EECS 575: Advanced Cryptography
Fall 2022
Lecture 19

Paul Grubbs

paulgrub@umich.edu

Beyster 4709

Agenda for this lecture

- Announcements
- Recap from last time
- Collision-resistant hashing
- one-time->many-time signatures
- Random oracle model
- Signatures from trapdoor permutations

Agenda for this lecture

- Announcements
- Recap from last time
- Collision-resistant hashing
- one-time->many-time signatures
- Random oracle model
- Signatures from trapdoor permutations

Announcements

- HW5 online, due 11/18 → will be on canvas
- Election!

Agenda for this lecture

- Announcements
- Recap from last time
- Collision-resistant hashing
- one-time->many-time signatures
- Random oracle model
- Signatures from trapdoor permutations

Digital Signatures

$\mathrm{SUFCMA}^{\mathcal{A}}_{\mathrm{SIG}}$:

$(pk, sk) \leftarrow Gen$; $Q = \{\}$

$(m', \sigma') \leftarrow \mathcal{A}^{Sign(\cdot)}(pk)$

Ret $Success(pk, Q, m, \sigma)$

Sign(m):

$\sigma \leftarrow Sign(sk, m)$

$Q[m] := \sigma$

Ret σ

Success(pk, Q, m, σ):

Ret $Ver(pk, \sigma, m) = 1$

$\wedge\ m \notin Q \quad / (m, \sigma) \notin Q$

→ Authenticity with public verif

SIG = (Gen, Sign, Verify)

Sigs from owFs

→ Alexandra, in discussion

Agenda for this lecture

- Announcements
- Recap from last time
- Collision-resistant hashing
- one-time->many-time signatures
- Random oracle model
- Signatures from trapdoor permutations

Collision-resistant hash functions (CRHFs)

- Compress long message into short digest.
  hard to "collide" inputs

$$h_s : \{D_s \rightarrow R_s\}_s$$

Say h_s is CRHF family if $\forall s, \forall n \; |R_s| < |D_s|$ (1)

(2) $\forall$ nupp A,

$$\Pr_{k \leftarrow \mathrm{Gen}(1^n)}[(x, x') \leftarrow A(h) : x \neq x', h(x) = h(x')] = \mathrm{negl}(n)$$

Collision-resistant hash functions (CRHFs)

- SHA256, SHA1, MD5, SHA3,
  SHA — Broken — "new" NIST hash

NO hash key sampling

???

Can't construct advice string

"Formalizing Human Ignorance" - Rogaway

Agenda for this lecture

- Announcements
- Recap from last time
- Collision-resistant hashing
- one-time->many-time signatures
- Random oracle model
- Signatures from trapdoor permutations

Many-time signatures via CRHFs

$OTS'[OTS, h_S]$:

- $OTS'[OTS, h_S].Gen$:
  $h \leftarrow \{h_S\}$
  $(pk, sk) \leftarrow OTS.Gen$
  $Ret((pk, h), (sk, h))$
- $Sign((sk, h), m)$:
  $Ret\ OTS.Sign(sk, h(m))$
- $Ver((pk, h), \sigma, m)$:
  $Ret\ OTS.Ver(pk, \sigma, h(m))$

- Gen: sample $x^{i,b} \leftarrow \{0,1\}^n$ for $i \in [n]$ $b \in \{0,1\}$
  $Ret\ ([F(x^{i,b})]_{\substack{i \in [n] \\ b \in \{0,1\}}}, [x^{i,b}]_{\substack{i \in [n] \\ b \in \{0,1\}}})$
- $Sign(sk, m)$:
  $Ret\ [x^{i,m_i}]_{i \in [n]}$
- $Ver(pk, \sigma, m)$:
  $Ret\ \bigwedge_{i=1}^{n} F(\sigma_i) = pk^{i,m_i}$

OTS signs n-bit msgs

$D_S \rightarrow$ n-bit

OTS' signs $\in D_S$

"Hash-then-Sign"

Use collision

Many-time signatures via CRHFs

MTS[OTS]: (OTS sigs |PK| + n bits)

- Gen: Ret OTS.Gen
- Sign(sk_i, m_i):
  - $(pk_{i+1}, sk_{i+1}) \leftarrow$ OTS.Gen
  - $\sigma_i \leftarrow$ OTS.Sign(sk_i, $pk_{i+1} \| m_i$)
  - Ret $(pk_{i+1}, \sigma_i, m_i, \ldots, pk_2, \sigma_1, m_1, vk_1)$
- Vcr(pk, $(pk_{i+1} \ldots pk_1)$):
  - $pk = pk_1$
  - OTS.Vcr(pk_j, σ_j, $m_j \| pk_{j+1}$) $\forall j$

MTS[OTS]
UF-CMA if
OTS UF-1CMA
+ h_S is CR

- Stateful Signer
- linear Vcr
  - Fixable

SPHINCS+ - NIST PQ Sig

Agenda for this lecture

- Announcements
- Recap from last time
- Collision-resistant hashing
- one-time->many-time signatures
- Random oracle model
- Signatures from trapdoor permutations

Random oracle model

"heuristic"

- All parties have access to same global rand. f'n

H^K rand. oracle

$F^H[\ldots]$

- RO methodology Bellare-Rogaway '93

ROM

uninstantiable

[CGH00]

- Design scheme in ROM
- Analyze security in ROM
- Instantiate with strong hash f'n (SHA-256) = not indiff

Not a huge concern; artificial

Most RO schemes believed secure

Prove indifferentiable from RO

Strong "structural" security

Agenda for this lecture

- Announcements
- Recap from last time
- Collision-resistant hashing
- one-time->many-time signatures
- Random oracle model
- Signatures from trapdoor permutations

Trapdoor permutations

↳ owp with trapdoor for inverting

$$\{F_s : D_s \longrightarrow D_s\}$$

- $(s, t) \leftarrow S(1^n)$

  s: index of f'n; t: trapdoor

$\forall$ n-ppt A,

$$\Pr_{(s,t)\leftarrow S(1^n)}[\,X \leftarrow D_s : A(s, f_s(X)) = X\,] = \text{negl}(n)$$

- compute f_s efficiently, with t, f_s^{-1} efficiently.

Signatures from TDPs ("FDH")

$SIG^H[F_S]$:

- Gen: Ret $(S, t) \leftarrow SC(1^n)$
- $Sign^H(sk = (S, t), m)$:
  Ret $F_S^{-1}(H(m))$
- $Ver^H(pk = (S), \sigma, m)$:
  Ret $F_S(\sigma) \stackrel{?}{=} H(m)$

Note:

If $H: \{0,1\}^* \to D_S$, SIG signs $\{0,1\}^*$

- Unique sigs
- UF $\Rightarrow$ SUF

Analyzing FDH

- Why is this UF-CMA?

To forge, (m, σ)

$f_s(\sigma) = H(m)$

$\sigma = f_s^{-1}(H(m))$

Need to invert on random point!

Thm: If f_s is TDP, then $SIG^H[f_s]$ is UF-CMA modelling H as RO

Analyzing FDH

Thm: If f_S is TDP, then
$SIG^H[f_S]$ is UF-CMA
modelling H as RO

Proof (sketch):

Build reduction B that simulates UF-CMA

B given S, $y = F_S(x)$.

Needs to simulate
Signing oracle
Needs to "program"
random oracle
to be able to
answer Sign queries