2004 IEEE Symposium on Security and Privacy

Large-Scale IP Traceback in High-Speed Internet

Jun (Jim) Xu

Networking & Telecommunications Group

College of Computing

Georgia Institute of **Tech**nology

(Joint work with Jun Li, Minho Sung, Li Li)

Introduction

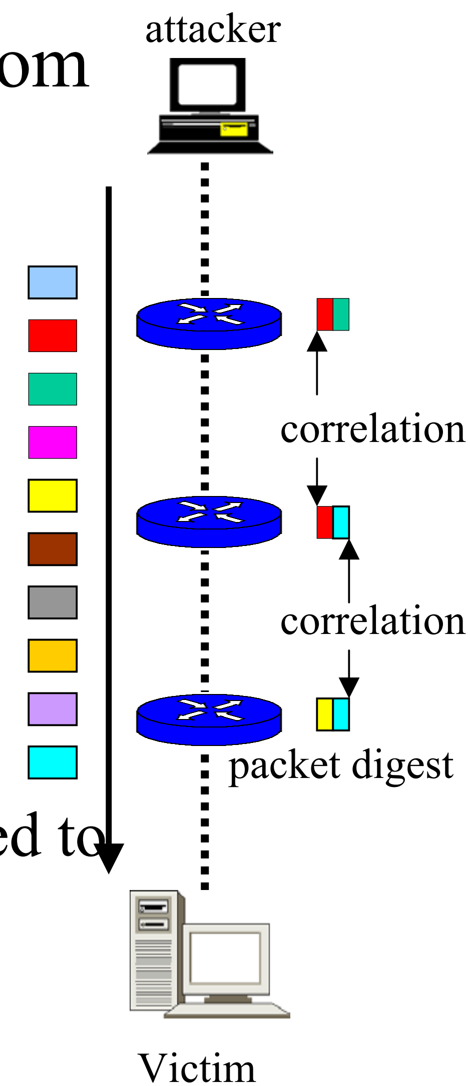
- Internet DDoS attack is an ongoing threat
  - on websites: Yahoo, CNN, Amazon, eBay, etc (Feb. 2000)
  - on Internet infrastructure: 13 root DNS servers (Oct, 2002)
- It is hard to identify attackers due to IP spoofing
- IP Traceback: trace the attack sources despite spoofing
- Two main types of proposed traceback techniques
  - Probabilistic Packet Marking schemes: routers put stamps into packets, and victim reconstructs attack paths from these stamps [Savage et. Al. 00] …… [Goodrich 02]
  - Hash-based traceback: routers store bloom filter digests of packets, and victim query these digests recursively to find the attack path [Snoeren et. al. 01]

Scalability Problems of Two Approaches

- Traceback needs to be scalable
  - When there are a large number of attackers, and
  - When the link speeds are high
- PPM is good for high-link speed, but cannot scale to large number of attackers [Goodrich 01]
- Hash-based scheme can scale to large number of attackers, but hard to scale to very high-link speed
- Our objective: design a traceback scheme that is scalable in both aspects above.

Design Overview

- Our idea: same as hash-based, but store bloom filter digests of sampled packets only
  - Use small sampling rate *p* (such as 3.3%)
  - Small storage and computational cost
  - Scale to 10 Gbps or 40 Gbps link speeds
  - Operate within the DRAM speed
- the challenge of the sampling
  - Need many more packets for traceback
  - Independent random sampling will not work: need to improve the "*correlation factor*"

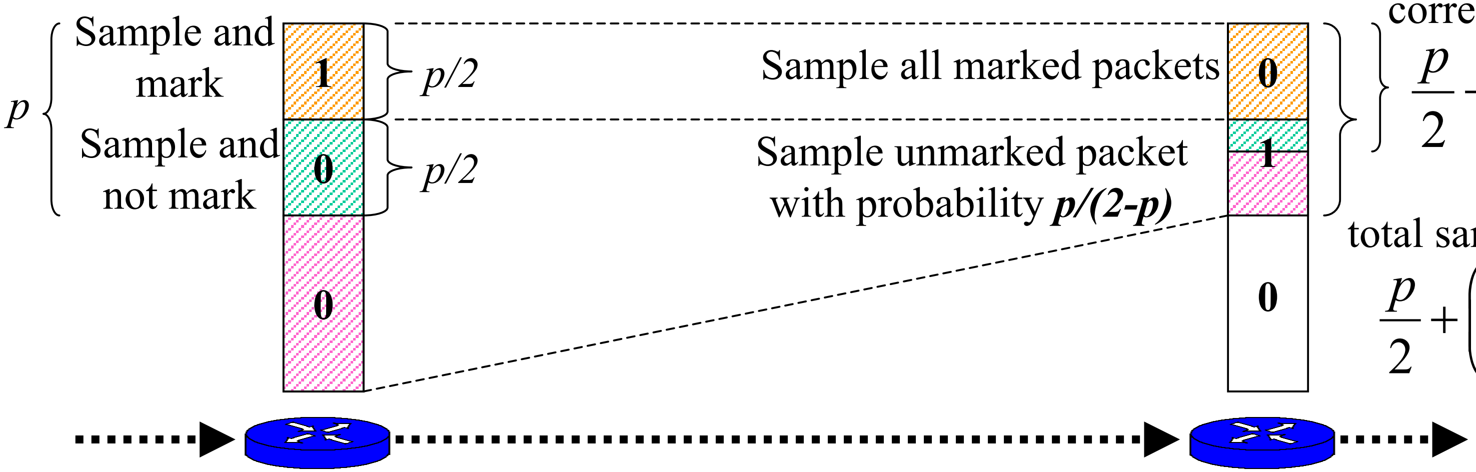


Overview of our hash-based traceback scheme

- Each router stores the bloom filter digests of sampled packets
- Neighboring routers compare with each other the digests of the packets they store for the traceback to proceed
  - Say P is an attack packet, then if you see P and I also see P, then P comes from me to you …
- When correlation is small, the probability that both see something in common is small

One-bit Random Marking and Sampling(ORMS)

- ORMS make correlation factor be larger than 50%
- ORMS uses only one-bit for coordinating the sampling among the neighboring routers



correlation :

$$\frac{p}{2} + \frac{p}{2} \cdot \frac{p}{2-p} = \frac{p}{2-p}$$

total sampling probability :

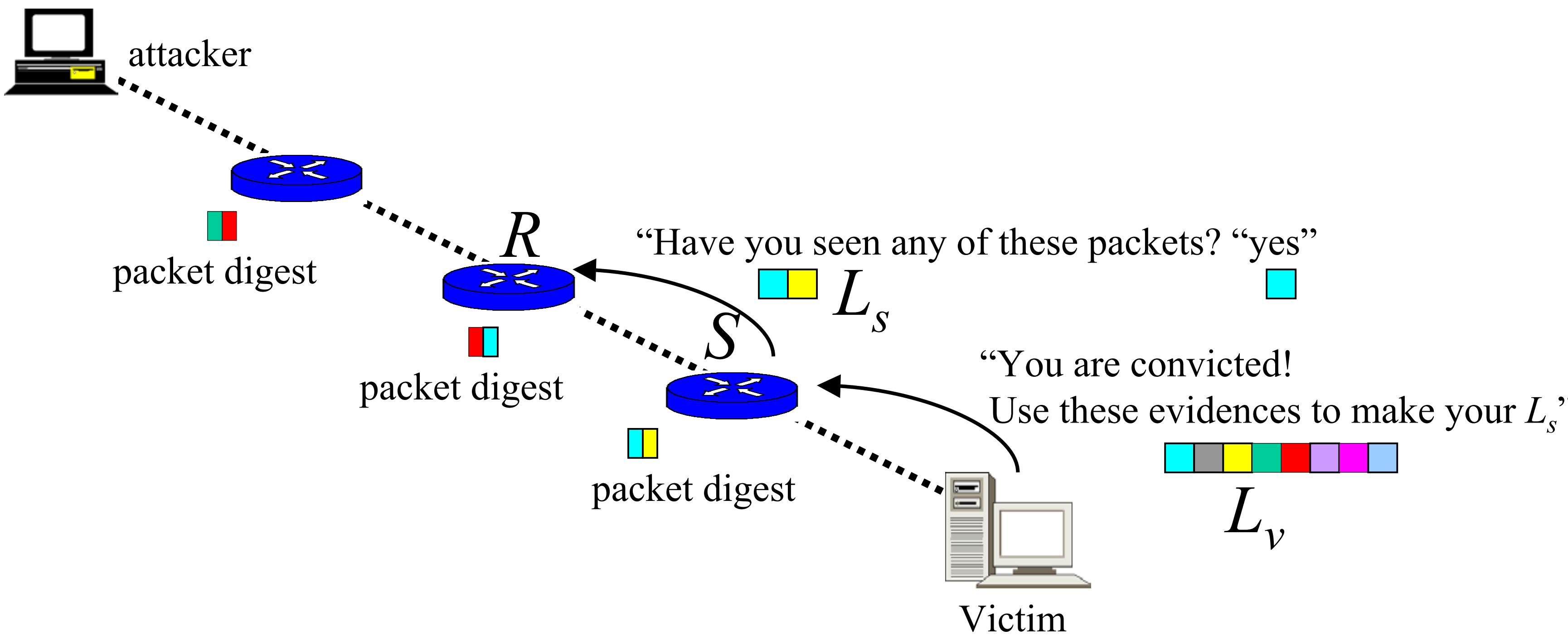
$$\frac{p}{2} + \left(1 - \frac{p}{2}\right) \cdot \frac{p}{2-p} = p$$

correlation factor (sampled by both) :
(> 50% because *0<p<1*)

$$\left(\frac{p}{2-p}\right) / p = \frac{1}{2-p}$$

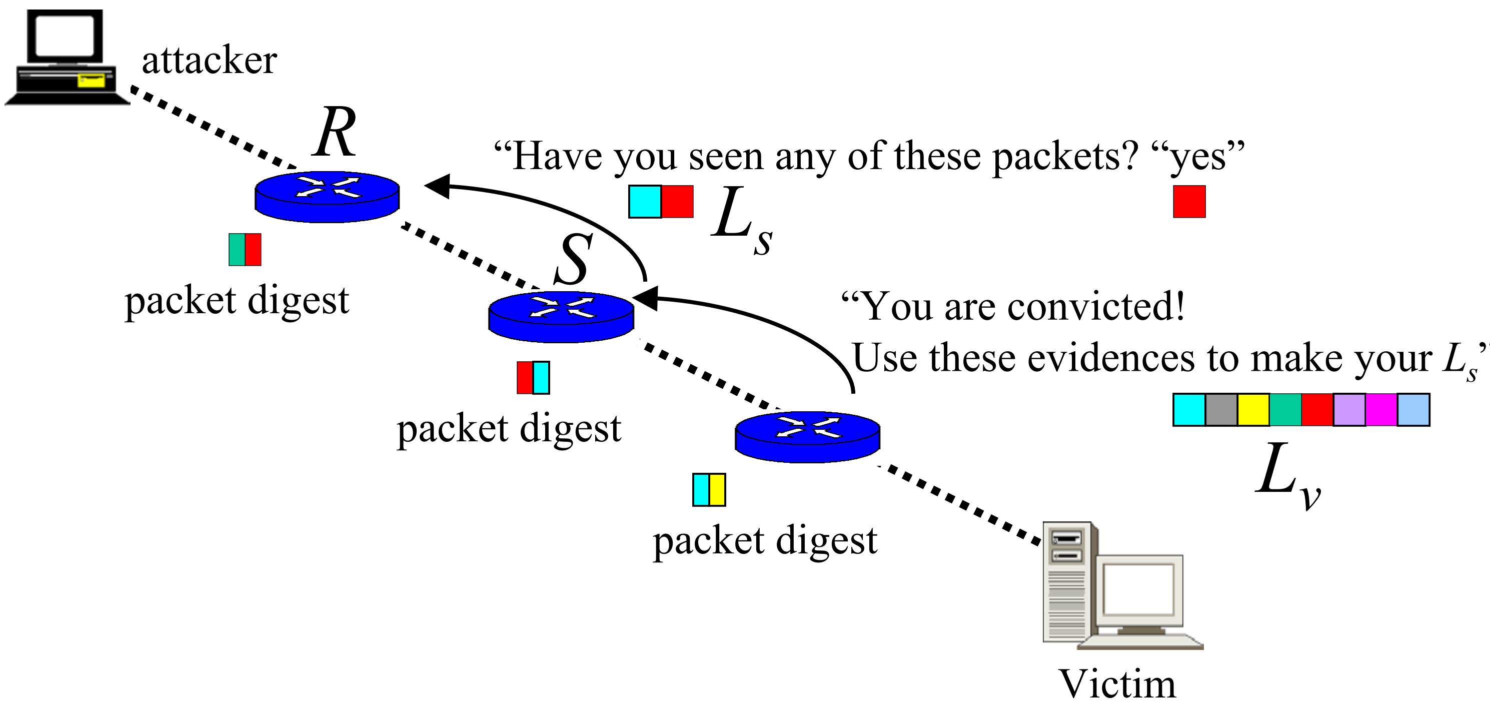
Traceback Processing

1. Collect a set of attack packets L_v
2. Check router S, a neighbor of the victim, with L_v
3. Check each router R (neighbor of S) with L_s



Traceback Processing

4. Pass L_v to R to be used to make new L_s
5. Repeat these processes



A fundamental optimization question

- Recall that in the original traceback scheme, the router records a bloom filter of 3 bits for each and every packets
- There are many different ways of spending this 3 bits per packet budget, representing different tradeoff points between size of digest and sampling frequency
  - e.g., use a 15-bit bloom filter but only record 20% of digests (15*20% =3)
  - e.g., use a 12-bit bloom filter but only record 25% of digests (12*25% =3)
  - Which one is better or where is the optimal tradeoff point?
- Answer lies in the information theory

Intuitions from the information theory

- View the traceback system as a communication channel
  - Increasing the size of digest reduces the false positive ratio of the bloom filter, and therefore improving the signal noise ratio (S/N)
  - Decreasing sampling rate reduces the bandwidth (W) of the channel
  - We want to maximize $C = W \log 2 (1+S/N)$
- C is the mutual information – maximize the mutual information between what is "observed" and what needs to be predicted – or minimize the conditional entropy
- Bonus from information theory: we derive a lower bound on the number of packets needed to achieve a certain level of traceback accuracy through **Fano's inequality**

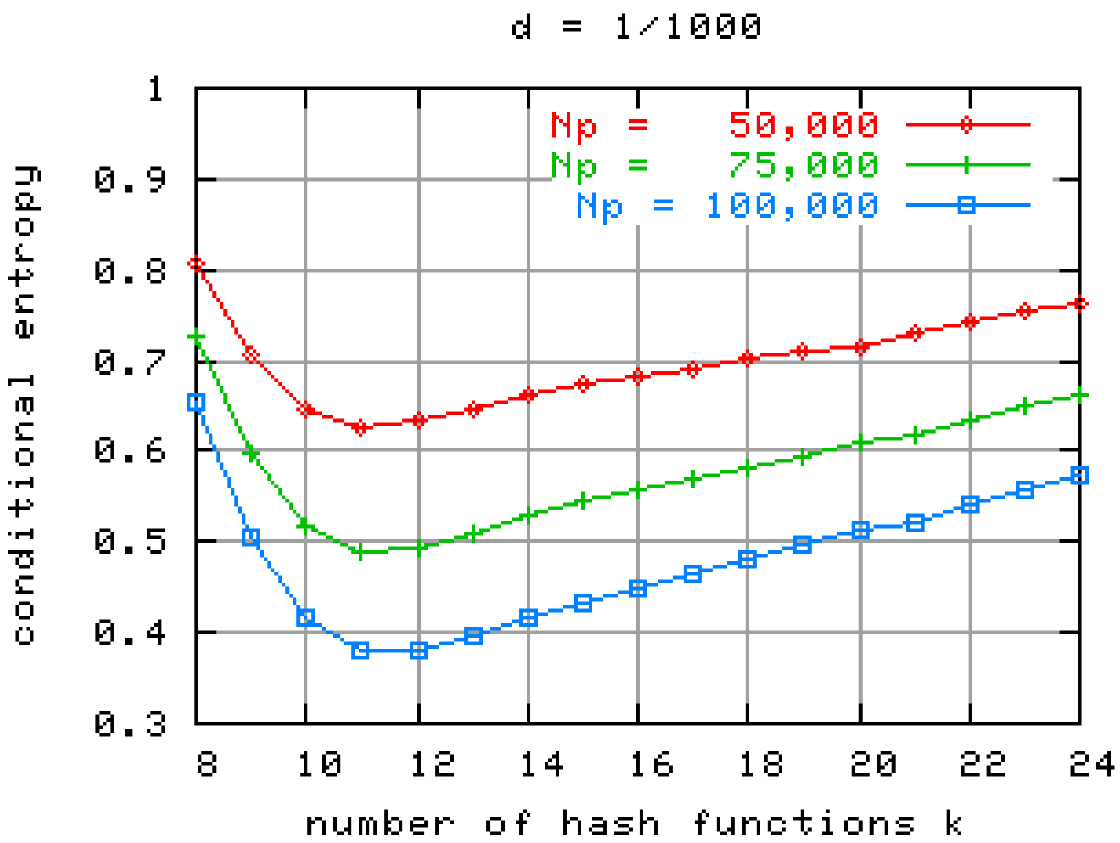
The optimization problem

$$k^* = \underset{k}{\operatorname{argmin}}\ H(Z \mid X_{t1}+X_{f1},\ Y_t+Y_f)$$

subject to the resource constraint ($s = k \times p$)

s: average number of bits "devoted" for each packet
p: sampling probability
k: size the bloom filter digest

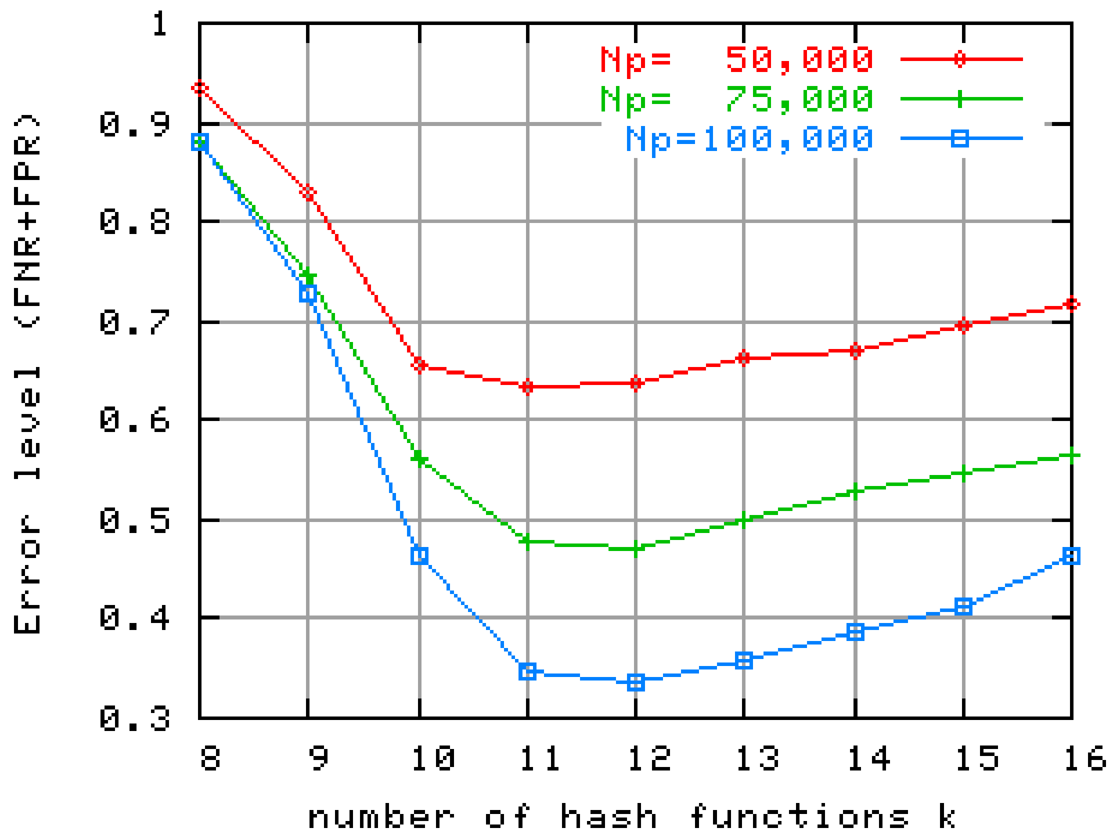
Applications of Information Theory



Resource constraint: $s = k \times p = 0.4$

Verification of Theoretical Analysis

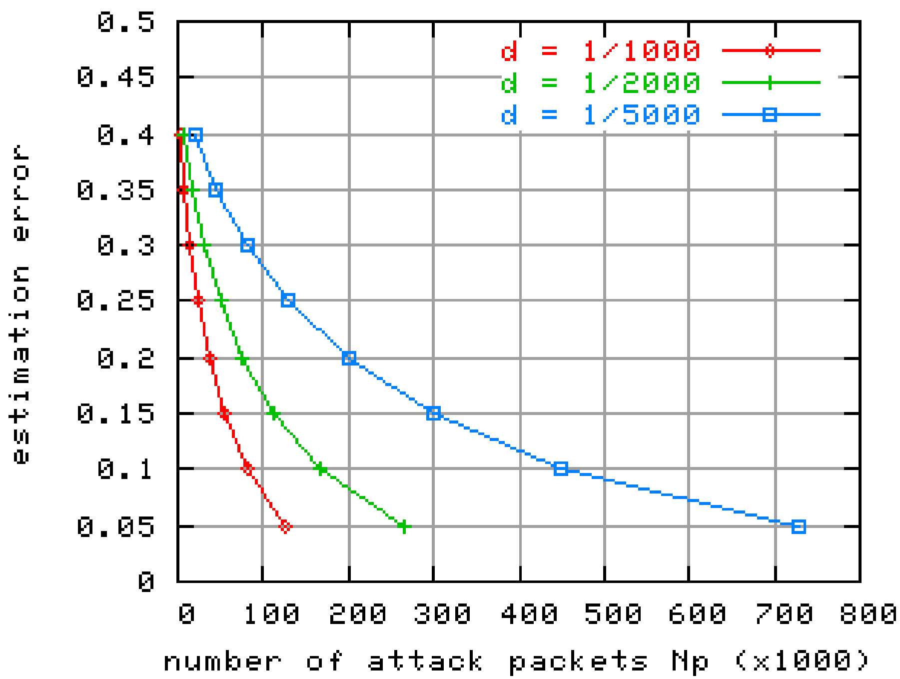
- Parameter tuning



Parameters: 1000 attackers, $s = k \times p = 0.4$

Lower bound through Fano's inequality

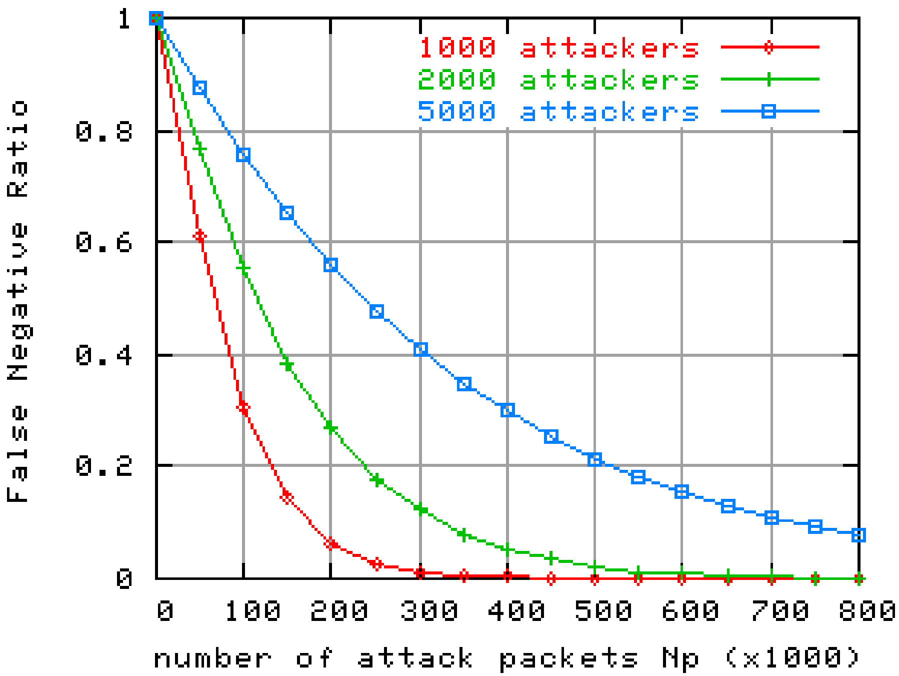
- $H(p_e) \geq H(\, Z \mid X_{t1}+X_{f1},\, Y_t+Y_f\,)$

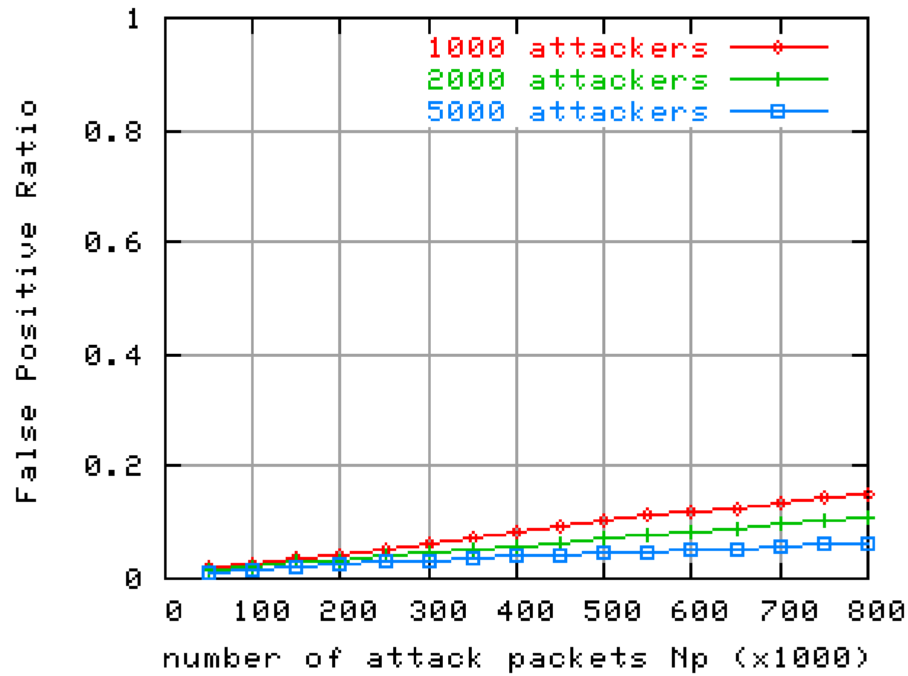


Parameters: s=0.4, k=12, p=3.3% (12 × 3.3% = 0.4)

Simulation results

- False Negative & False Positive on Skitter I topology

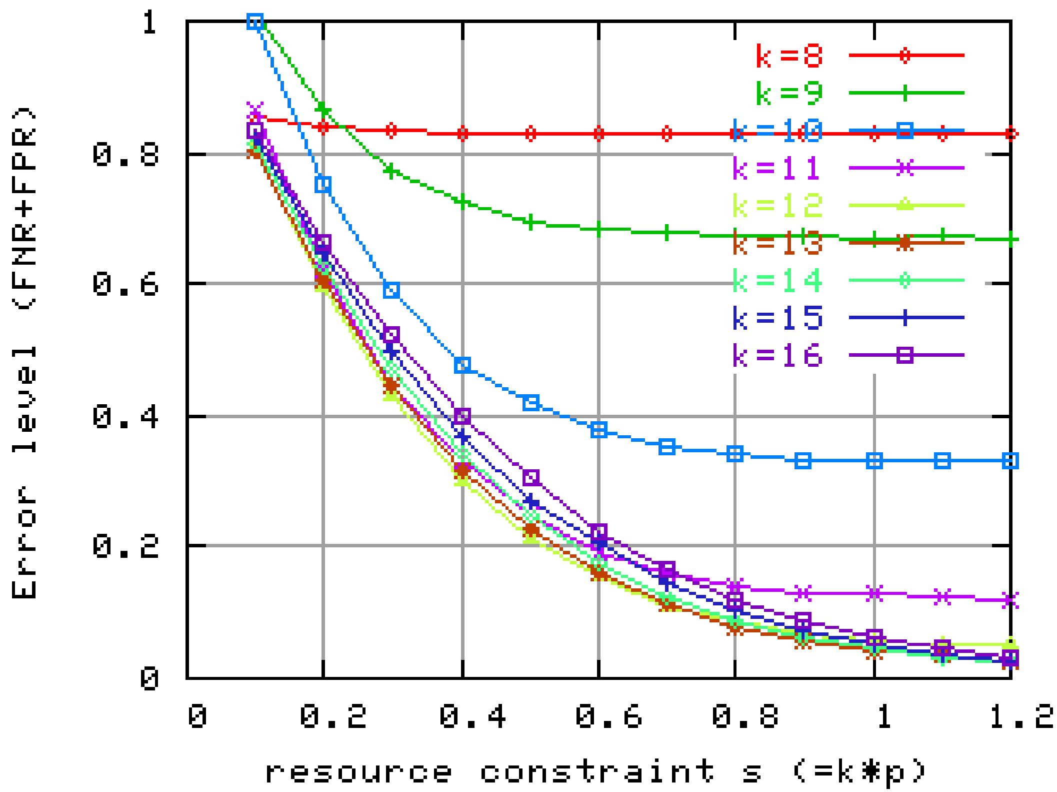




Parameters: s=0.4, k=12, p=3.3% (12 × 3.3% = 0.4)

Verification of Theoretical Analysis

- Error levels by different *k* values



Parameters: 2000 attackers, N_p=200,000

Future work and open issues

1. Is correlation factor *1/(2-p)* optimal for coordination using one bit?
2. What if we use more that one bit for coordinating sampling?
3. How to optimally combine PPM and hash-based scheme – a Network Information Theory question.
4. How to know with 100% certainty that some packets are attack packets? How about we only know with a certainty of p?

Conclusions

- Design a sampled hash-based IP traceback scheme that can scale to a large number of attackers and high link speeds
- Addressed two challenges in this design:
  - Tamper-resistant coordinated sampling to increase the "correlation factor" to beyond 50% between two neighboring routers
  - An information theory approach to answer the fundamental parameter tuning question, and to answer some lower bound questions
- Lead to many new questions and challenges