Deniable Liasons

Abhinav Narain,
Prof. Nick Feamster,
Prof. Alex Snoeren (UCSD)

College of Computing
School of Computer Science
Georgia Institute of Technology

Monday 28th April, 2014

Communication is Observable

- There is an increasing need for private communication in public places
- Unfortunately, most communication is observable, even if it is confidential
  - Phone calls
  - Internet traffic
  - Face-to-face meetings

Deniable communication

Motivating Examples

- Covert message passing between activists for public protest
- Whistleblower in an office environment
- Message exchange by a spy and a handler in coffee shop

Communication in proximity is still Observable

- Dead drops
- Normal conversations
- Physical media exchange

Existing techniques

- on Wide Area Networks
  - Generate benign, legitimate cover traffic
  - Transmit the actual traffic between it
  - Can provide Anonymity, but not Deniability
- on Wireless LANs
  - Timing Covert channels

Wireless is Noisy

- Non stationary process
- Has characteristic of being truly random
- Idea : Use inherent randomness of noisy wireless channel to hide secret messages

Packet Corruption in Wifi

- Ubiquitous, naturally occurring phenomenon
- Difficult to model
- Caused by
- Interference
  - Multipath
  - Non-wifi
- Collisions
- Hidden terminals
- Low signal strength

A New covert Channel

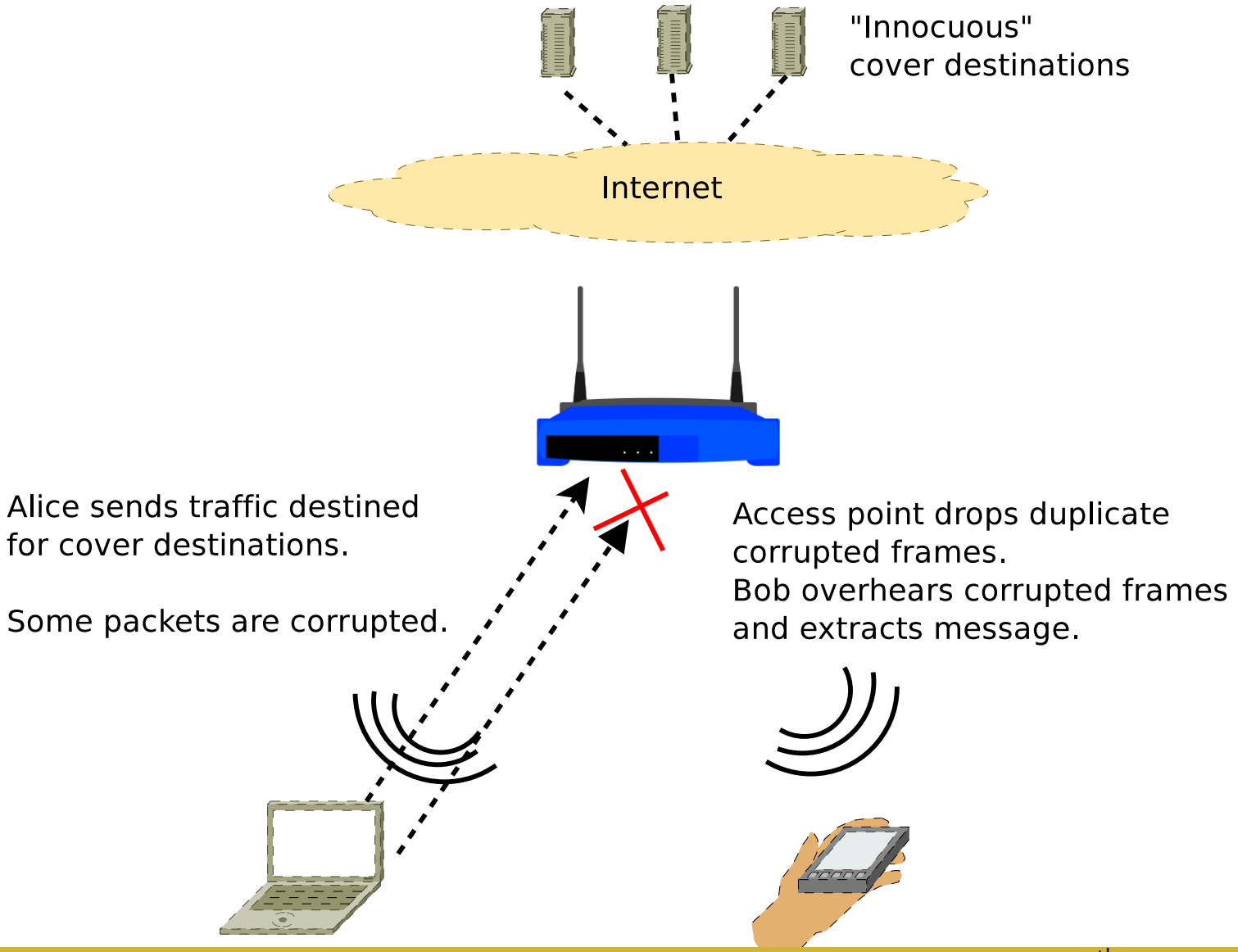
- Wireless broadcast medium
- Ubiquitous and natural phenomenon of packet corruption in Wifi
- Hide messages in corrupted frames
- Challenge : Make message indistinguishable from *natural* corruption

Chaffing and Winnowing

- *Chaff* is the actual corrupted frames on the channel due to packet corruption[1]
- *Grain* is the crafted SSL frames which are deliberately corrupted by the sender for the secret communication

[1] Rivest *et. al* Chaffing and Winnowing: Confidentiality without Encryption, CryptoBytes (RSA Laboratories), volume 4, number 1 (summer 1998), 12–17.

Environment Setting



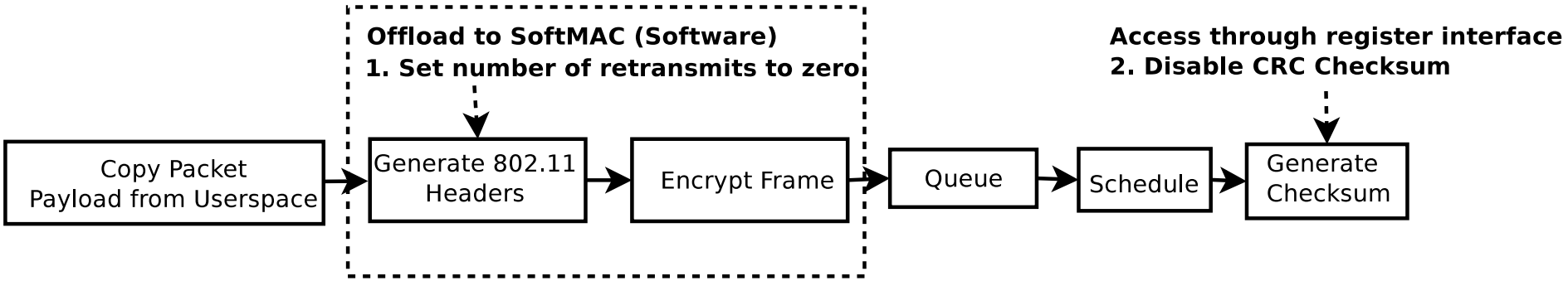
Security Goals

- **Deniable**
  - Ability to deny the communication
- **Anonymous**
  - Cannot be identified specifically
- **Confidential**
  - Adversary cannot recover message
- **Robustness**
  - Cannot be disrupted

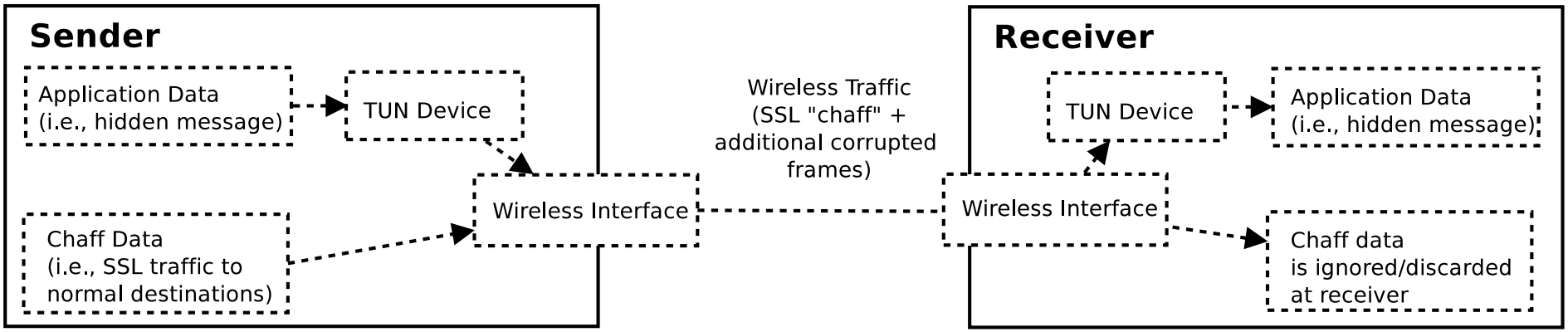
Prototype Implementation

- Write custom tool which runs on Alice and Bob's wireless devices
- Dual wireless chips on a laptop
  - Pre existing (AR9485), USB based (AR9170)
  - Modify inbuilt wireless driver so that it transmits frames with incorrect FCS
  - USB wireless chip works functions as usual

Two Modifications



Communication Channel

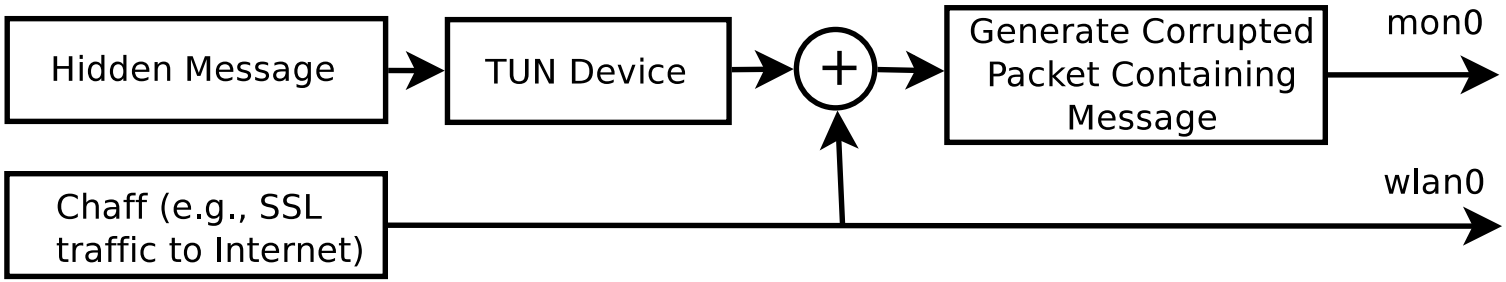


- Sender encrypts message using recipient's public key and hides message in existing SSL *cover* stream
- Wireless interface adds incorrect checksum to packet
  - Access Point discards *corrupted* frames
  - Recipient overhears corrupted frames and can separate grain from chaff

Embedding Secret Messages

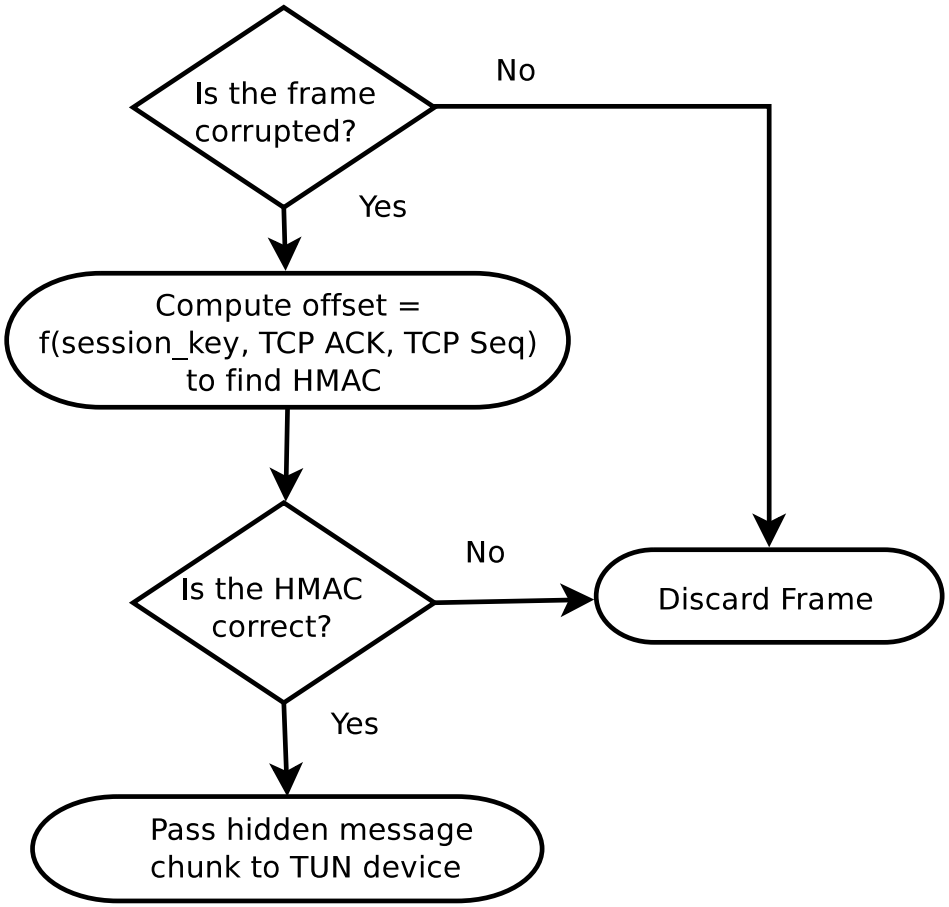
- The secret messages from the application are encrypted using public key cryptography
- Encrypted message is embedded in SSL payload of the 802.11 frame
- Incorrect checksum at Layer 2 is attached to the while it is transmitted in the air

Sending a Message

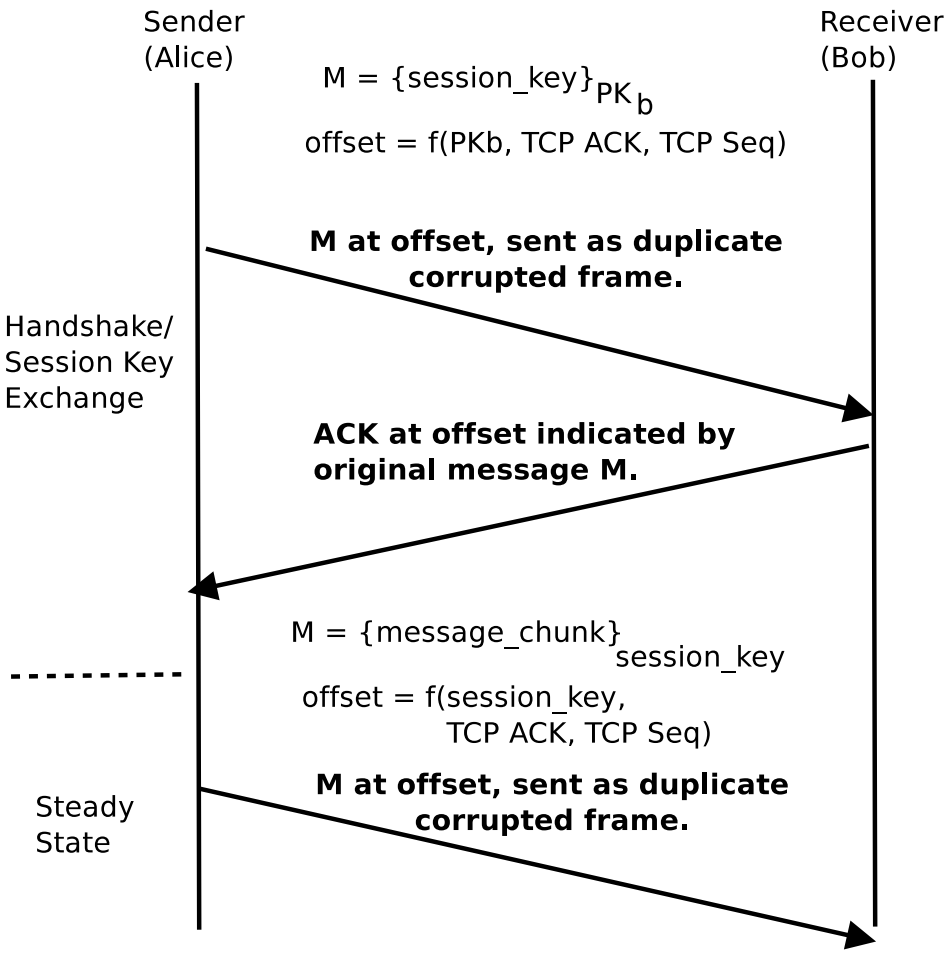


- Hidden message sent through TUN device (fix no item)
- Combined with duplicate copies of real application traffic to generate corrupted duplicates
- Corrupted frames sent on second interface

Receiving a message



Steps in Protocol



Traffic Characteristics

- Packet error rates
- Higher packet error rates, higher the chances of deniability
- Higher the packet corruption rate, better the performance of Denali

Corruptions in frames: Considerations

- Corruption of bits occur in blocks
  - Channel is not memoryless
  - Interference occurs in bursts
- Bits further into the frame are more likely to be corrupted
  - Lack of receiver-sender synchronization

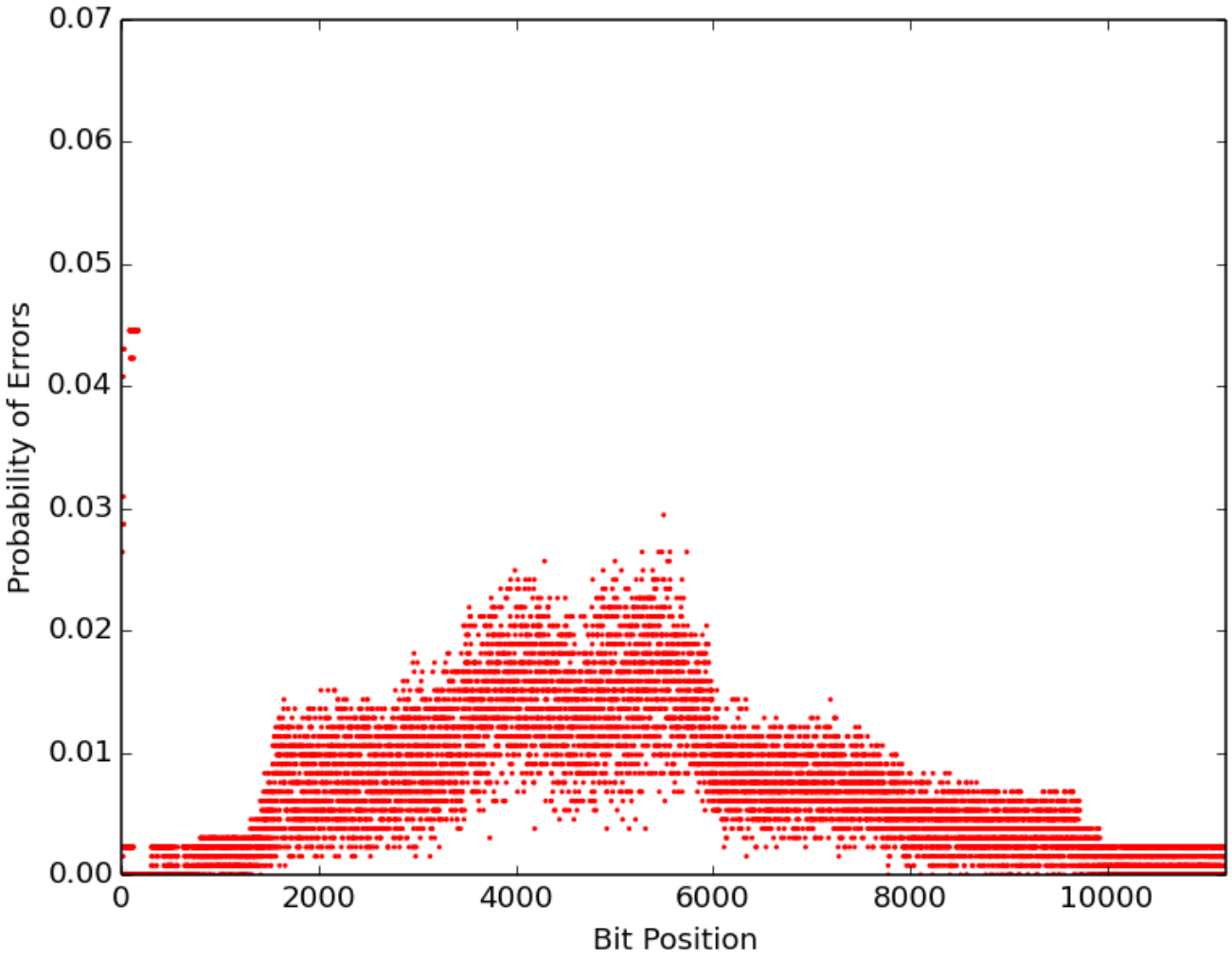
Evaluation

- Present the error distribution patterns to the adversary indicating two worlds
  - in presence of secret communication
  - in absence of secret communication
- Error distributions should be indistinguishable
  - Bit Error distribution
  - Packet Error distribution

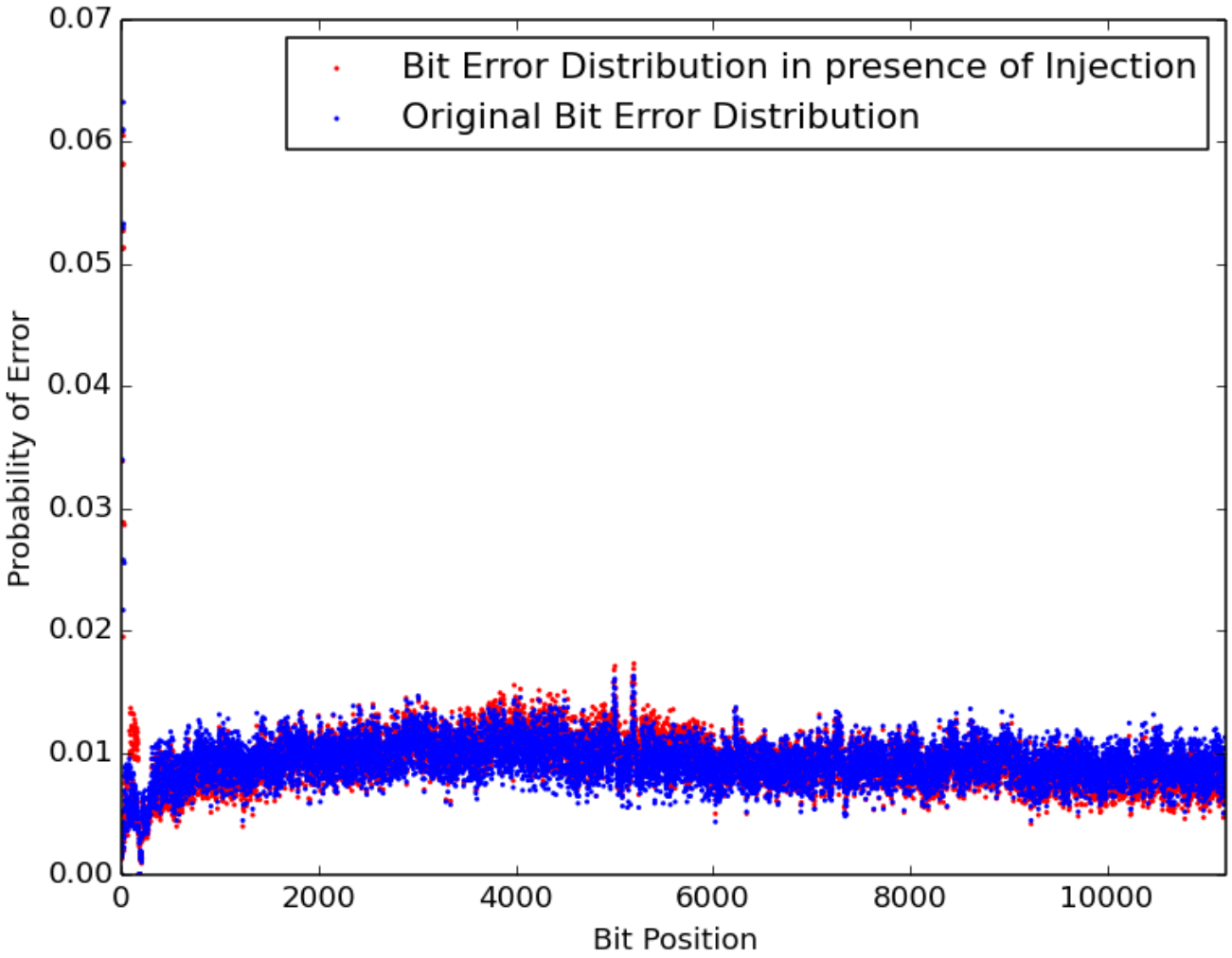
Quantifying Deniability

- Let $f'(x)$ be the discrete probability error distribution with secret communication
- Let $f(x)$ be the discrete probability error distribution without secret communication
- $\epsilon = \frac{1}{B} \cdot \Sigma |f'(x) - f(x)|$
- ϵ is the bitwise L1 distance between two distributions
- B is the size of the packet in bits
- $\sum_{x=0}^{B} |f'(x) - f(x)|$ is the normalized L1 distance between the distributions
- The lower the ϵ, the more deniable the communication

Bit Error Distribution with Denali



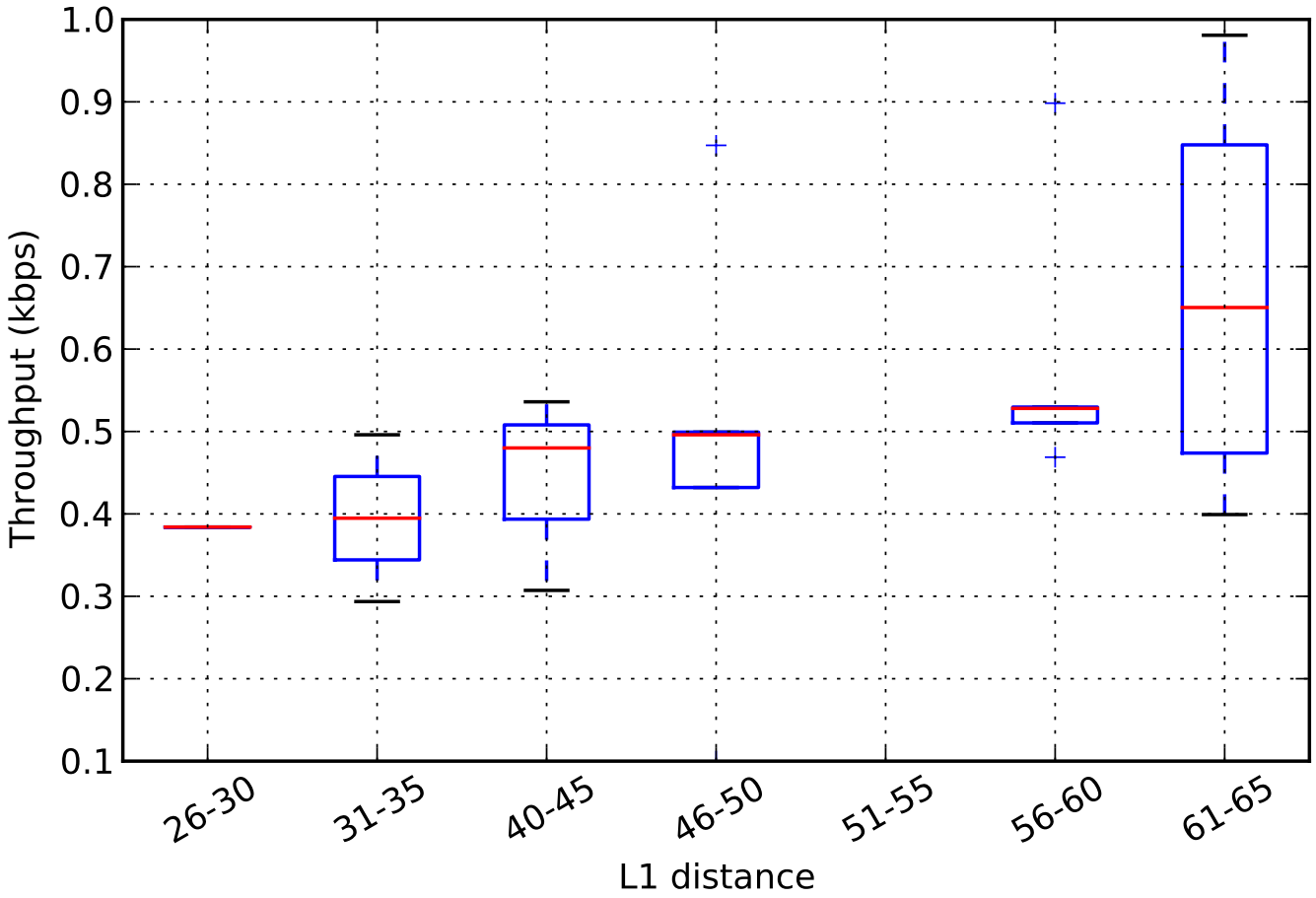
Matching Bit Error Distribution



Deniability vs Throughput

- As we inject more packets into the channel, the L1 distance of the new distribution is going to increase from the noise (assume the distribution of corrupted bits is same)
- The more the throughput, the higher the chances of secret messages being detected

Deniability vs Throughput



Conclusion and Future Work

- Denali provides Deniability, Confidentiality, Robustness
- Main idea: Hide messages in corrupted frames
- Sender and receiver can separate *chaff*
- Overhead is significant, but existing traffic can provide the cover
- Extending Denali to Mobile devices and Multi-hop Networks

Thanks!

- Questions

References I