Robust NIZK & CCA2 Encryption

Seminar in Cryptographic Protocols

Roy Kasher

Overview

- Construction of CCA1 encryption from NIZK [Naor Yung]
- First construction of CCA2 encryption [DDN]
- Strengthening NIZK [Sahai] [De Santis et al]
  - Non-malleablility, one time simulation soundness, robustness
- Simplified construction of CCA2 encryption from one time simulation sound NIZK [Sahai] [Lindell]
- More strengthening NIZK [De Santis et al]
  - Many time simulation soundness

Preliminaries - PKE

- Types of attacks:
  - *Chosen plaintext attack (CPA)*
    Adversary has access to encryption oracle
  - *Passive chosen ciphertext attack (CCA1)*
    Adversary has access to decryption oracle, prior to encryption ("lunchtime attack")
  - *Adaptive chosen ciphertext attack (CCA2)*
    Adversary has unlimited access to decryption oracle
- Strongest security: Existential unforgeability

Preliminaries - IND Security

- Let (E,D,G) be a triplet of PPT algorithms
  - IND CPA Game 0:
    - $(pk,sk) \leftarrow Gen(1^n)$
    - $(m_0,m_1) \leftarrow A(pk)$
    - $c \leftarrow E_{pk}(m_0)$
    - $b \leftarrow A(pk, c)$
  - IND CPA Game 1:
    - $(pk,sk) \leftarrow Gen(1^n)$
    - $(m_0,m_1) \leftarrow A(pk)$
    - $c \leftarrow E_{pk}(m_1)$
    - $b \leftarrow A(pk, c)$
- For any PTT A, $|Pr_0\,[b=1] - Pr_1[b=1]| < \nu(n)$

Preliminaries - IND Security

- Let (E,D,G) be a triplet of PPT algorithms
  - IND CCA1 Game 0:
    - $(pk,sk) \leftarrow Gen(1^n)$
    - $(m_0,m_1) \leftarrow A^{D_{sk}}(pk)$
    - $c \leftarrow E_{pk}(m_0)$
    - $b \leftarrow A(pk, c)$
  - IND CCA1 Game 1:
    - $(pk,sk) \leftarrow Gen(1^n)$
    - $(m_0,m_1) \leftarrow A^{D_{sk}}(pk)$
    - $c \leftarrow E_{pk}(m_1)$
    - $b \leftarrow A(pk, c)$
- For any PTT A, $|Pr_0[b=1] - Pr_1[b=1]| < \nu(n)$

Preliminaries - IND Security

- Let (E,D,G) be a triplet of PPT algorithms
  - IND CCA2 Game 0:
    - $(pk,sk) \leftarrow Gen(1^n)$
    - $(m_0,m_1) \leftarrow A^{D_{sk}}(pk)$
    - $c \leftarrow E_{pk}(m_0)$
    - $b \leftarrow A^{D_{sk}}(pk, c)$
  - IND CCA2 Game 1:
    - $(pk,sk) \leftarrow Gen(1^n)$
    - $(m_0,m_1) \leftarrow A^{D_{sk}}(pk)$
    - $c \leftarrow E_{pk}(m_1)$
    - $b \leftarrow A^{D_{sk}}(pk, c)$
- For any PTT A, $|Pr_0 [b=1] - Pr_1[b=1]| < \nu(n)$
- Recall: CCA2 security is equivalent to non-malleability

Preliminaries - Adaptive NIZK

- A pair of PPT (P, V) is an adaptive non-interactive proof system for a language $L \in NP$ if it satisfies:
  - *Completeness*: For all $(x,w) \in R_L$,
    $Pr[r \leftarrow \{0,1\}^*; \Pi \leftarrow P(r,x,w): V(r,x,\Pi)=1] = 1$
  - *Adaptive soundness*: For all $x \notin L$, PPT A,
    $Pr[r \leftarrow \{0,1\}^*; (x, \Pi) \leftarrow A(r): V(r,x,\Pi)=1] < \nu(n)$

Preliminaries - Adaptive NIZK

- A pair of PPT (P, V) is an adaptive non-interactive *zero knowledge* proof system for a language $L \in NP$ if it is adaptive NIP, and in addition, satisfies:
  - *Adaptive zero-knowledge*: There exists PPT sim. S such that the distributions $\{r,x,\Pi\}$ are indistinguishable in the following two games, for any PPT adversary A:

- ZK real:
  - $r \leftarrow \{0,1\}^{poly(n)}$
  - $(x,w) \leftarrow A(r)$
  - $\Pi \leftarrow P(r,x,w)$
- ZK sim:
  - $r \leftarrow S(1^n)$
  - $(x,w) \leftarrow A(r)$
  - $\Pi \leftarrow S(r,x)$

CCA1 Encryption - Construction

- Due to Naor and Yung
- Let (P,V) be an adaptive NIZK proof system, and (E,D,G) an IND-CPA secure encryption scheme
  - **Key Generation**: Obtain two independent keys from G, and choose random reference string.
  - **Encryption**: Encrypt m twice, once with each public key. Prove consistency of encryptions.
  - **Decryption**: Verify the proof is accepting, and decrypt one of the ciphertexts using the matching key

CCA1 Encryption - Construction

- $G^*(1^n)$:
  - $(pk_1,sk_1), (pk_2,sk_2) \leftarrow G(1^n)$
  - $r \leftarrow \{0,1\}^{poly(n)}$
  - $pk^* = (pk_1,p,k_2,r)$
  - $sk^* = (sk_1,sk_2)$
- $D^*(c_1,c_2,\Pi)$:
  - Verify $V(r,(c_1,c_2,pk_1,pk_2),\Pi) = 1$
  - Output $D_{sk_1}(c_1)$
- $E^*(m)$:
  - $c_1 = E_{pk_1}(m;w_1)$, $c_2 = E_{pk_2}(m;w_2)$
  - $\Pi = P(r,(c_1,c_2,pk_1,pk_2),(m,w_1,w_2))$ Proof checks that both ciphertexts encrypt same msg
  - Output (c_1,c_2,Π)

CCA1 Encryption

- Cryptosystem based on *secret hiding* principle:
  - Introduced by Feige and Shamir
  - System has two “secrets”
  - In order to operate it, only one of the secrets needs to be known (Decryption with one key; Verification public)
  - To an outsider, it should be indistinguishable which of the secrets is known

CCA1 Encryption - Proof

- Want to show games are indistinguishable
  - Game 0:
    - $pk^* = (pk_1, pk_2, r_{uni})$
      $sk^* = (sk_1, sk_2)$
    - $(m_0, m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
    - $c_1 = E_{pk_1}(m_0; w_1)$
      $c_2 = E_{pk_2}(m_0; w_2)$
      $\Pi = P(r, (c_1, c_2), (m_0, w_1, w_2))$
    - $b \leftarrow A(c_1, c_2, \Pi)$
  - Game 1:
    - $pk^* = (pk_1, pk_2, r_{uni})$
      $sk^* = (sk_1, sk_2)$
    - $(m_0, m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
    - $c_1 = E_{pk_1}(m_1; w_1)$
      $c_2 = E_{pk_2}(m_1; w_2)$
      $\Pi = P(r, (c_1, c_2), (m_1, w_1, w_2))$
    - $b \leftarrow A(c_1, c_2, \Pi)$
- Problem: Adversary against CPA cannot simulate proof

CCA1 Encryption - Proof

- … Except that by definition of NIZK, he can:
  - Game 0:
    - $pk^* = (pk_1,pk_2,r_{uni})$
      $sk^* = (sk_1,sk_2)$
    - $(m_0,m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
    - $c_1 = E_{pk_1}(m_0;w_1)$
      $c_2 = E_{pk_2}(m_0;w_2)$
      $\Pi = P(r,(c_1,c_2),(m,w_1,w_2))$
    - $b \leftarrow A(c_1,c_2,\Pi)$
  - Game 0_{sim}:
    - $pk^* = (pk_1,pk_2,r_{sim})$
      $sk^* = (sk_1,sk_2)$
    - $(m_0,m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
    - $c_1 = E_{pk_1}(m_0;w_1)$
      $c_2 = E_{pk_2}(m_0;w_2)$
      $\Pi = S(r,(c_1,c_2))$
    - $b \leftarrow A(c_1,c_2,\Pi)$
- Next: Second encryption with m_1 instead of m_0

CCA1 Encryption - Proof

- Easy, because decryption oracle uses sk_1:
  - Game 0_{sim}:
    - $pk^* = (pk_1, pk_2, r_{sim})$
      $sk^* = (sk_1, sk_2)$
    - $(m_0, m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
    - $c_1 = E_{pk_1}(m_0; w_1)$
      $c_2 = E_{pk_2}(m_0; w_2)$
      $\Pi = S(r, (c_1, c_2))$
    - $b \leftarrow A(c_1, c_2, \Pi)$
  - Game $0/1_{sim}$:
    - $pk^* = (pk_1, pk_2, r_{sim})$
      $sk^* = (sk_1, sk_2)$
    - $(m_0, m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
    - $c_1 = E_{pk_1}(m_0; w_1)$
      $c_2 = E_{pk_2}(m_1; w_2)$
      $\Pi = S(r, (c_1, c_2))$
    - $b \leftarrow A(c_1, c_2, \Pi)$
- Next: First encryption with m_1 instead of m_0

CCA1 Encryption - Proof

- Problem: Adversary cannot simulate decryption
- Recall: Verifier ensures c_1 and c_2 encrypt same plaintext
- Idea: Decrypt the second message, instead of first
- Fails when proof is *invalid*: $D_{sk_1}(c'_1) \neq D_{sk_2}(c'_2)$ but verify pass

- Game $0/1_{sim}$:
  - $pk^* = (pk_1, pk_2, r_{sim})$
    $sk^* = (sk_1, sk_2)$
  - $(m_0, m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
  - $c_1 = E_{pk_1}(m_0; w_1)$
    $c_2 = E_{pk_2}(m_1; w_2)$
    $\Pi = S(r, (c_1, c_2))$
  - $b \leftarrow A(c_1, c_2, \Pi)$

- $D^*_{sk_1}(c'_1, c'_2, \Pi')$:
  - Verify $V(r, (c'_1, c'_2), \Pi') = 1$
  - Output $D_{sk_1}(c'_1)$

CCA1 Encryption - Proof

- Need to show: A cannot generate *invalid* proofs
- Let's review our games so far

- Game 0:
  - $pk^* = (pk_1, pk_2, r_{uni})$ $sk^* = (sk_1, sk_2)$
  - $m_0, m_1 \leftarrow A^{D^*_{sk_1}}(pk)$
- Game 0_{sim}:
  - $pk^* = (pk_1, pk_2, r_{sim})$ $sk^* = (sk_1, sk_2)$
  - $m_0, m_1 \leftarrow A^{D^*_{sk_1}}(pk)$
- Game $0/1_{sim}$:
  - $pk^* = (pk_1, pk_2, r_{sim})$ $sk^* = (sk_1, sk_2)$
  - $m_0, m_1 \leftarrow A^{D^*_{sk_1}}(pk)$

- In game 0, validity of proofs by adaptive soundness
- $\text{Invalid}_0 \approx \text{Invalid}_{0sim}$ since $r_{uni} \approx r_{sim}$ by ZK
- $\text{Invalid}_{0sim} \approx \text{Invalid}_{0/1sim}$ since games are identical
- Hence, validity of proofs guaranteed

CCA1 Encryption - Proof

- Can now replace decryption oracle

- Game $0/1_{sim}$:
  - $pk^* = (pk_1,pk_2,r_{sim})$
    $sk^* = (sk_1,sk_2)$
  - $(m_0,m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
  - $c_1 = E_{pk_1}(m_0;w_1)$
    $c_2 = E_{pk_2}(m_1;w_2)$
    $\Pi = S(r,(c_1,c_2))$
  - $b \leftarrow A(c_1,c_2,\Pi)$

- Game $0/1_{sim}$ (alt key):
  - $pk^* = (pk_1,pk_2,r_{sim})$
    $sk^* = (sk_1,sk_2)$
  - $(m_0,m_1) \leftarrow A^{D^*_{sk_2}}(pk)$
  - $c_1 = E_{pk_1}(m_0;w_1)$
    $c_2 = E_{pk_2}(m_1;w_2)$
    $\Pi = S(r,(c_1,c_2))$
  - $b \leftarrow A(c_1,c_2,\Pi)$

- (Note we have proved adaptive NIZK is sound against simulated reference strings)

CCA1 Encryption - Proof

- Repeating previous arguments,
  - Game $0/1_{sim}$ (alt key):
    - $pk^* = (pk_1, pk_2, r_{sim})$
      $sk^* = (sk_1, sk_2)$
    - $(m_0, m_1) \leftarrow A^{D^*_{sk_2}}(pk)$
    - $c_1 = E_{pk_1}(m_0; w_1)$
      $c_2 = E_{pk_2}(m_1; w_2)$
      $\Pi = S(r, (c_1, c_2))$
    - $b \leftarrow A(c_1, c_2, \Pi)$
  - Game 1_{sim} (alt key):
    - $pk^* = (pk_1, pk_2, r_{sim})$
      $sk^* = (sk_1, sk_2)$
    - $(m_0, m_1) \leftarrow A^{D^*_{sk_2}}(pk)$
    - $c_1 = E_{pk_1}(m_1; w_1)$
      $c_2 = E_{pk_2}(m_1; w_2)$
      $\Pi = S(r, (c_1, c_2))$
    - $b \leftarrow A(c_1, c_2, \Pi)$

CCA1 Encryption - Proof

- Repeating previous arguments,
  - Game 1_{sim} (alt key):
    - $pk^* = (pk_1,pk_2,r_{sim})$
      $sk^* = (sk_1,sk_2)$
    - $(m_0,m_1) \leftarrow A^{D^*_{sk_2}}(pk)$
    - $c_1 = E_{pk_1}(m_1;w_1)$
      $c_2 = E_{pk_2}(m_1;w_2)$
      $\Pi = S(r,(c_1,c_2))$
    - $b \leftarrow A(c_1,c_2,\Pi)$
  - Game 1_{sim}:
    - $pk^* = (pk_1,pk_2,r_{sim})$
      $sk^* = (sk_1,sk_2)$
    - $(m_0,m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
    - $c_1 = E_{pk_1}(m_1;w_1)$
      $c_2 = E_{pk_2}(m_1;w_2)$
      $\Pi = S(r,(c_1,c_2))$
    - $b \leftarrow A(c_1,c_2,\Pi)$

CCA1 Encryption - Proof

- Repeating previous arguments,
  - Game 1_{sim}:
    - $pk^* = (pk_1, pk_2, r_{sim})$
      $sk^* = (sk_1, sk_2)$
    - $(m_0, m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
    - $c_1 = E_{pk_1}(m_1; w_1)$
      $c_2 = E_{pk_2}(m_1; w_2)$
      $\Pi = S(r, (c_1, c_2))$
    - $b \leftarrow A(c_1, c_2, \Pi)$
  - Game 1:
    - $pk^* = (pk_1, pk_2, r_{uni})$
      $sk^* = (sk_1, sk_2)$
    - $(m_0, m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
    - $c_1 = E_{pk_1}(m_1; w_1)$
      $c_2 = E_{pk_2}(m_1; w_2)$
      $\Pi = P(r, (c_1, c_2), (m, w_1, w_2))$
    - $b \leftarrow A(c_1, c_2, \Pi)$

CCA1 Encryption - Proof

- By "Chain of Indistinguishability" :
  $0 \leftrightarrow 0_{sim} \leftrightarrow 0/1_{sim} \leftrightarrow 0/1_{sim}$ (key) $\leftrightarrow 1_{sim}$ (key) $\leftrightarrow 1_{sim} \leftrightarrow 1$
  => $|Pr_0 [b=1] - Pr_1[b=1]| < \nu(n)$
- This completes the proof of the NY scheme
- Seven game proof from lecture notes of Jonathan Katz
- Naor Yung define parameterized games (b_1,b_2); Use only four games

CCA1 Encryption - Not CCA2

- Unfortunately, the NW scheme is not secure against *adaptive* chosen ciphertext attacks
- Take any adaptive NIZK proof system and modify:
  New prover adds extra bit to proof
  New verifier ignores last bit
- An attacker can request challenge encryption, swap the last bit and query the decryption oracle
- Intuitively, since the proof is malleable, so is the encryption scheme (More on this later…)

CCA1 Encryption - Not CCA2

- Where does our proof break?
- A could not generate invalid proofs due to soundness
- This no longer holds when A is invoked the 2^{nd} time

  - Game $0/1_{sim}$:
    - $pk^* = (pk_1, pk_2, r_{sim})$
      $sk^* = (sk_1, sk_2)$
    - $(m_0, m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
    - $c_1 = E_{pk_1}(m_0; w_1)$
      $c_2 = E_{pk_2}(m_1; w_2)$
      $\Pi = S(r, (c_1, c_2))$
    - $b \leftarrow A^{D^*_{sk_1}}(c_1, c_2, \Pi)$

CCA1 Encryption - Not CCA2

- In fact, this is the only part where our proof fails
- Can we fix this?
- We can, by strengthening the NIZK
- But first let's start from scratch, as this was the chronological order of things

CCA2 - History

- CCA2 definition [Rackoff Simon 91]
- 1st CCA2 based on general assumptions [DDN 91/2000]
- Random Oracle Model [Bellare Rogaway 93]
  - With respect to our model - Heuristic only
- Efficient CCA2 based on DDH [Cramer Shoup 98]
- NY Paradigm + stronger NIZK
  - Non-malleable [Sahai 99]
  - Many time simulation soundness, robust [De Santis et al 01]
  - One time simulation soundness [Lindell 06]
- CCA2 from Identity Based Encryption [Canetti Halevi Katz 05]

CCA2 - DDN

- Dolev, Dwork and Naor 2000
- First construction based on general assumptions
- Exploits intricate interplay between several components
  - Many encryptions
  - NIZK proofs
  - Digital signatures
- Hard to teach in a course on cryptography, for example

CCA2 - DDN (Construction)

- Public key consists of n pairs of public keys, $(pk_{1,0}, pk_{1,1})\ldots(pk_{n,0}, pk_{n,1})$ and a ref string for NIZK
- Encryption:
  - Choose an instance of a *digital signature scheme*
  - View the public verification key as a sequence of bits selecting public encryption keys (vk)
  - Encrypt plaintext under each of the selected keys (C)
  - Provide a NIZK of consistency (Π)
  - Sign on the ciphers and the proof (σ)
  - Ciphertext is a quad (vk, C, Π, σ)

CCA2 - DDN (Intuition)

- Attacker is given ciphertext (vk, C, Π, σ) it wishes to maul
- If attacker uses vk, it will be unable generate a valid signature on any other content
- If attacker changes signature scheme, there will be at least one pair of encryption keys ($pk_{i,0}$, $pk_{i,1}$) so that C contains $E_{pk_{i,0}}(m)$, and the adversary needs $E_{pk_{i,1}}(m')$ for m' related to m. Since keys are chosen independently, he has no idea how to do this

Non Malleable NIZK

- First considered by Sahai as an intuitive interpretation of zero knowledge
- *Non malleability*: What one can prove after seeing a NIZK proof one could also have proved before seeing it (except the ability to duplicate the proof)
- Does not follow from current defs of NIZK:
  - Let $L \in NP$ a hard language, $L' = \{(x,y) \mid x,y \in L\}$
  - Build proof system by concatenation
  - Proof for (x,y) + witness for x' allows proving (x',y)

Non Malleable NIZK

- Many flavours
  - Non malleability
  - Adaptive non malleability
  - Non malleability with respect to multiple proofs
    - Bounded
    - or unbounded
- Consider *adaptive non malleability*: Adversary can ask for a proof of a theorem of its choosing
- Formalization surprisingly hard

Non Malleable NIZK

- Who provides the witness for the proof?
  - **Adversary**: Makes definition trivial
  - **All-powerful party**: Allows adversary to learn which theorems are true
- Alternative: Define non-malleability with respect to *simulated* proofs
- Can consider a similar, yet incomparable, approach: *Simulation soundness*: Adversary cannot prove a false statement, even after seeing simulated proof(s)

Non Malleable NIZK

- Constructions
  - [Sahai 99] *Adaptive non-malleable* and *many time simulation sound* NIZK
  - [De Santis et al 01] *unbounded many time simulation sound* NIZK
  - [Lindell 06] Simple *one time simulation sound* NIZK
- As observed by Sahai, all notions above suffice for constructing CCA2 secure encryption
- Specifically, by plugging in the strong NIZK in the NW construction

1 Time Simulation Soundness

- Let (P,V) be an adaptive NIZK proof system for a language L with simulator S
- We say (P,V,S) is *one-time simulation sound* if for every PPT A, it succeeds in the following experiment with negligible probability:
    - $r \leftarrow S(1^n)$
    - $x \leftarrow A(r)$
    - $\Pi \leftarrow S(x,r)$
    - $(x', \Pi') \leftarrow A(x,r,\Pi)$
    - A wins if $x' \notin L$, $(x',\Pi') \neq (x,\Pi)$ but $V(x',r,\Pi')=1$

NY Revisited

- Couldn't prove A generates verifiable *invalid* proofs $D_{sk_1}(c_1) \neq D_{sk_2}(c_2)$ with negligible probability
  - Game $0/1_{sim}$:
    - $pk^* = (pk_1, pk_2, r_{sim})$
      $sk^* = (sk_1, sk_2)$
    - $(m_0, m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
    - $c_1 = E_{pk_1}(m_0; w_1)$
      $c_2 = E_{pk_2}(m_1; w_2)$
      $\Pi = S(r, (c_1, c_2))$
    - $b \leftarrow A^{D^*_{sk_1}}(c_1, c_2, \Pi)$

- $D^*_{sk_1}(c'_1, c'_2, \Pi')$:
  - Verify $V(r, (c'_1, c'_2), \Pi') = 1$
  - Output $D_{sk_1}(c'_1)$

- This is no longer the case

NY Revisited

- Easily reduced to *one time simulation soundness*
- Adversary receives simulated ref r_{sim}, chooses (m_0,m_1) as below, observes simulated proof Π, and outputs verifiable *invalid* proof
  (In particular, A's output $\neq(c_1,c_2,\Pi)$)
- Result: Can safely replace sk_1 with sk_2

- Game $0/1_{sim}$:
  - $pk^* = (pk_1,pk_2,r_{sim})$
    $sk^* = (sk_1,sk_2)$
  - $(m_0,m_1) \leftarrow A^{D^*_{sk_1}}(pk)$
  - $c_1 = E_{pk_1}(m_0;w_1)$
    $c_2 = E_{pk_2}(m_1;w_2)$
    $\Pi = S(r,(c_1,c_2))$
  - $b \leftarrow A^{D^*_{sk_1}}(c_1,c_2,\Pi)$

NY Revisited

- Rest of proof as before (Verify at home…)
- Note:
  - NW scheme with adaptive NIZK is CCA1 secure
  - NW scheme with adaptive OTSS NIZK is CCA2 secure
- Conclusion:
  - CCA2 secure encryption schemes exist if enhanced trapdoor permutations exist
    - (Late fact: Adaptive NIZK requires enhanced trapdoor permutations)

OTSS - Tools

- For our construction, we will need the following tools:
- **Non-interactive perfectly-binding commitment schemes** satisfying:
  - *Hiding*: it is hard to distinguish $C(s_1)$ from $C(s_2)$
  - *Binding*: $C(s_1;r_1) \neq C(s_2;r_2)$ for every r_1,r_2
  - *Pseudorandom range*: Output should be pseudorandom
  - *Negligible support*: A random string is a commitment with negligible probability
- All properties are easily satisfiable with OWP-based commitment scheme

OTSS - Tools

"Strong" one-time signature schemes

- Triplet of PPT algorithms (G,Sign,Ver)
- *Validity*:

  $$\mathrm{Ver}(vk,m,\mathrm{Sign}(sk,m))=1 \text{ where } (vk,sk)\leftarrow G(1^n)$$

- *Security*: Probability to produce $(m,\sigma)\neq(m',\sigma')$ s.t. $\mathrm{Ver}(vk,m',\sigma')=1$ is negligible
- Constructed using universal one-way hash and 1-1 OWF

- SIGN Game:
  - $(vk,sk) \leftarrow G(1^n)$
  - $m \leftarrow A(vk)$
  - $\sigma \leftarrow \mathrm{Sign}(sk,m)$
  - $(m, \sigma') \leftarrow A(vk,m,\sigma)$

OTSS - Construction

- Reference string is divided into two parts (r_1, r_2)
- Following [FLS], prove a compound statement:
  - Define L': Either $x \in L$ or r_1 has some special property
  - Random r_1 has property with negligible property
  - Simulator generated r_1 does have special property
- In [Lindell], r_1 is a commitment to a verification key
- Note compound language in NP if $L \in NP$
  - Witness to (x, r_1, vk) is either witness to x or random tape for commitment ($r_1 = C(vk;w)$)

OTSS - Construction

- **Common reference string**: (r_1, r_2)
- **Prover(x,w)**:
  - Choose random pair of signature keys (vk,sk)
  - Prove compound statement $(x,r_1,vk)\in L'$ using w and r_2
  - Sign on proof $\sigma = Sign_{sk}(x,p)$
  - Output (vk,x,p,σ)
- **Verifier(vk,x,p,σ)**:
  - Verify signature $Ver_{sk}((x,p),\sigma)=1$
  - Verify proof $V((x,r_1,vk),r_2,p)=1$

OTSS - Proof

- Completeness immediate
- Soundness:
  - Random string is a valid commitment with negligible probability
  - For a random r_1, $x \notin L$ implies $(x,vk,r_1) \notin L'$
- Adaptive soundness:
  - Proofs generated using random r_2
  - Immediate from adaptive soundness of underlying NIZK

OTSS - Proof

- Zero knowledge:
  - Proves $(x,vk,r_1) \in L'$ based on r_1 = Commit(vk)
  - Reference string is pseudorandom because commitment has pseudorandom range
  - Underlying proof is indistinguishable due to WI of adaptive NIZK
  - Formally, define two hybrids

OTSS - Proof

- Zero knowledge:
  - Simulator (ref string):
    - Choose random (vk,sk)
    - Compute r_1 = Commit(vk)
    - Choose random r_2
    - Output (r_1,r_2)
  - Simulator (proof):
    - Prove statement based on r_1 = Commit(vk)
    - Sign input, proof with sk
    - Output proof (vk,x,p,σ)

OTSS - Proof

- One time simulation soundness:
  - Sim ref string (r_1,r_2), sim proof (vk,x,p,σ)
    Adversary outputs verifiable (vk',x',p',σ'), $x' \notin L$
  - $vk \neq vk'$:
    - By perfect binding, $r_1 \notin Commit(vk')$
    - $x' \notin L \Rightarrow (x',r_1,vk') \notin L'$
    - Negligible by soundness of underlying NIZK (r_2 uniform)
  - $vk = vk'$:
    - $((x,p),\sigma) \neq ((x',p'),\sigma')$
    - Negligible by the strong security of the signature (sk unused)

Time's up…