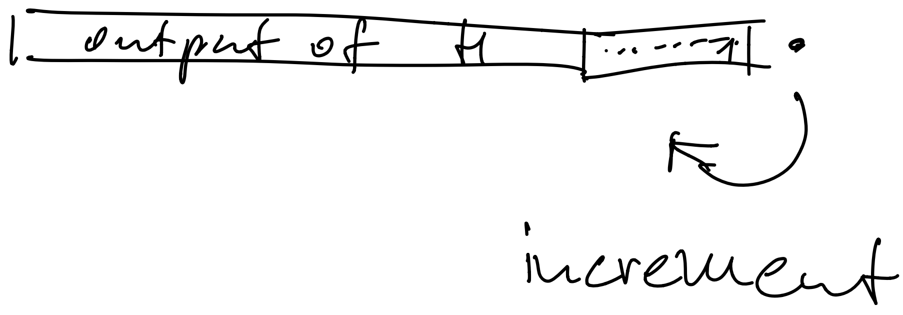
Cryptographic Protocols 2.6.21

RSA accumulator

Let $H: \{0,1\}^* \longrightarrow \mathbb{N}$ (primes!)

Idea:



- KeyGen()

$p, q \leftarrow$ primes

$pk \leftarrow p \cdot q \quad (:= N)$

$sk \leftarrow \varphi(N) = (p-1)(q-1)$

- **Init** (N, X) $\qquad X = (x_1, \ldots, x_n)$

```
$(x_1, \ldots, x_n) \leftarrow X$
$r \leftarrow \mathbb{Z}_N$
$\alpha \leftarrow r^{\prod_{i=1}^{n} H(i \| x_i)} \bmod N$
for $i = 1, \ldots, n$ do
    $w_i \leftarrow \alpha^{1/H(i \| x_i)} \bmod N$
    // to compute $H(i \| x_i)^{-1} \bmod \varphi(N)$,
       need secret key sk
    // $w_i$ is witness for $x_i$
return $(\underbrace{(x_1, \ldots, x_n; w_1, \ldots, w_n)}_{\bar{X}}, \alpha)$
```

- **Query** $(\bar{x}, \alpha, q)$

```
/where $q$ = read$(i)$
return $(x_i, w_i)$
```

- Verify (N, α, q, x_i, w_i)

  return $w_i^{H(i \| x_i)} \overset{?}{\equiv} \alpha \pmod{N}$

Properties

Completeness

Clear, from scheme, because

$$h_i \leftarrow H(i \| x_i)$$

and

$$w_i^{h_i} = w_i^{H(i \| x_i)} = \alpha$$

Security

using strong RSA assumption:

given x, produce z, y s.t.

$$z^y \equiv x \pmod{N}$$

is infeasible.

Let $\alpha \equiv r^{\pi_i H(i \| x_i)} \pmod{N}$.

Suppose A produces $\tilde{x}_i$, $\tilde{w}_i$ with $\tilde{x}_i \neq x_i$, such that

$$\text{Verify}(\ldots, \tilde{x}_i, \tilde{w}_i) = 1.$$

Then

$$\tilde{w}_i^{\tilde{h}_i} \equiv \alpha \pmod{N} \quad \circledast$$

with $\tilde{h}_i = H(i \| \tilde{x}_i)$

But $\tilde{h}_i \neq H(i \| x_i)$.

Then $\circledast$ contridicts the Strong RSA assumption.

- Update(sk, $\bar{x}$, α, u)

/where u = write(i, v)

$sk = \varphi(N)$

$$\alpha' \leftarrow \alpha^{\underbrace{\left(1/H(i\|x_i)\cdot H(i\|v)\right)}_{\text{in } \mathbb{Z}_{\varphi(N)}}} \bmod N$$

needs sk with $\varphi(N)$.

Update or recompute from scratch all n witnesses $w_1, \dots, w_n$.

return (i, v, $(w_1, \dots, w_n)$, α')

- Refresh(pk, $\bar{x}$, α', u)

u = write(i, v)

$x_i \leftarrow v$

$\vdots$

⋮

Recompute all witnesses

```
for $i = 1, \dots, n$ do
    $w_i \leftarrow r^{\prod_{j=1, j \neq i}^{n} H(j \| X_j)} \pmod{N}$
```

This is __expensive__!

Properties

Efficiency

- Query and Verify take const. number of op.s

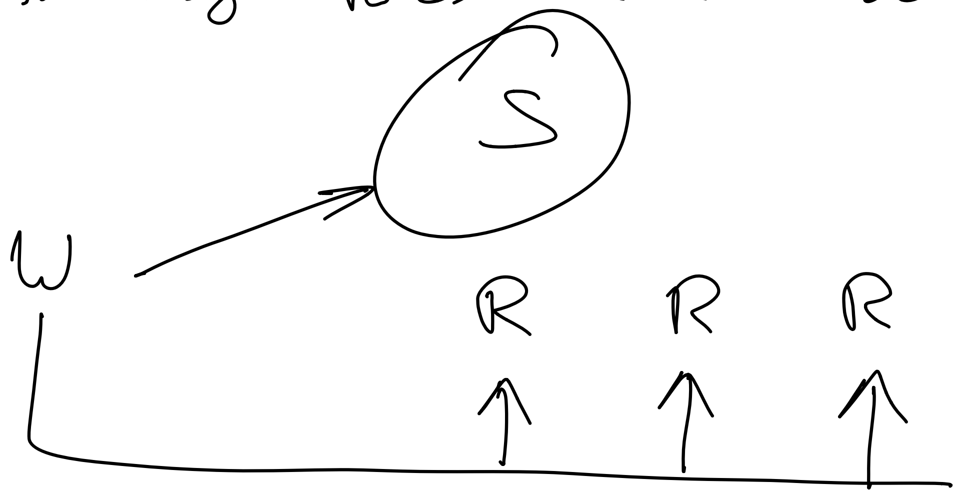
+ Update and Refresh take $O(n)$

__Space__

- $O(n)$ extra space

Trivial authenticated data structure

- Writer signs every value x_i
- All signatures are stored at S



To prevent replay attks, and to ensure freshness, writer also needs a timestamp (ts).
Counts write op.

Each update signs again all x_i
as $\sigma_i \leftarrow \mathrm{sign}(sk, i \| ts \| x_i)$

Comparison of ADS

Scheme	Update time	Refresh time	Verify time	Proof size
Hash tree	$O(\log n)$	$O(\log n)$	$O(\log n)$	$O(\log n)$
Accumulator	$O(1)$	$\underline{O(n)}$	$O(1)$	$O(1)$
Trivial signatures	$O(n)$	$O(n)$	$O(1)$	$O(1)$

In practice, hash trees are preferred in almost all applications.