RSA CONFERENCE 2014
FEBRUARY 24 – 28 | MOSCONE CENTER | SAN FRANCISCO

Broadcast Steganography or How to Broadcast a Secret *Covertly*

SESSION ID: CRYP-T08

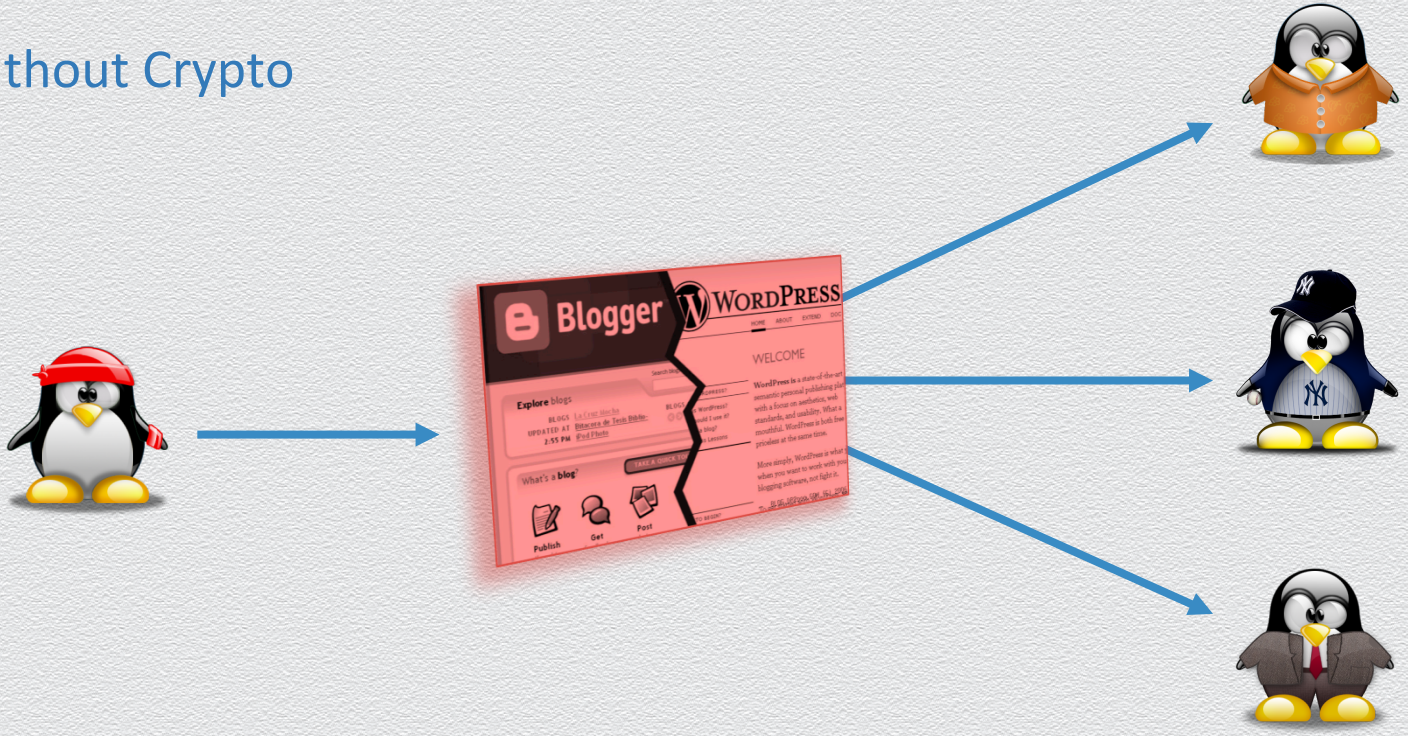
Nelly Fazio
The City College of CUNY
fazio@cs.ccny.cuny.edu

Antonio R. Nicolosi
Stevens Institute of Technology
nicolosi@cs.stevens.edu

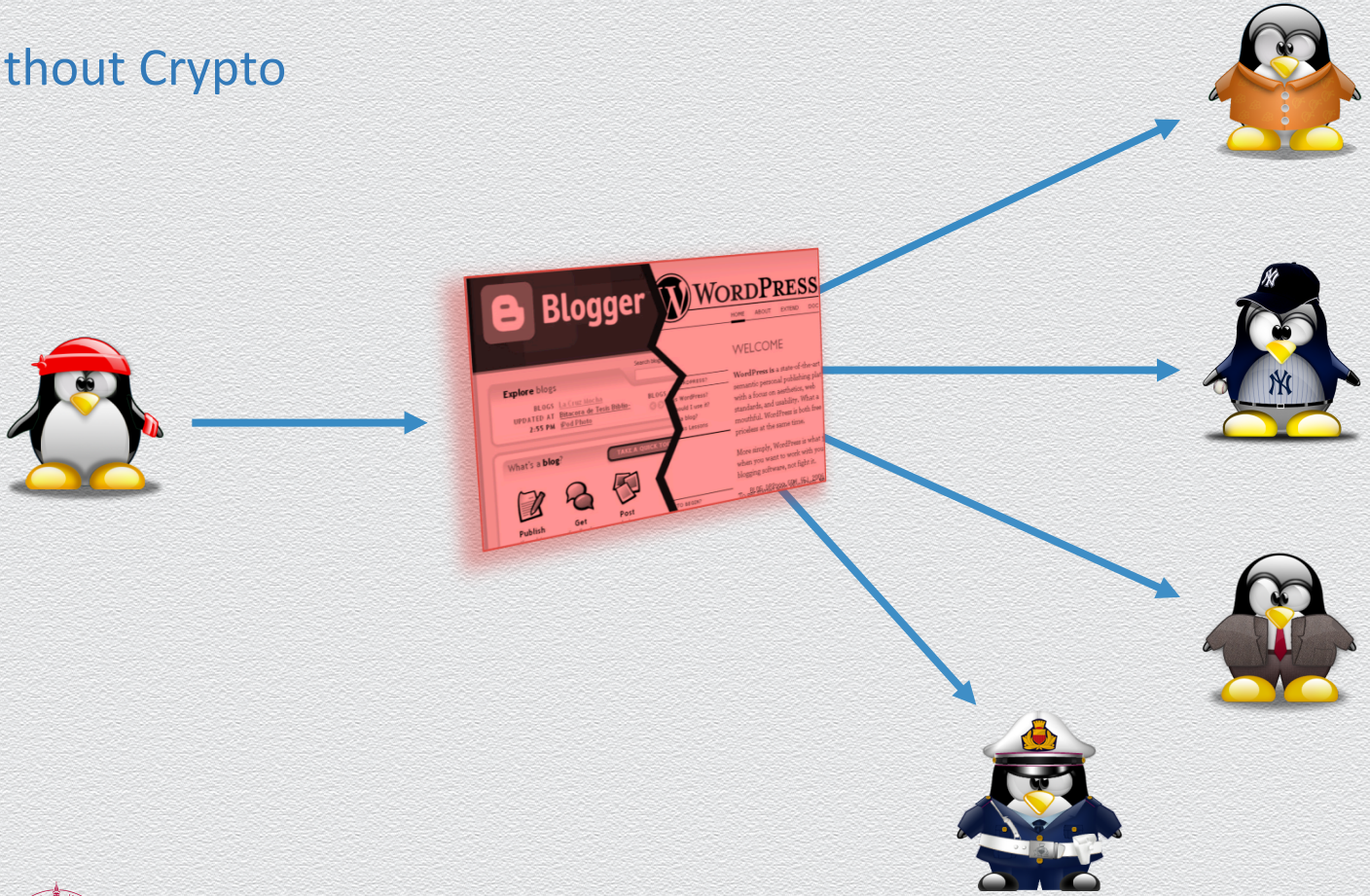
Irippuge Milinda Perera
The Graduate Center of CUNY
iperera@gc.cuny.edu

Without Crypto

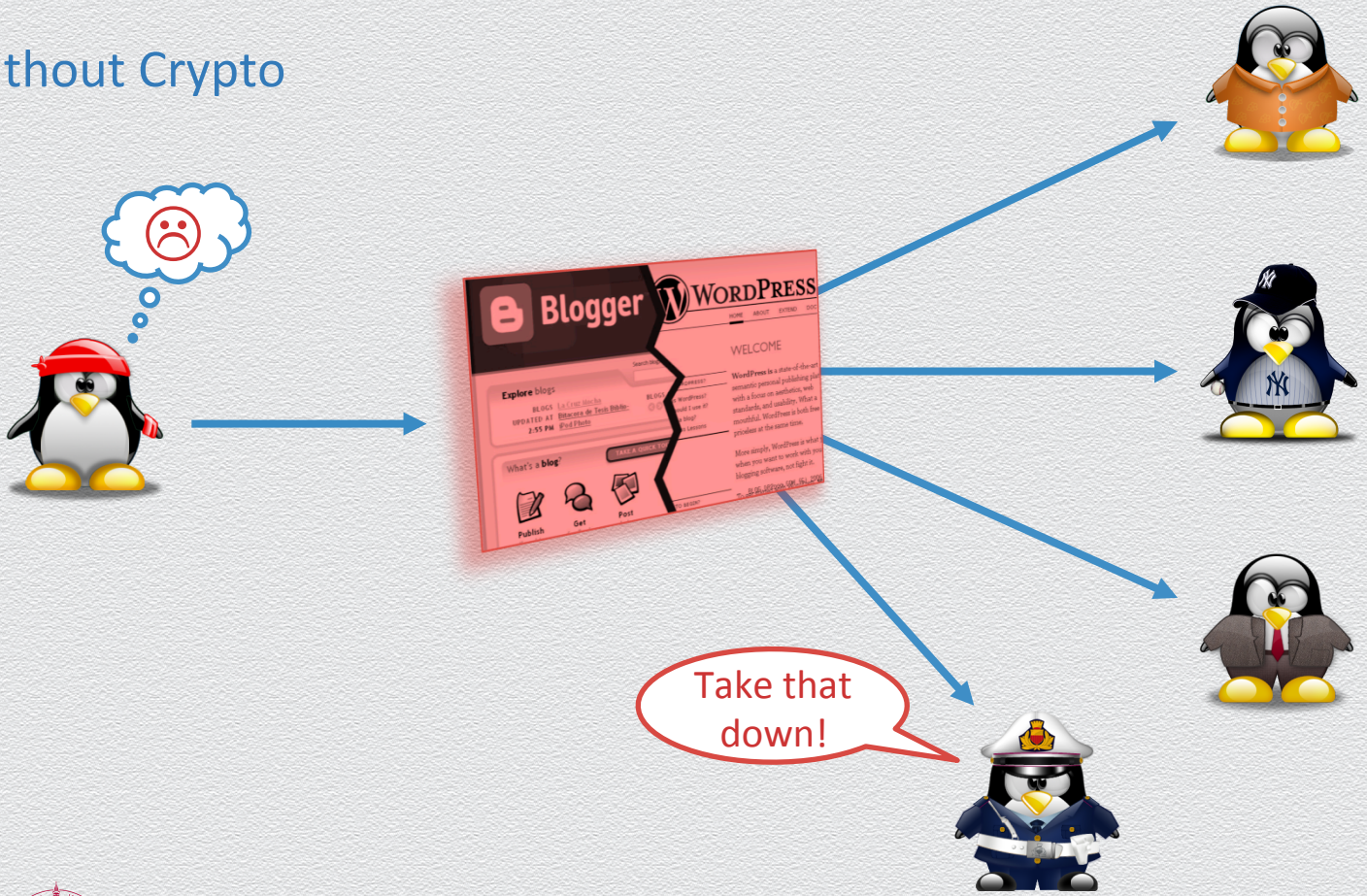
Without Crypto



Without Crypto

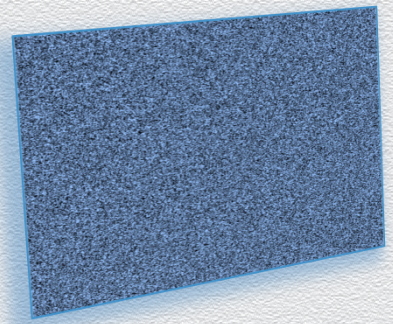


Without Crypto

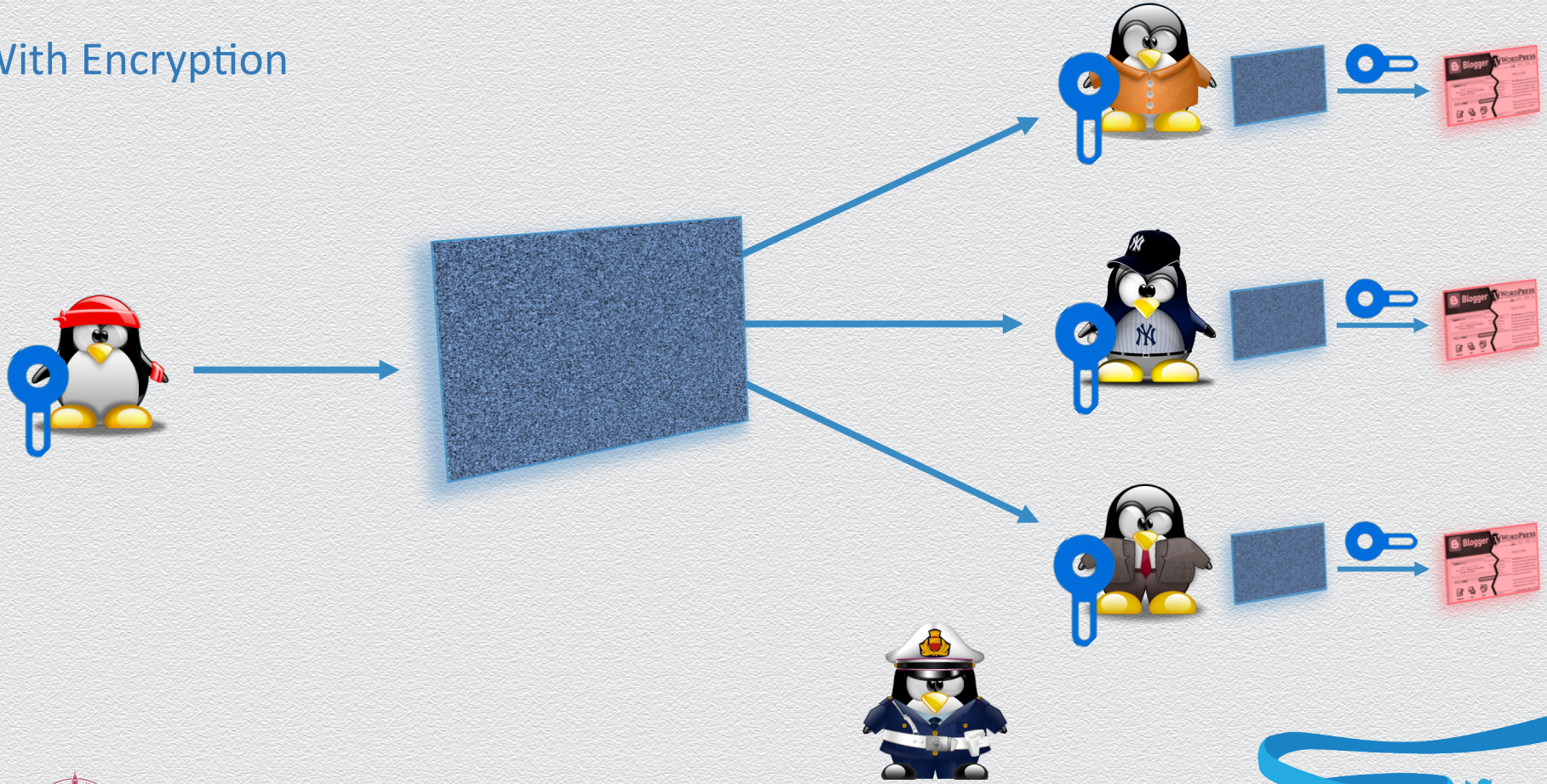


With Encryption

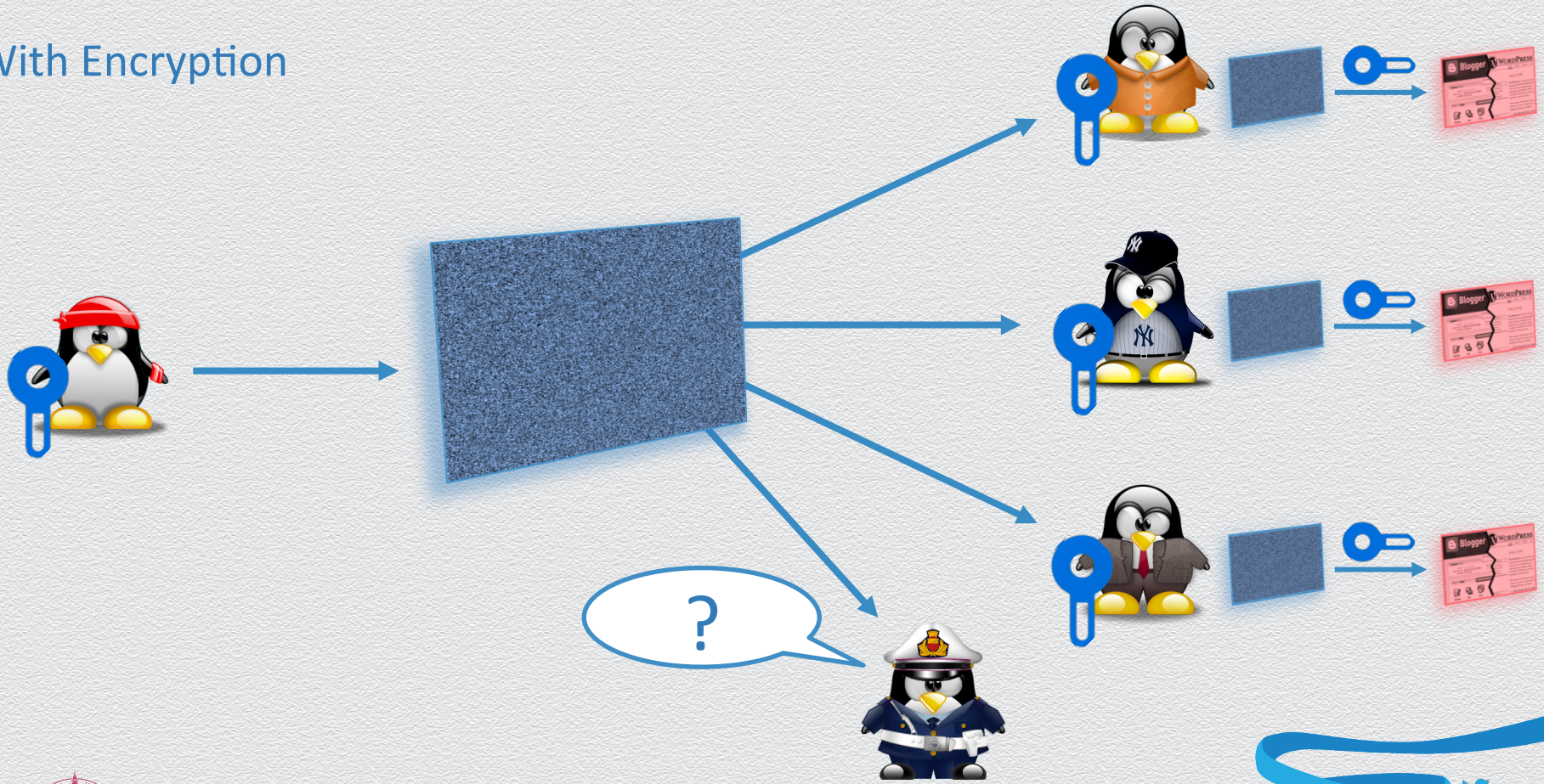
With Encryption



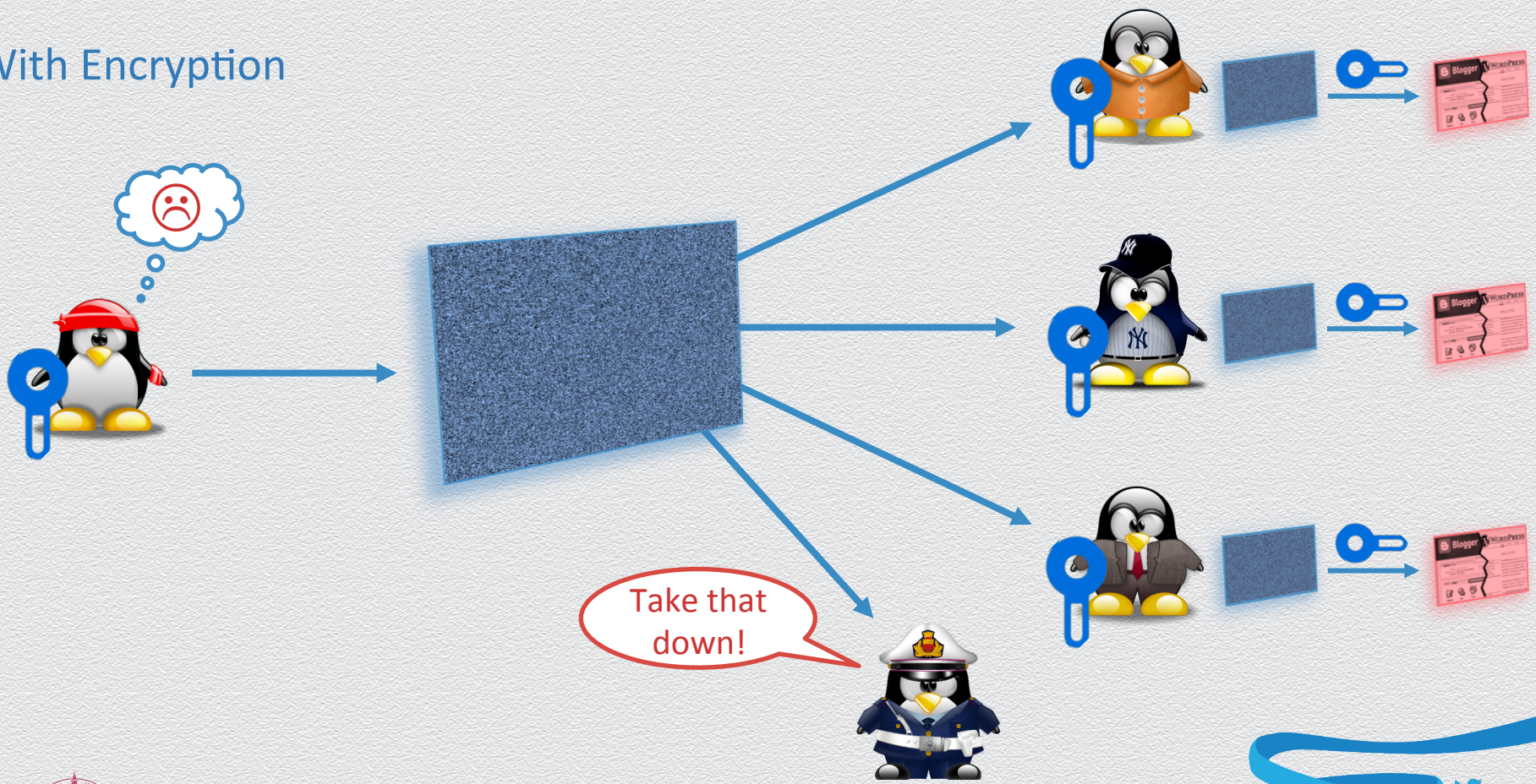
With Encryption



With Encryption



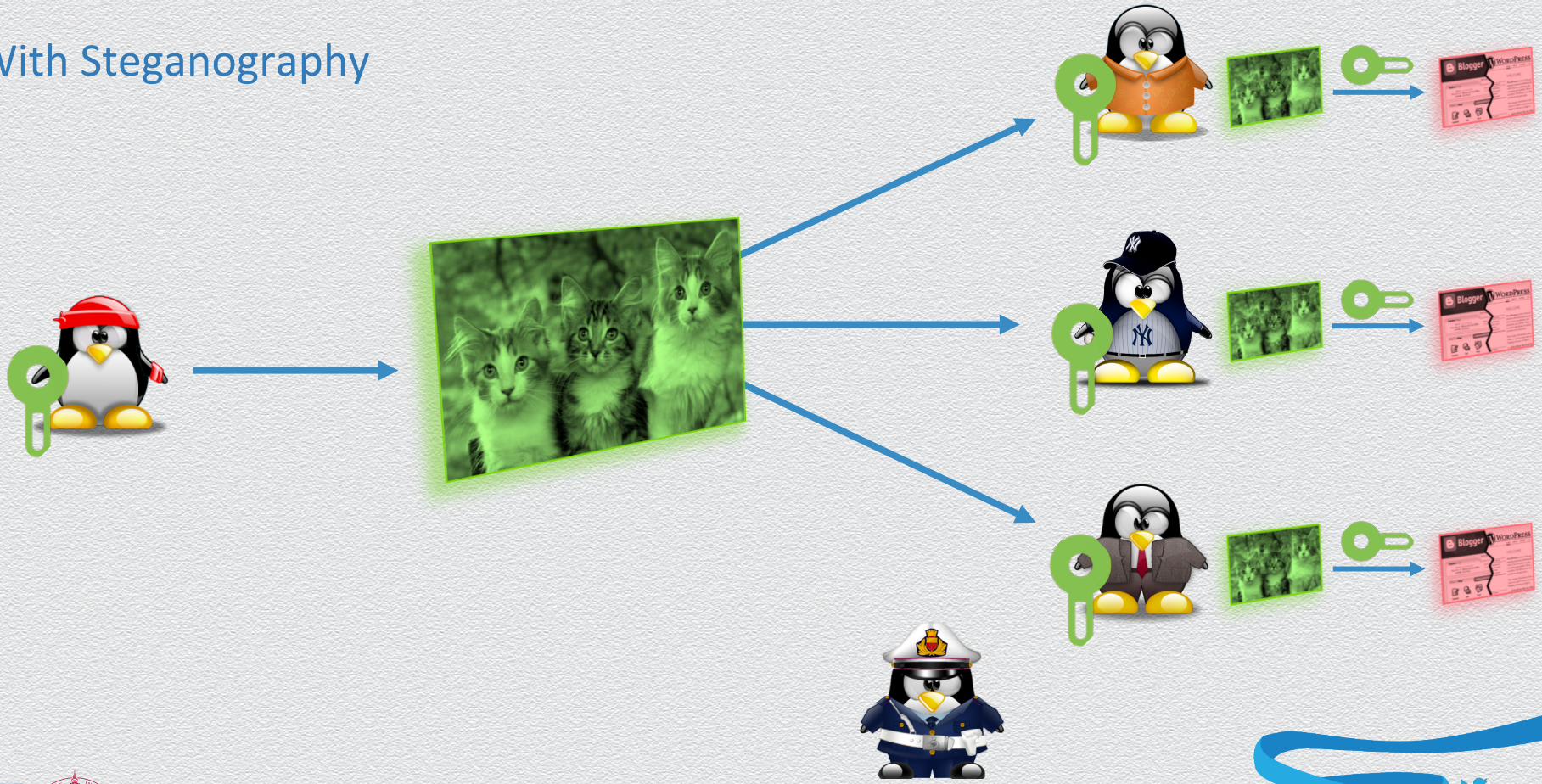
With Encryption



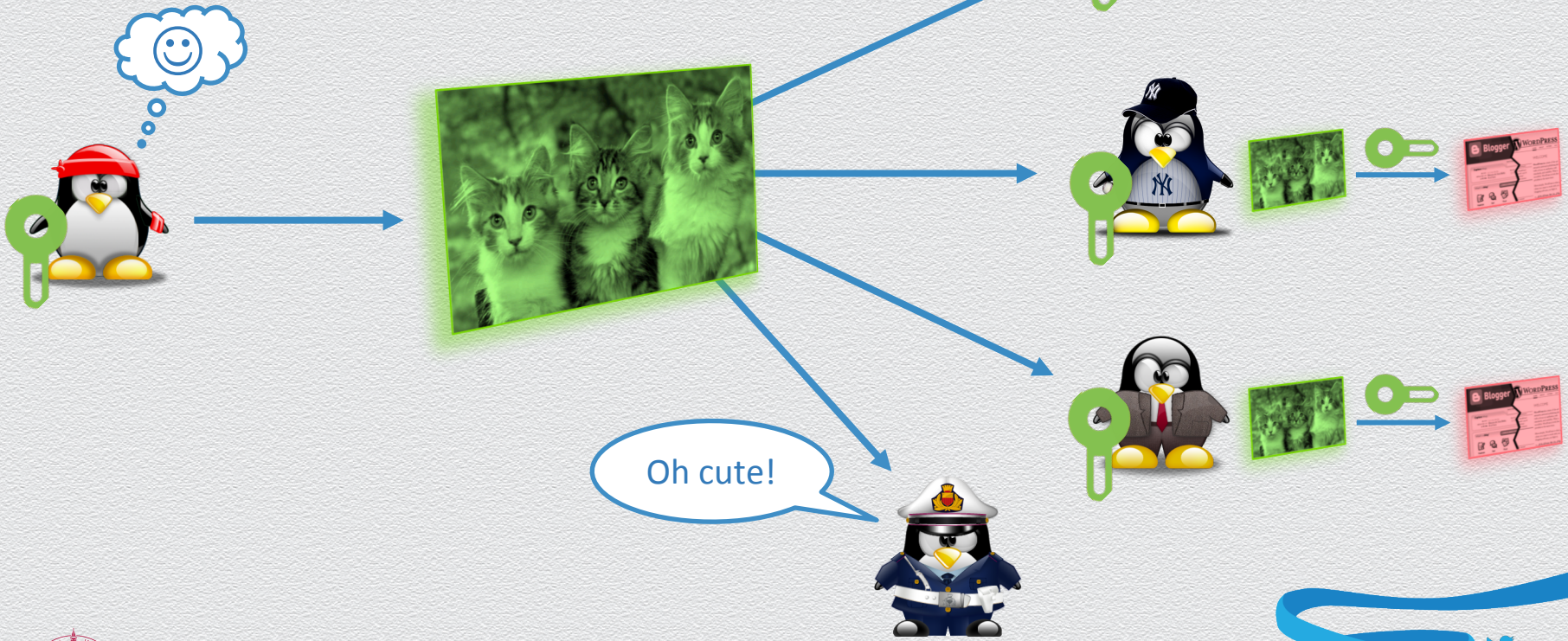
With Steganography

With Steganography

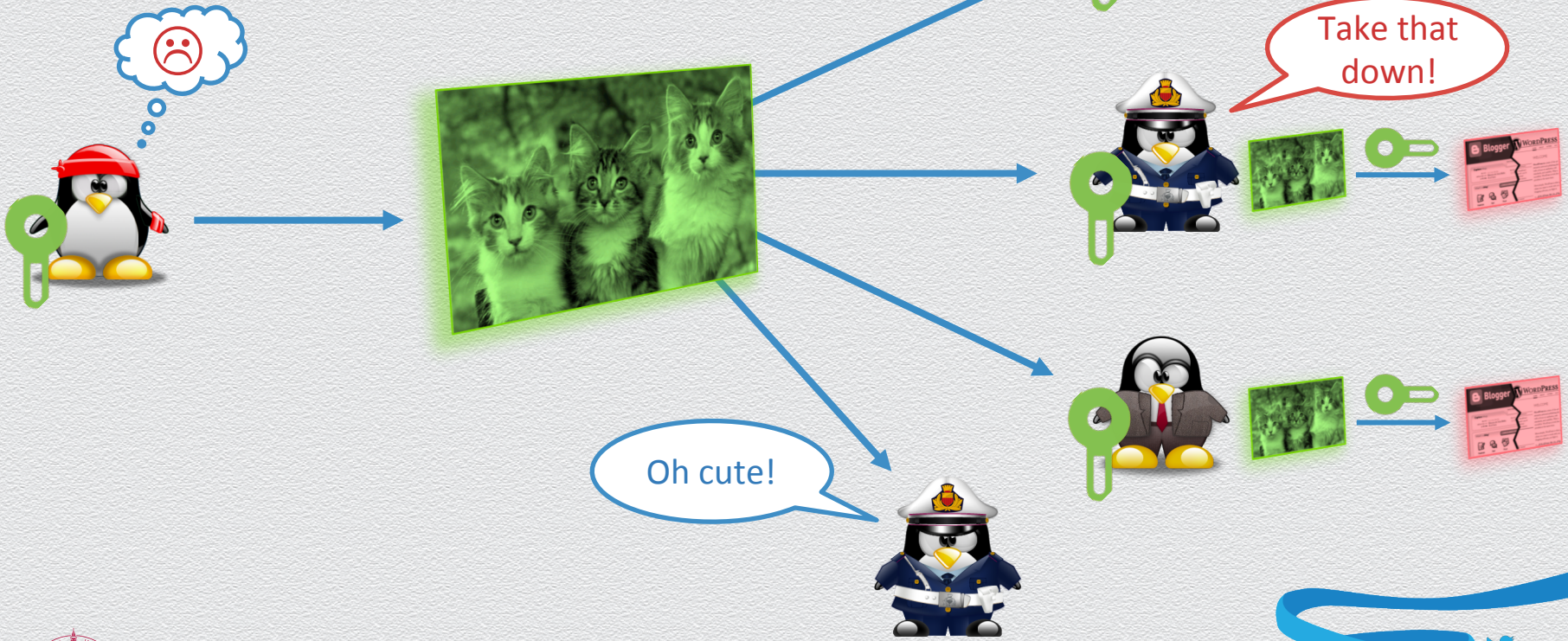
With Steganography



With Steganography

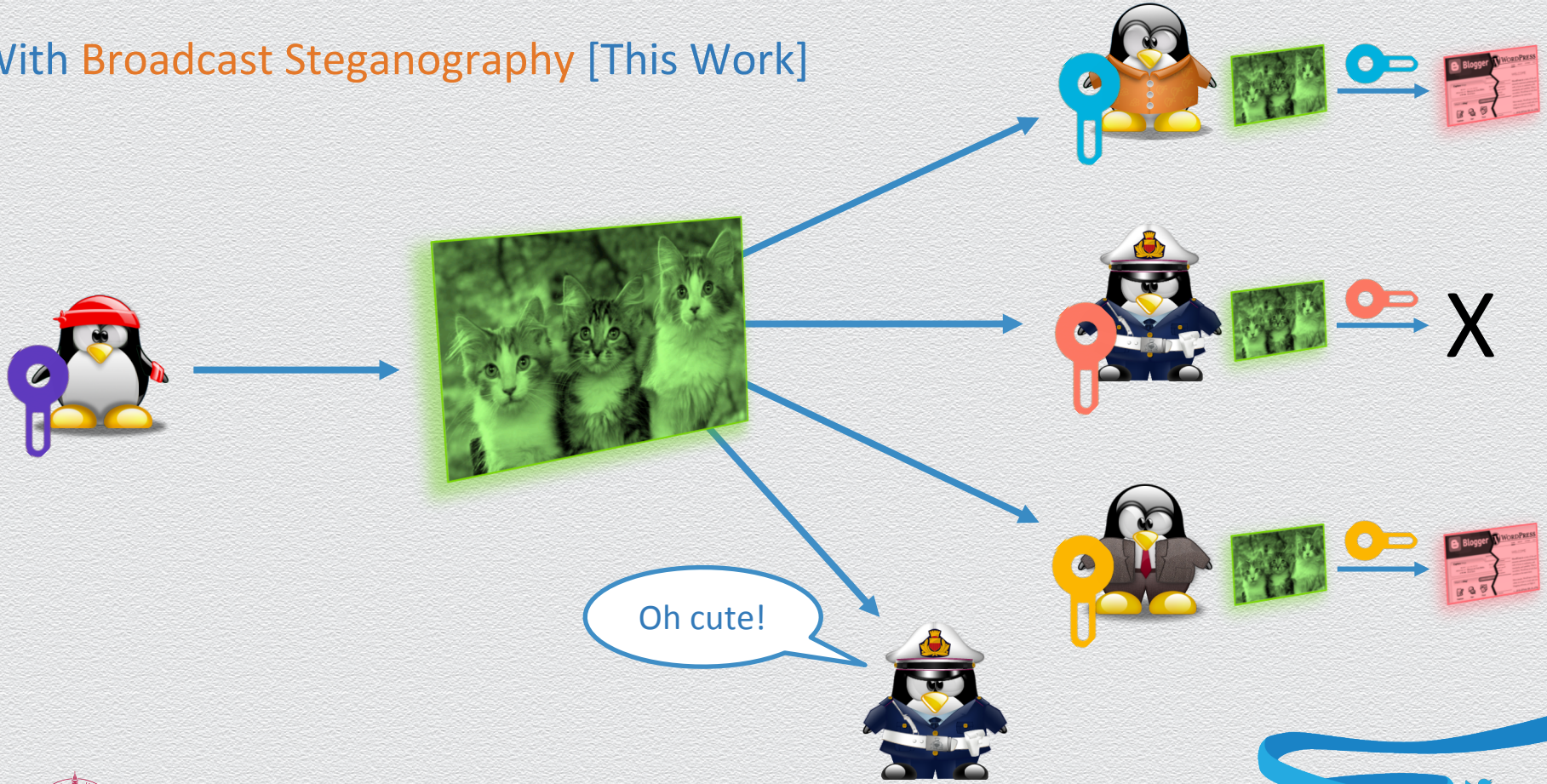


With Steganography

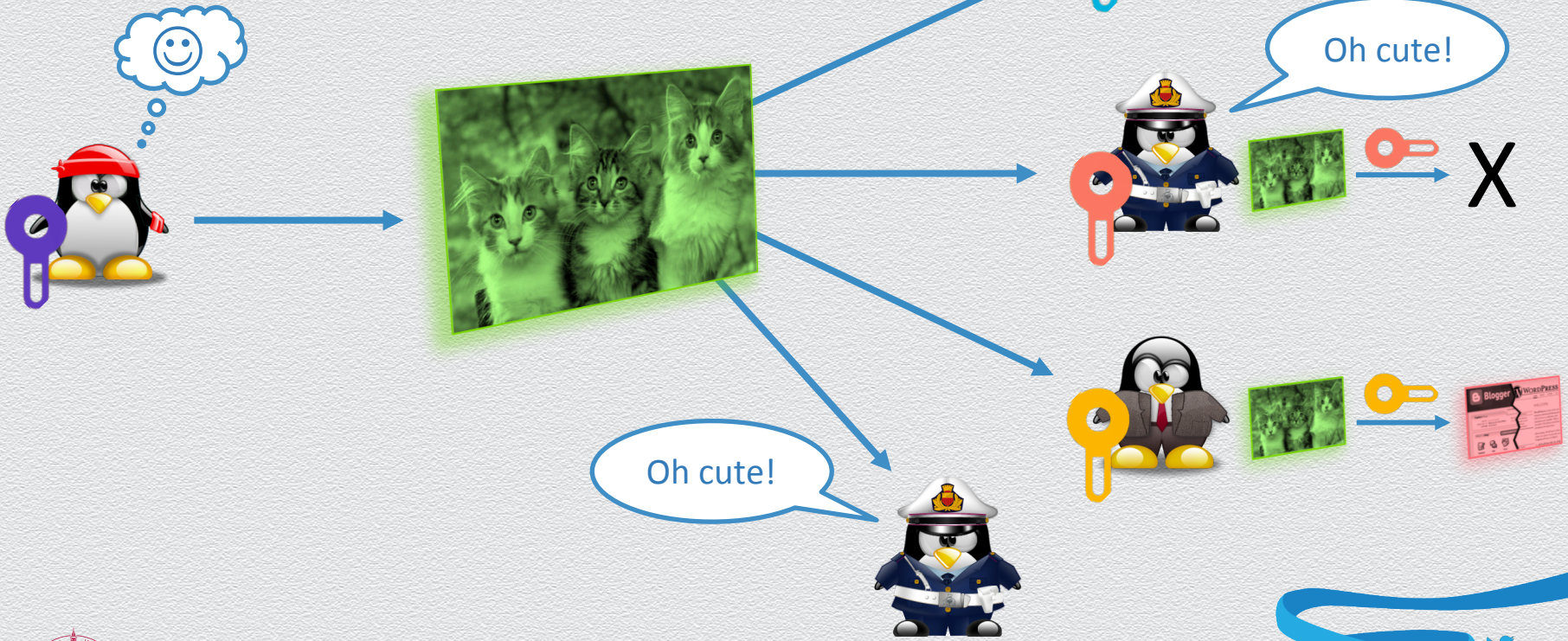


With Broadcast Steganography [This Work]

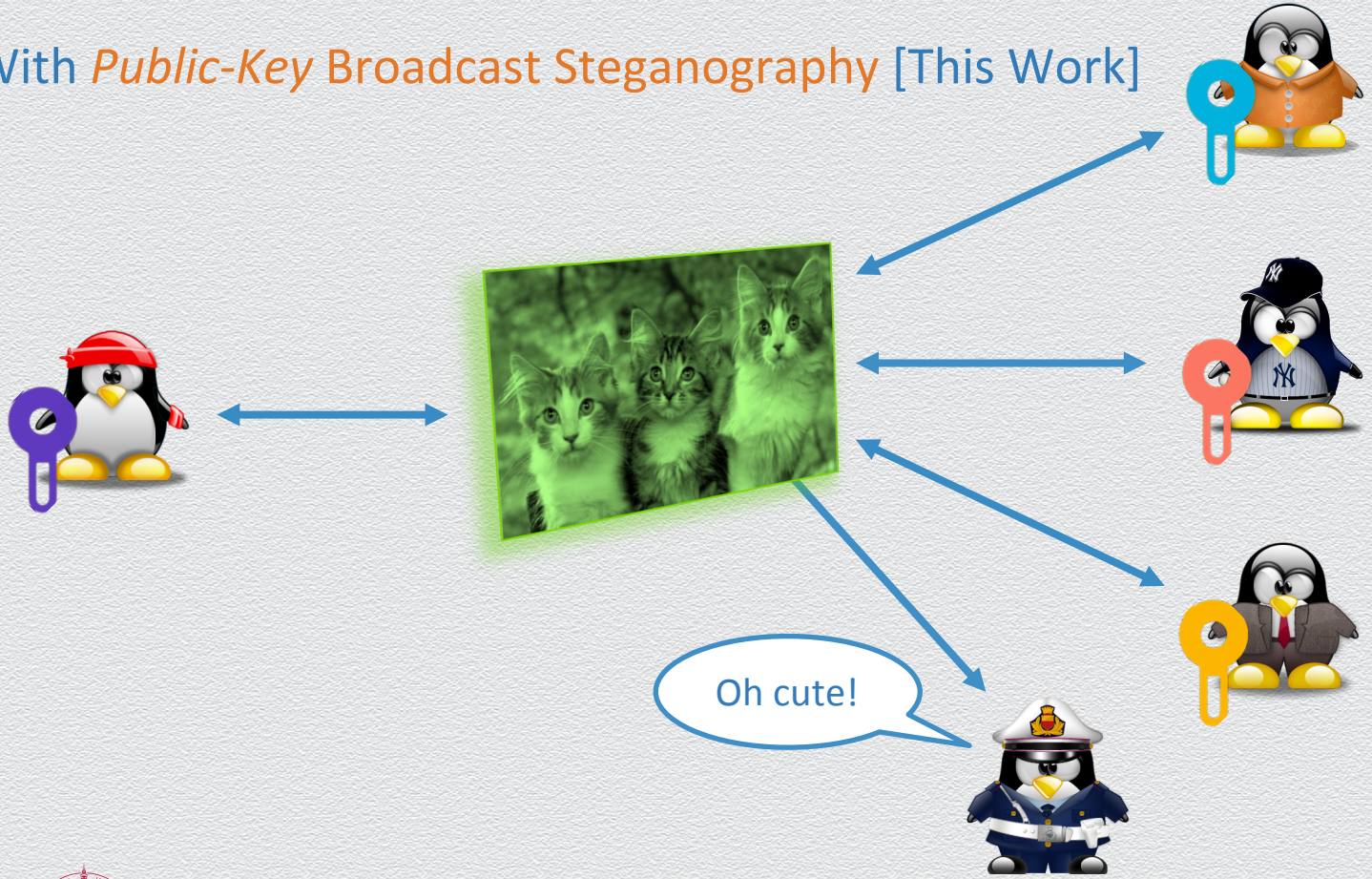
With Broadcast Steganography [This Work]

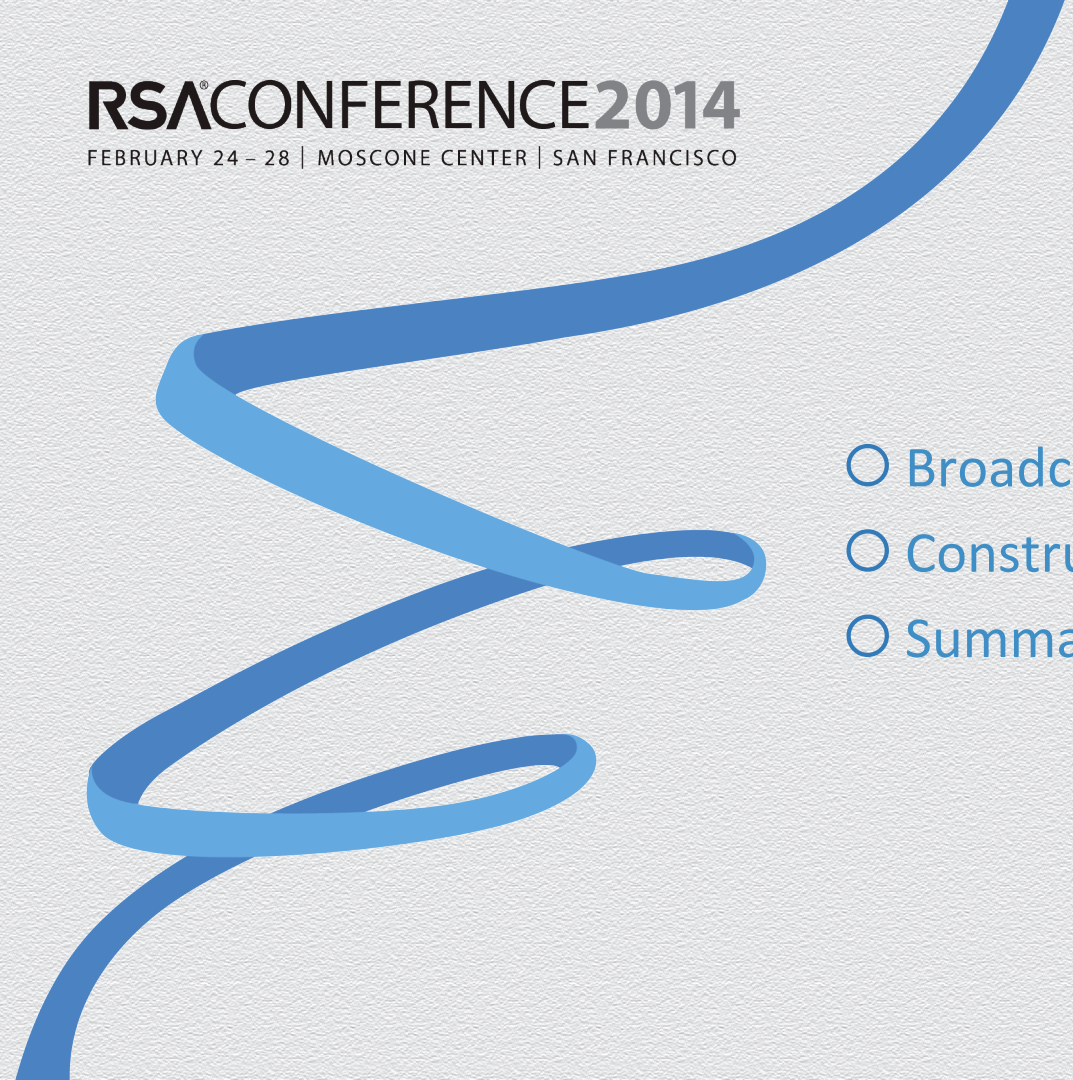


With Broadcast Steganography [This Work]

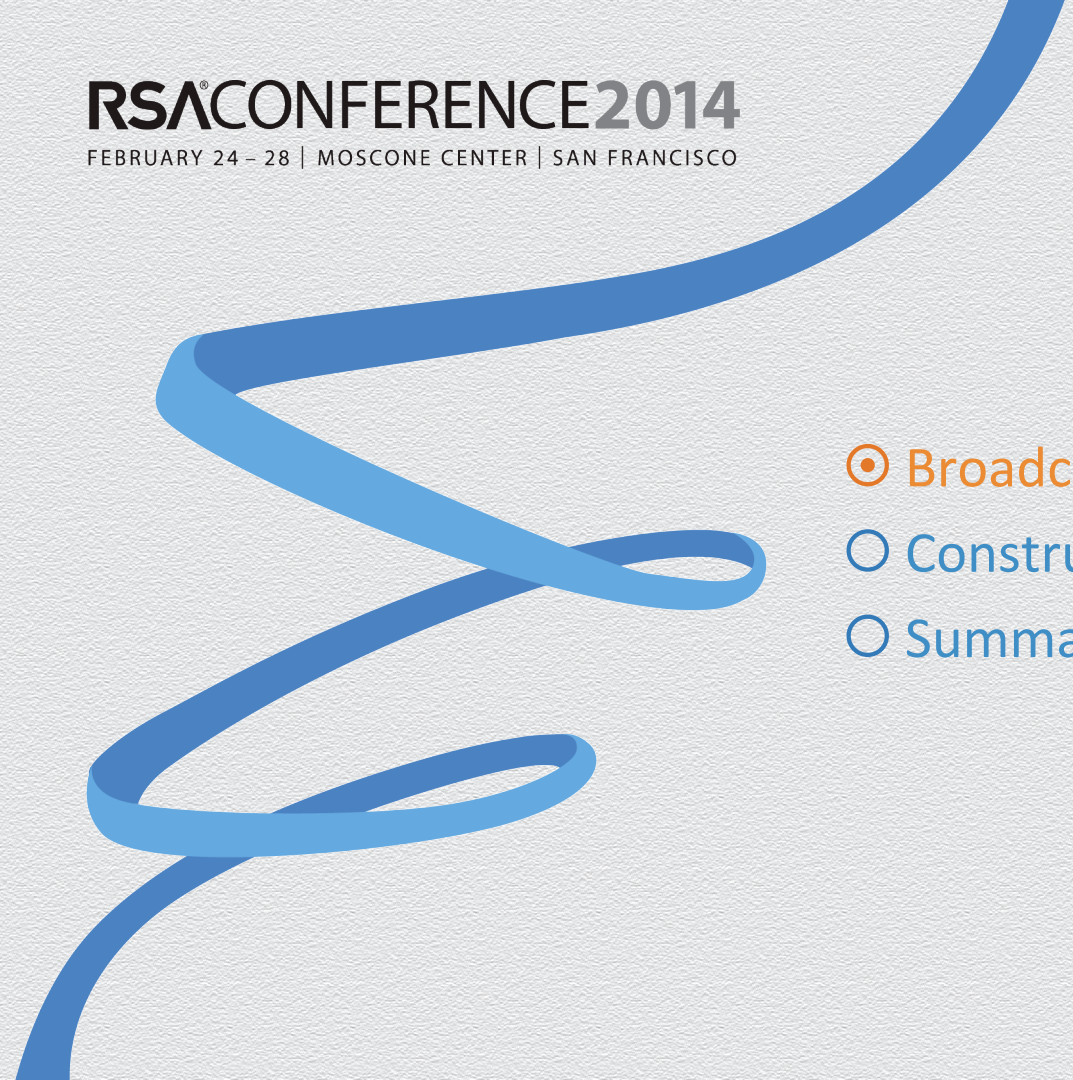


With *Public-Key* Broadcast Steganography [This Work]





- Broadcast Steganography (BS)
- Constructions
- Summary



- ⊙ Broadcast Steganography (BS)
- ○ Constructions
- ○ Summary

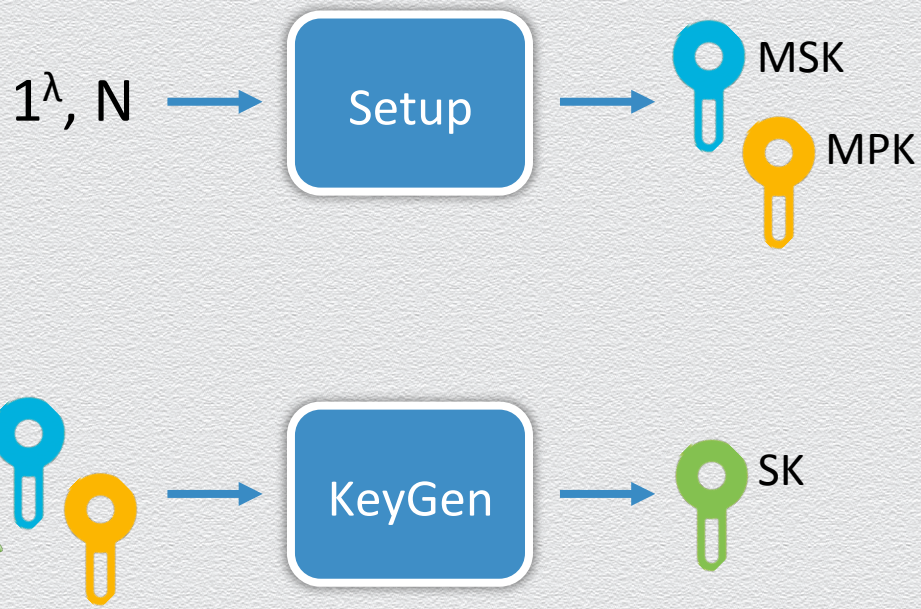
The Setting

The Setting

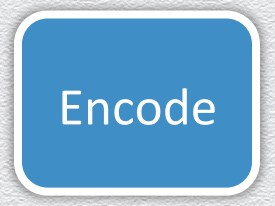
The Setting

KeyGen

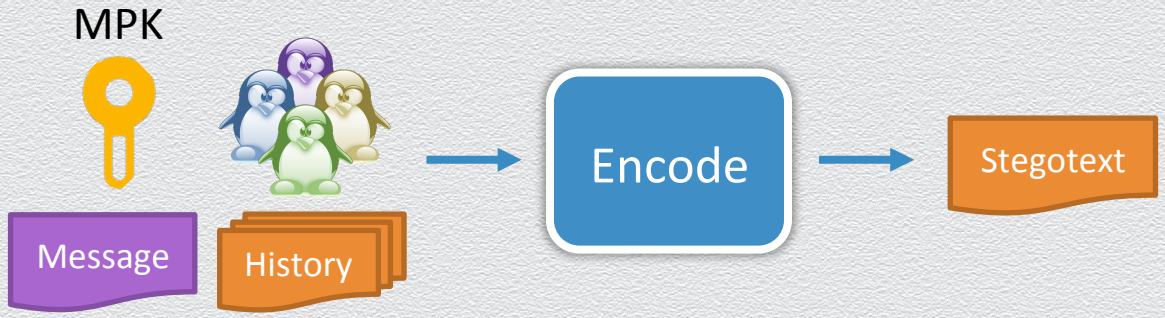
The Setting



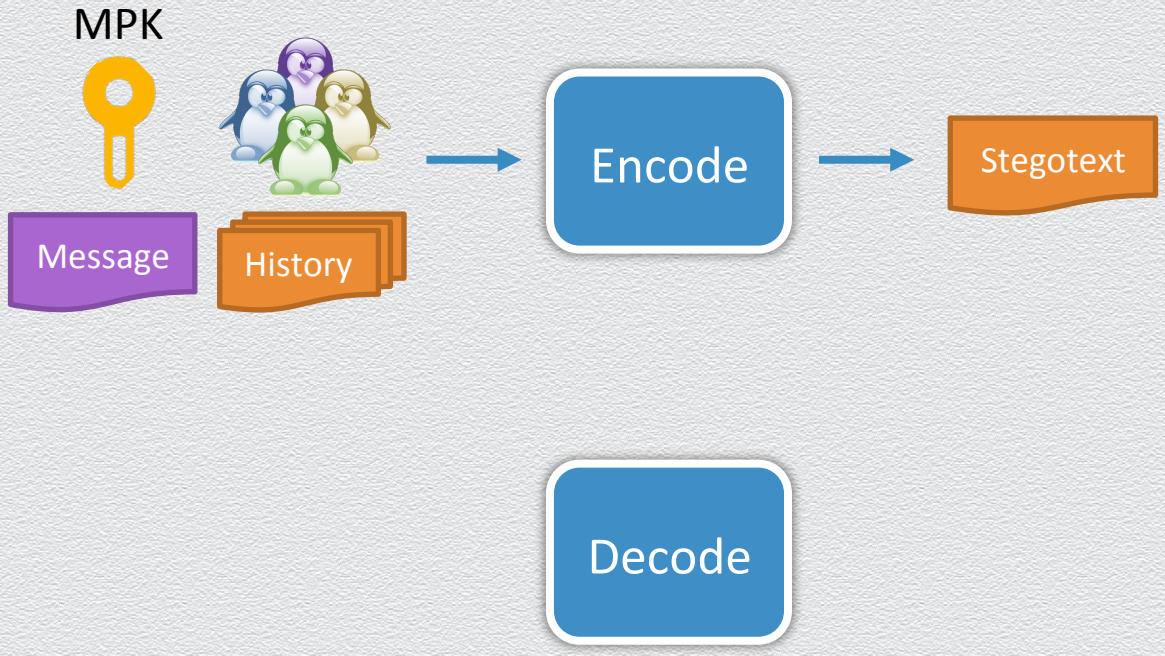
The Setting



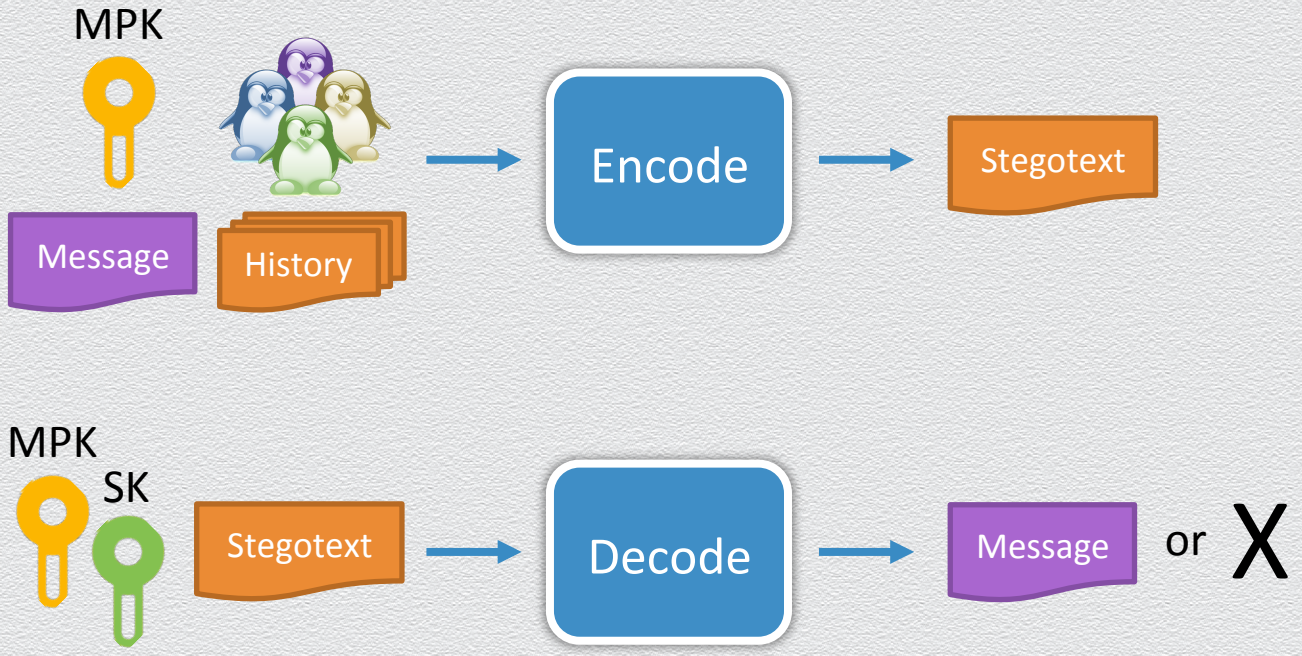
The Setting



The Setting

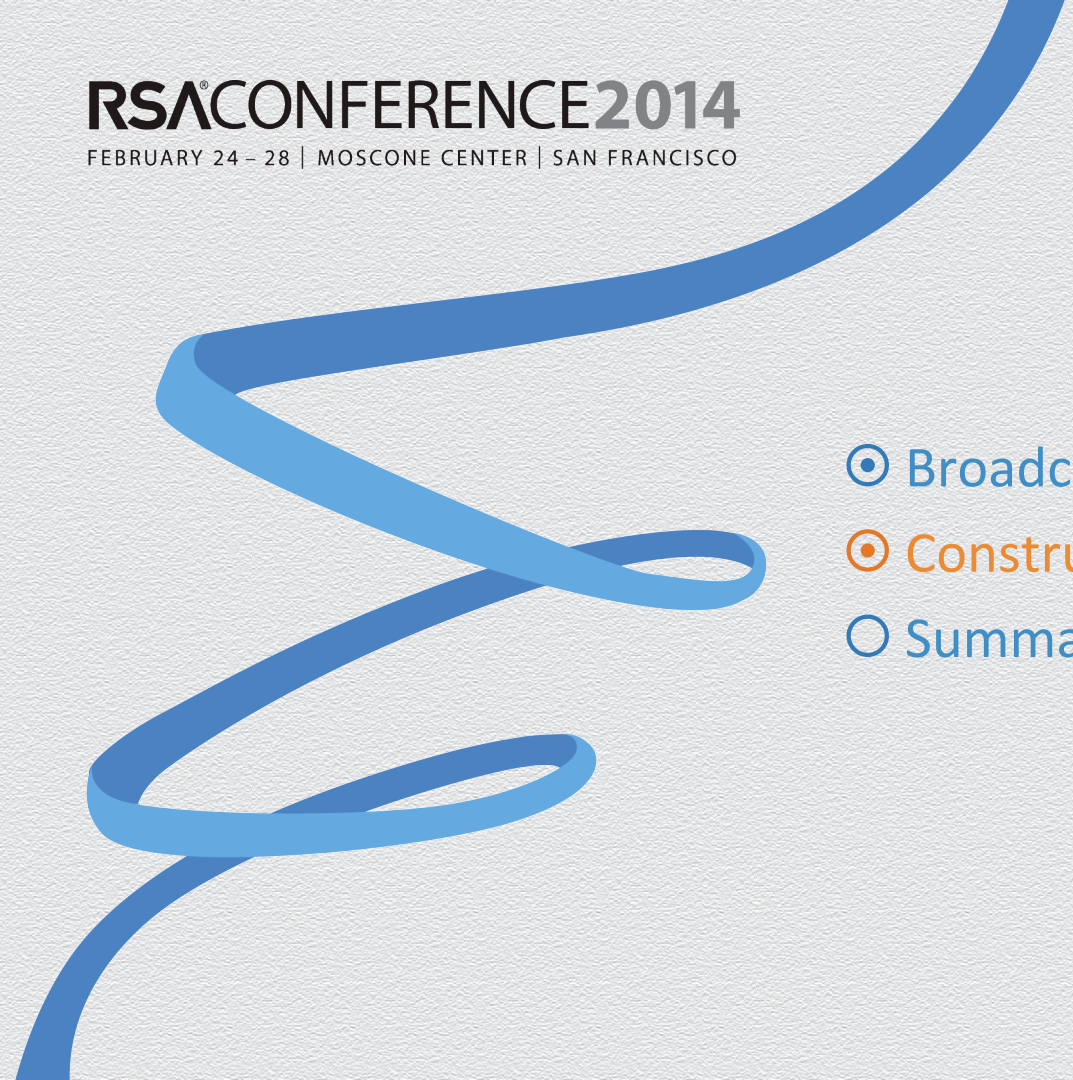


The Setting


MPK
Message
History
Encode
Stegotext
MPK
SK
Stegotext
Decode
Message
or
X

The Security Model

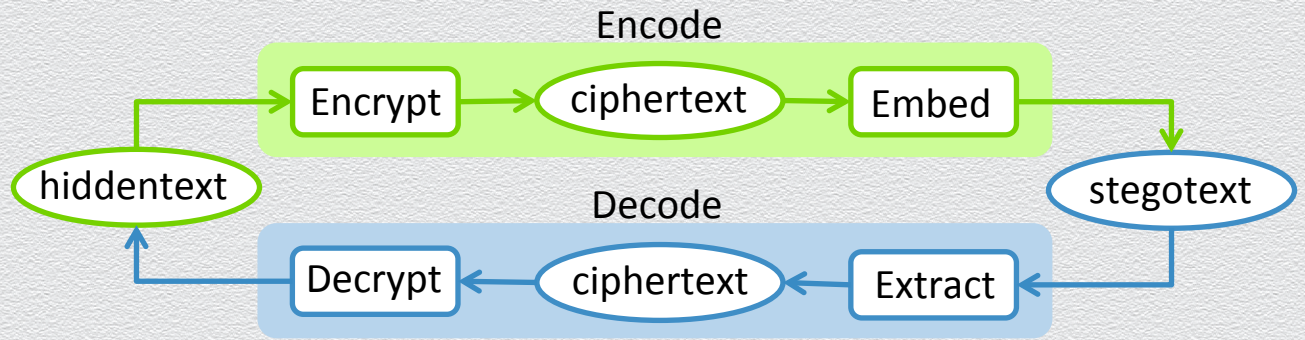
1. Chosen-Covertext Attack (BS-IND-CCA)
   - Analogous to BE-IND-CCA model
   - Adversary is allowed to corrupt users
   - Adversary is also given access to a decoding oracle
2. Publicly-Detectable Replayable Chosen Covertext Attack (BS-IND-PDR-CCA)
   - Similar to BS-IND-CCA, but with stricter restrictions on allowable decoding queries
3. Chosen-Hiddentext Attack (BS-IND-CHA)
   - Analogous to BE-IND-CPA model
   - Adversary is only allowed to corrupt users
   - No decoding queries



- ⊙ Broadcast Steganography (BS)
- ⊙ Constructions
- ○ Summary

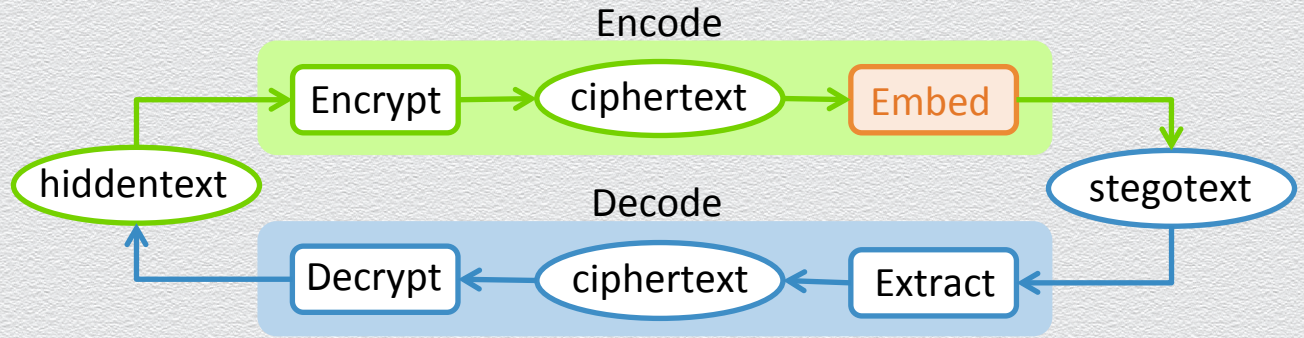
Realizing Broadcast Steganography

- Encrypt-then-Embed Paradigm [HLvA02, BaCa05]



Realizing Broadcast Steganography

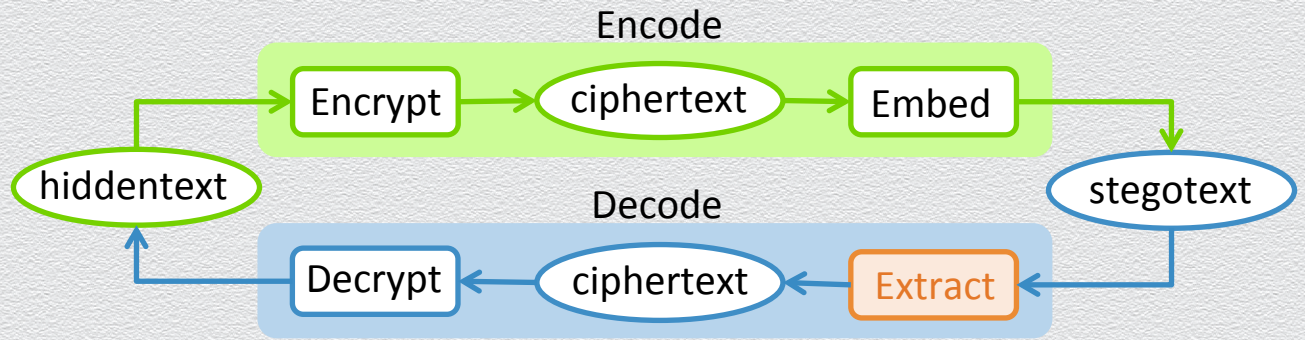
- Encrypt-then-Embed Paradigm [HLvA02, BaCa05]



- Embed (rejection-sampling)
  1. Let H be a strongly universal hash function
  2. Break the ciphertext c into bits $c_1, c_2, \ldots, c_l$
  3. To embed c_i, sample s_i from the channel until $H(s_i) = c_i$
  4. Output $s = s_1 || s_2 || \cdots || s_l$

Realizing Broadcast Steganography

- Encrypt-then-Embed Paradigm [HLvA02, BaCa05]



- Extract
  1. Break the stegotext s into documents $s_1, s_2, \ldots, s_l$
  2. Set $c_i = H(s_i)$
  3. Output $c = c_1 || c_2 || \cdots || c_l$

Broadcast Encryption + Encrypt-then-Embed = Broadcast Steganography?

- Encrypt-then-Embed requires pseudorandom ciphertexts …
- … but, Broadcast ciphertexts have structure

broadcast ciphertext format

- Neither header nor body is pseudorandom

Outsider-Anonymous Broadcast Encryption [FaPe12]

- Motivation: Anonymous Broadcast Encryption with short ciphertexts
  - A fully anonymous ciphertext length is subject to a linear lower bound [KiSa12]
  - In some applications, content may give recipient set away
    - ⇒ Suffices to protect anonymity of receivers from outsiders

- Outsider-Anonymity in Broadcast Encryption
  - Trades some degree of anonymity for better efficiency
  - Allows constructions with sub-linear ciphertext length

oABE Encryption in [FaPe12]

- Encrypt(S, m)
  1. Group users in S into S', a set of disjoint subsets
     - $|S'|$ is sub-linear in $|S|$
  2. Generate a ciphertext c_i for each s_i in S' (using anonymous IBE)
  3. Attach a tag t_i to each c_i (for efficient decryption at the receivers)
  4. Bundle all (t_i, c_i) components using one-time signature

oABE Encryption in [FaPe12]

- Encrypt(S, m)
  1. Group users in S into S', a set of disjoint subsets
     - |S'| is sub-linear in |S|
  2. Generate a ciphertext c_i for each s_i in S' (using anonymous IBE)
  3. Attach a tag t_i to each c_i (for efficient decryption at the receivers)
  4. Bundle all (t_i, c_i) components using one-time signature

oABE Encryption in [FaPe12]

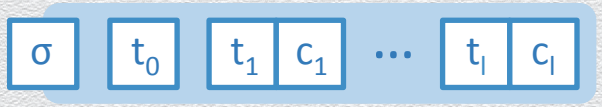
- Encrypt(S, m)
  1. Group users in S into S', a set of disjoint subsets
     - |S'| is sub-linear in |S|
  2. Generate a ciphertext c_i for each s_i in S' (using anonymous IBE)
  3. Attach a tag t_i to each c_i (for efficient decryption at the receivers)
  4. Bundle all (t_i, c_i) components using one-time signature

oABE Encryption in [FaPe12]

- Encrypt(S, m)
  1. Group users in S into S', a set of disjoint subsets
     - |S'| is sub-linear in |S|
  2. Generate a ciphertext c_i for each s_i in S' (using anonymous IBE)
  3. Attach a tag t_i to each c_i (for efficient decryption at the receivers)
  4. Bundle all (t_i, c_i) components using one-time signature

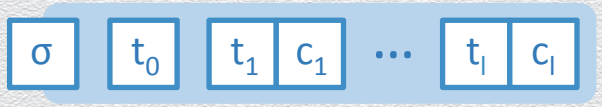
oABE Encryption in [FaPe12]

- Encrypt(S, m)
  1. Group users in S into S', a set of disjoint subsets
     - |S'| is sub-linear in |S|
  2. Generate a ciphertext c_i for each s_i in S' (using anonymous IBE)
  3. Attach a tag t_i to each c_i (for efficient decryption at the receivers)
  4. Bundle all (t_i, c_i) components using one-time signature



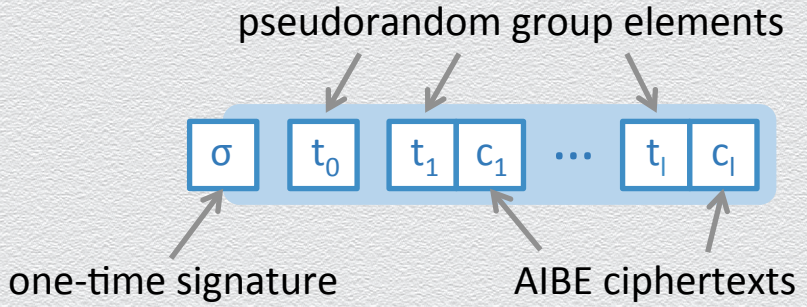
oABE Encryption in [FaPe12]

- Encrypt(S, m)
  1. Group users in S into S', a set of disjoint subsets
     - $|S'|$ is sub-linear in $|S|$
  2. Generate a ciphertext c_i for each s_i in S' (using anonymous IBE)
  3. Attach a tag t_i to each c_i (for efficient decryption at the receivers)
  4. Bundle all (t_i, c_i) components using one-time signature



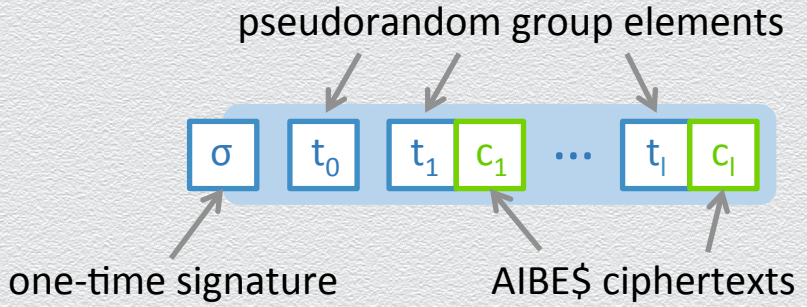
- Notice that ciphertexts have no header …
- … but still exhibit structure due to tags and signature
- Idea: Toward a BS construction, make these components pseudorandom

oABE with Pseudorandom Ciphertexts (oABE$) [This Work]



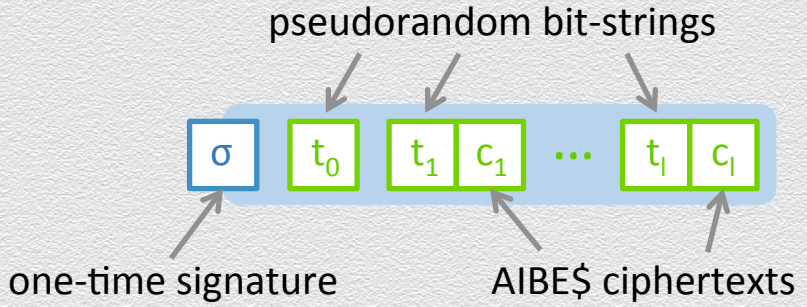
- How to make oABE ciphertexts pseudorandom?
  1. Replace the underlying AIBE with AIBE$ [AgBo09]
  2. Apply an entropy smoothing hash to group elements
  3. Replace one-time signature with a MAC (implemented via PRF)

oABE with Pseudorandom Ciphertexts (oABE$) [This Work]



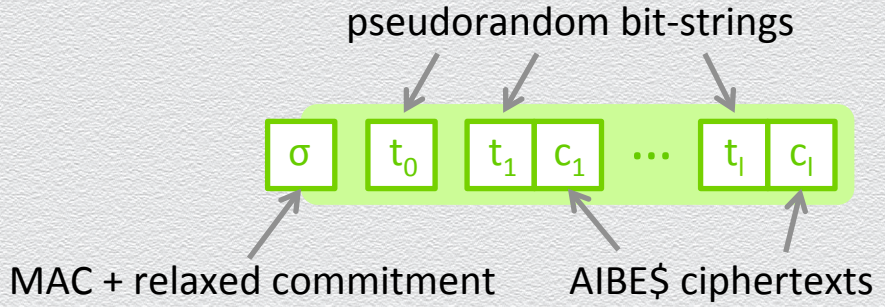
- How to make oABE ciphertexts pseudorandom?
  1. Replace the underlying AIBE with AIBE$ [AgBo09]
  2. Apply an entropy smoothing hash to group elements
  3. Replace one-time signature with a MAC (implemented via PRF)

oABE with Pseudorandom Ciphertexts (oABE$) [This Work]



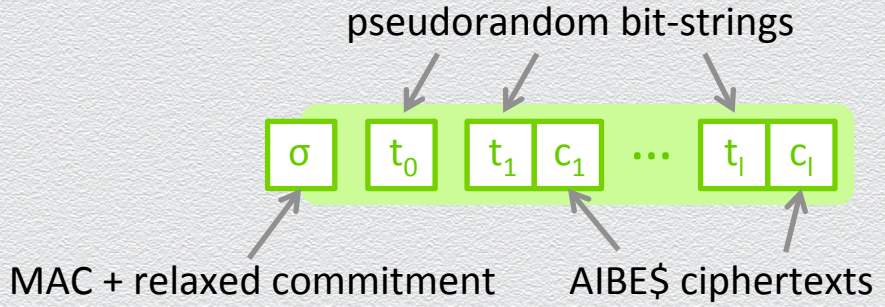
- How to make oABE ciphertexts pseudorandom?
  1. Replace the underlying AIBE with AIBE$ [AgBo09]
  2. Apply an entropy smoothing hash to group elements
  3. Replace one-time signature with a MAC (implemented via PRF)

oABE with Pseudorandom Ciphertexts (oABE$) [This Work]



- How to make oABE ciphertexts pseudorandom?
  1. Replace the underlying AIBE with AIBE$ [AgBo09]
  2. Apply an entropy smoothing hash to group elements
  3. Replace one-time signature with a MAC (implemented via PRF)

oABE with Pseudorandom Ciphertexts (oABE$) [This Work]



- How to make oABE ciphertexts pseudorandom?
  1. Replace the underlying AIBE with AIBE$ [AgBo09]
  2. Apply an entropy smoothing hash to group elements
  3. Replace one-time signature with a MAC (implemented via PRF)

Question: How to embed the MAC key in c_i's and still obtain CCA security?

Solution: Construct an encapsulation mechanism [DoKa05, BoKa05]
with pseudorandom commitments

Comparison of BE Schemes with Anonymity Properties

Scheme	|PK|	|sk|	|c|	Security Model	Anonymity
BBW06	O(N)	O(1)	O(N-r)	Static, RO	Full
LPQ12	O(N)	O(1)	O(N-r)	Adaptive, Standard	Full
FaPe12a	O(N)	O(log N)	O(r log (N/r))	Adaptive, Standard	Outsider
FaPe12b	O(N log N)	O(N)	O(r)	Adaptive, Standard	Outsider
This Work	O(N)	O(log N)	O(r log (N/r))	Adaptive, Standard	Outsider

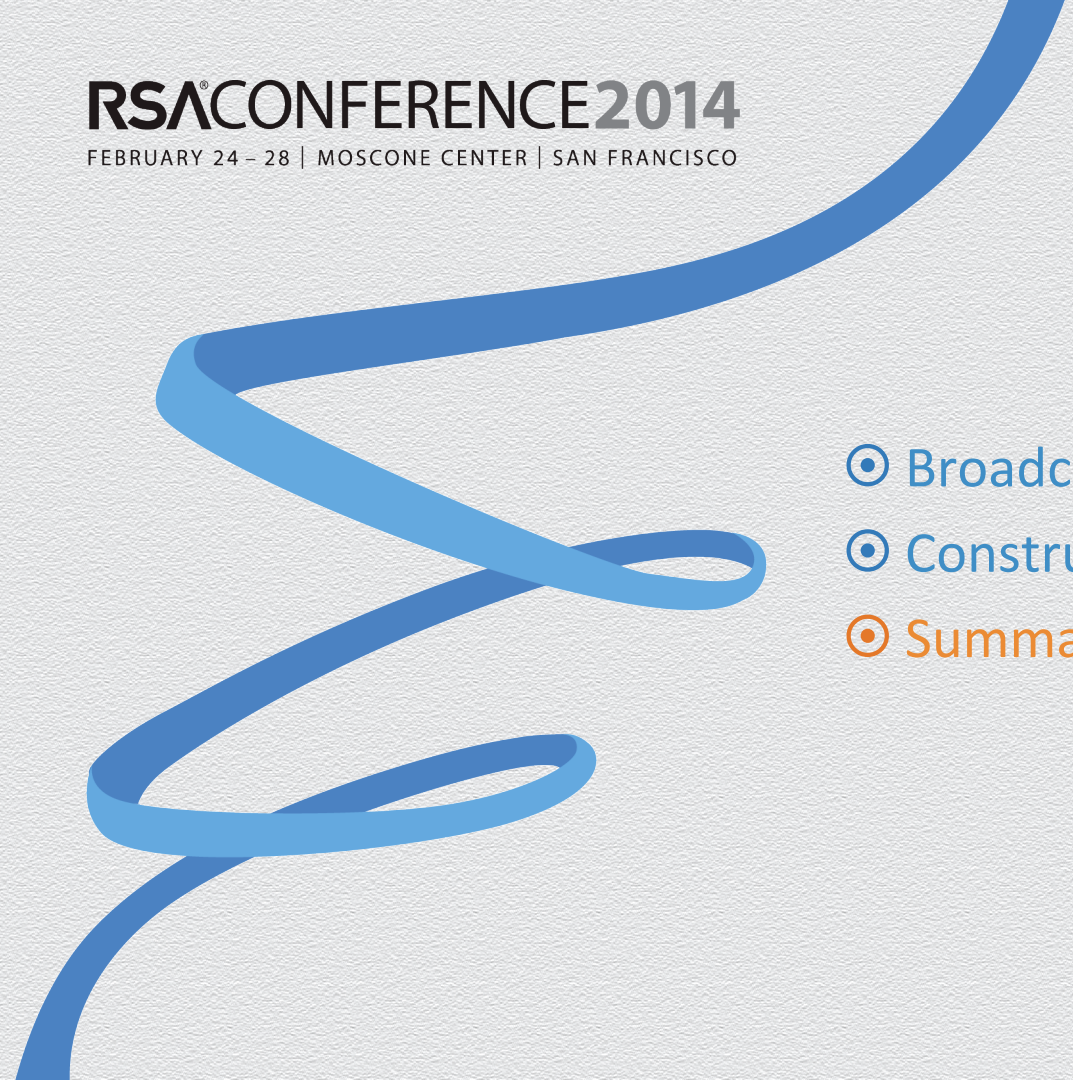
N: total number of users, r: number of revoked users

- Only oABE$ provides pseudorandom ciphertexts

Our Construction of Broadcast Steganography

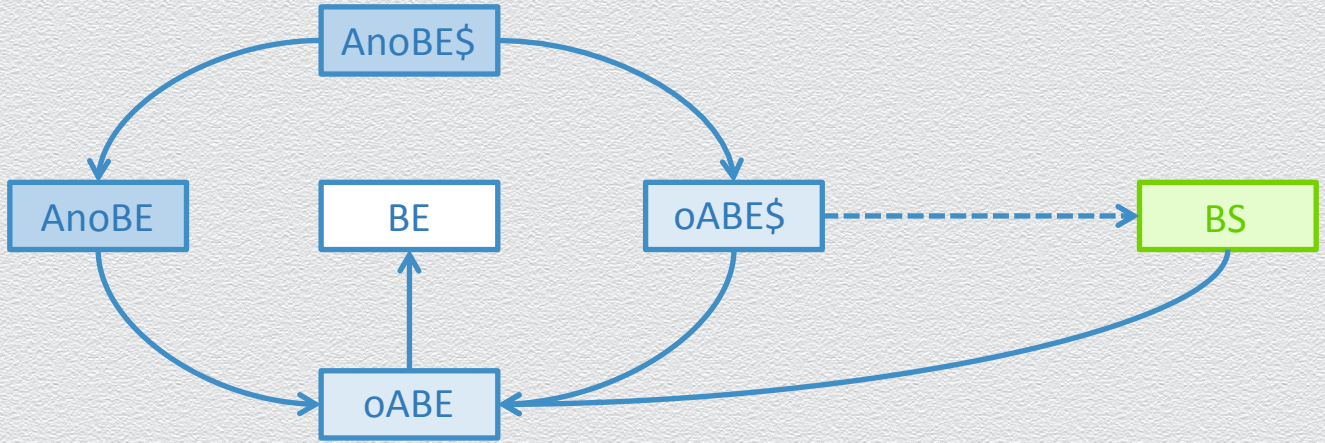
- Highlights
  - oABE$ + Encrypt-then-Embed = Broadcast Steganography
  - Our constructions have sub-linear stegotext length
  - For CCA security, requires stateless channel

- Constructions:
  1. BS-CHA
  2. BS-PDR-CCA
  3. BS-CCA



- Broadcast Steganography (BS)
- Constructions
- Summary

BE and Friends



Summary

- Initiated the study of Broadcast Steganography
  - A multi-recipient communication tool to plant undetectable messages in innocent-looking conversations
- Put forth sublinear constructions of broadcast steganography under a range of security notions
- In the process, devised efficient broadcast encryption schemes with pseudorandom ciphertexts and anonymity properties
  - Implementing CCA checks without imposing structure on broadcast ciphertexts required overcoming multiple technical hurdles