A Tradeoff between Caching Efficiency and Data Protection for Video Services in CCN

2014. 2. 23.

Eunsang Cho*, Jongho Shin, Jaeyoung Choi,
Ted “Taekyoung” Kwon, Yanghee Choi

escho@mmlab.snu.ac.kr

Network Convergence & Security Lab,
Seoul National University, Korea

BACKGROUND AND STATUS QUO

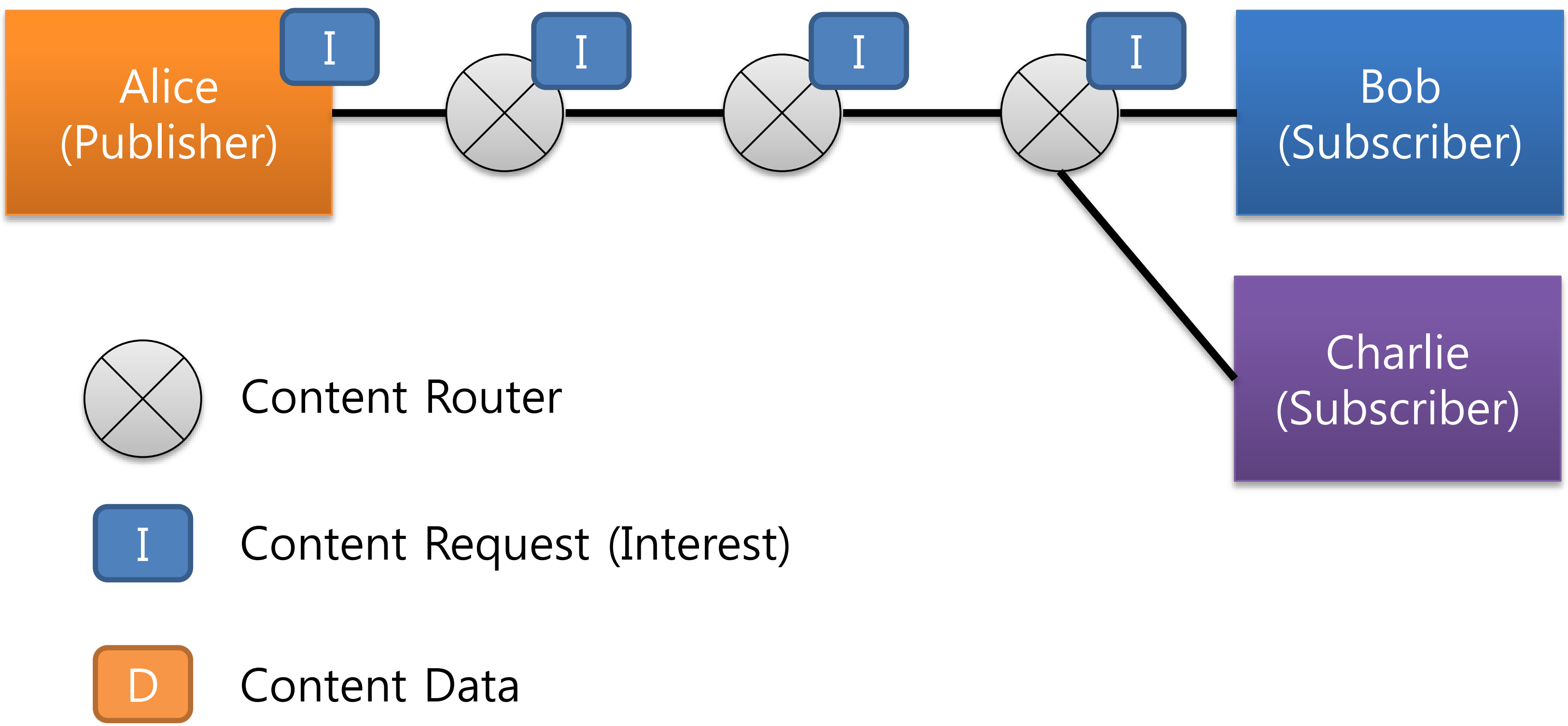
Background

- **Video content** is the one of major data sources with *massive volume*.

- **CCN** (Content-Centric Networking) is able to handle the video content well, thanks to *in-network caching*.

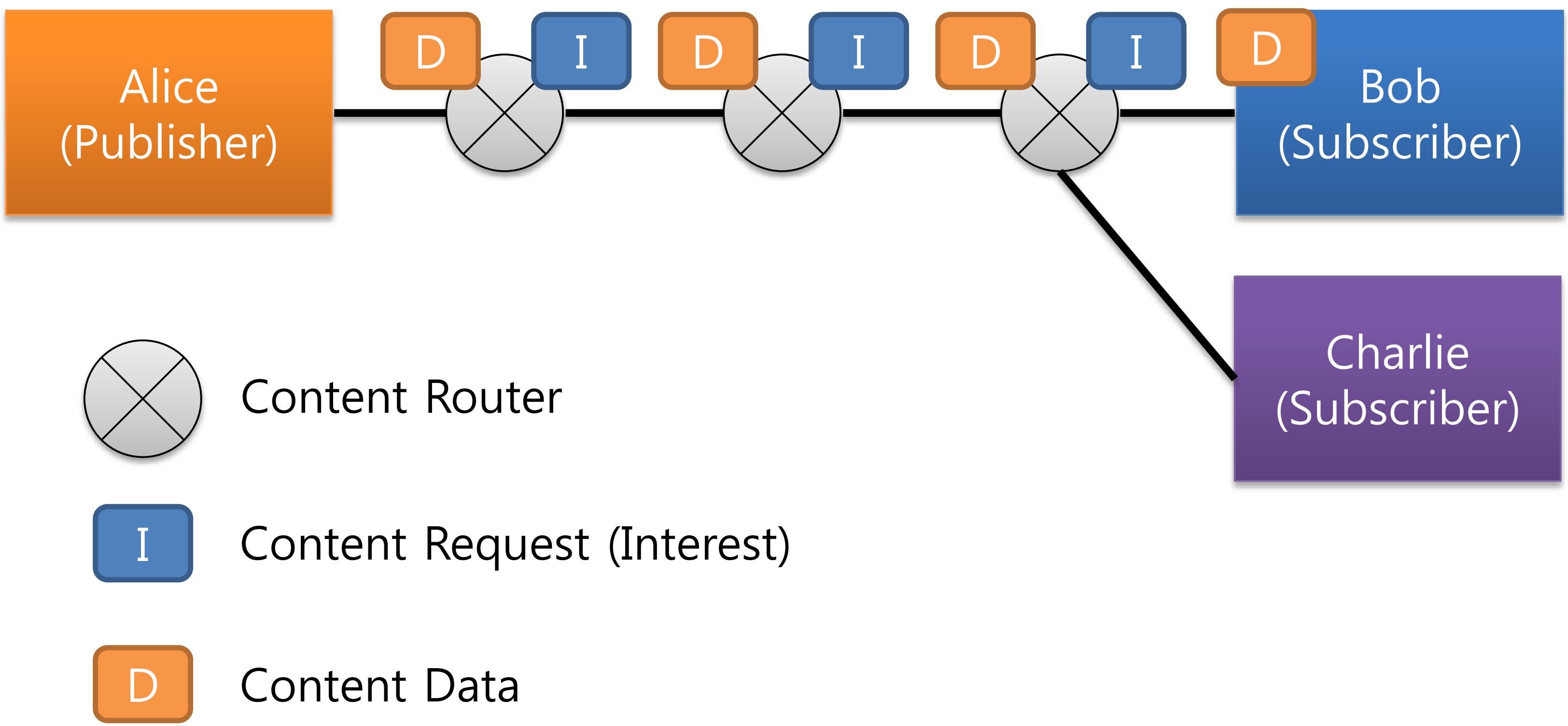
CCN In-network Caching

- First content request (Interest): from Bob to Alice



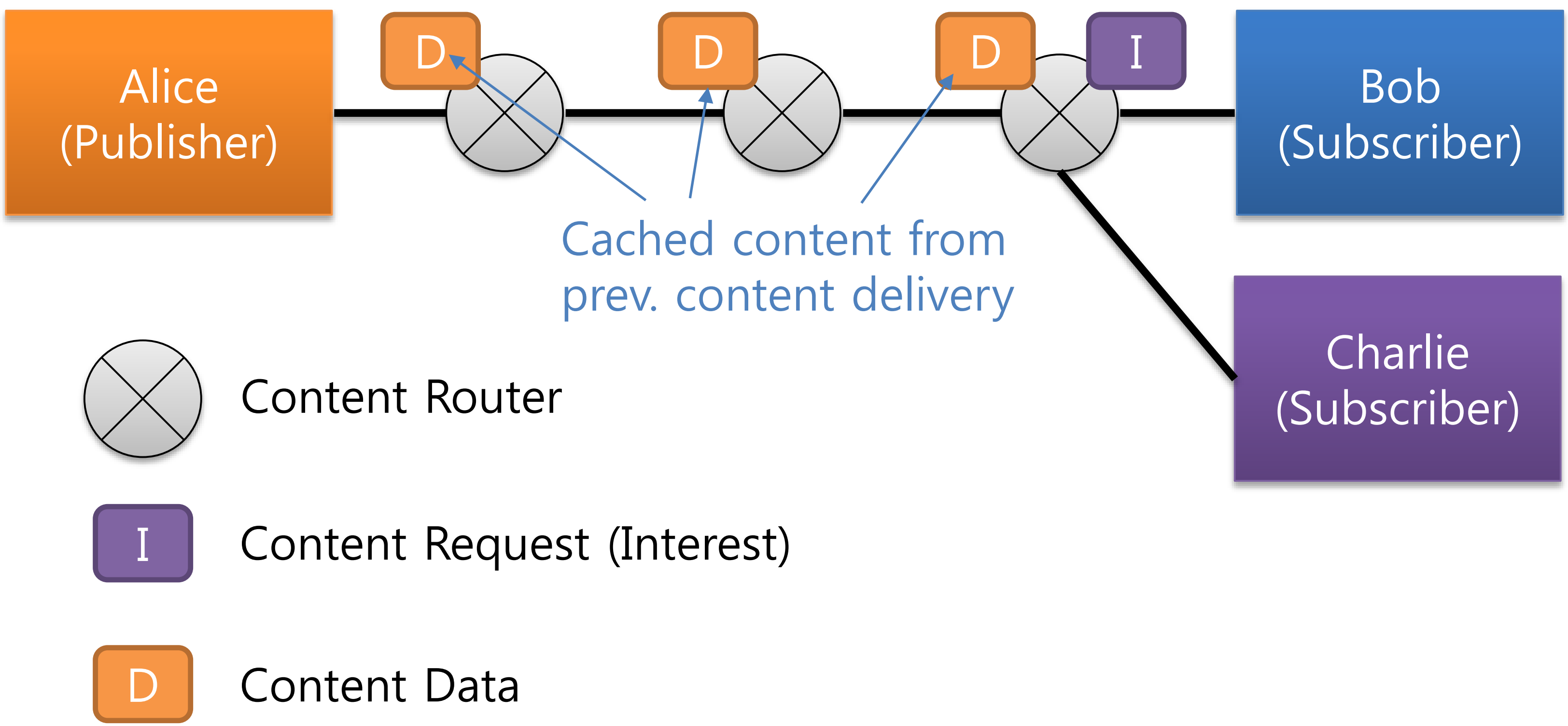
CCN In-network Caching

- First content delivery: from Alice to Bob



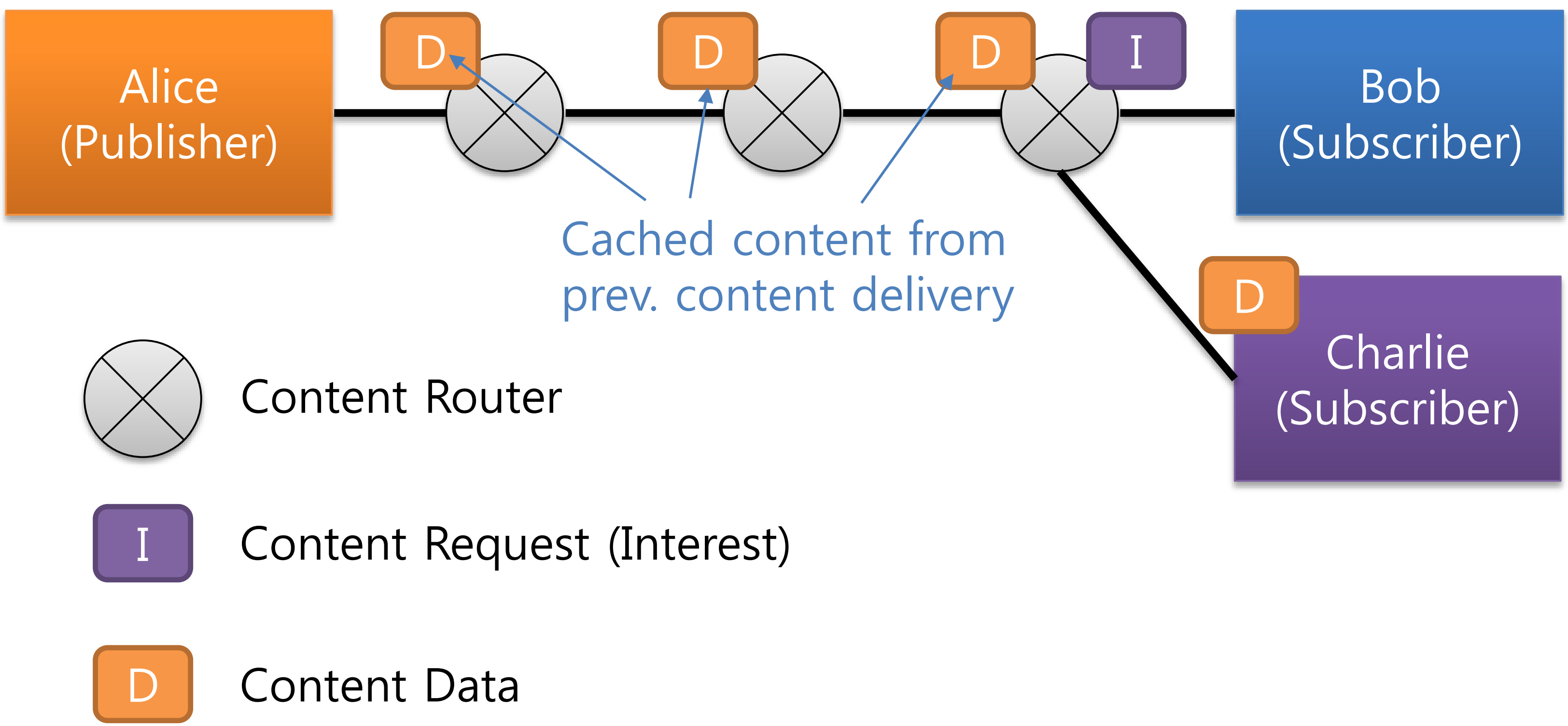
CCN In-network Caching

- Second content request (Interest): from Charlie to Alice



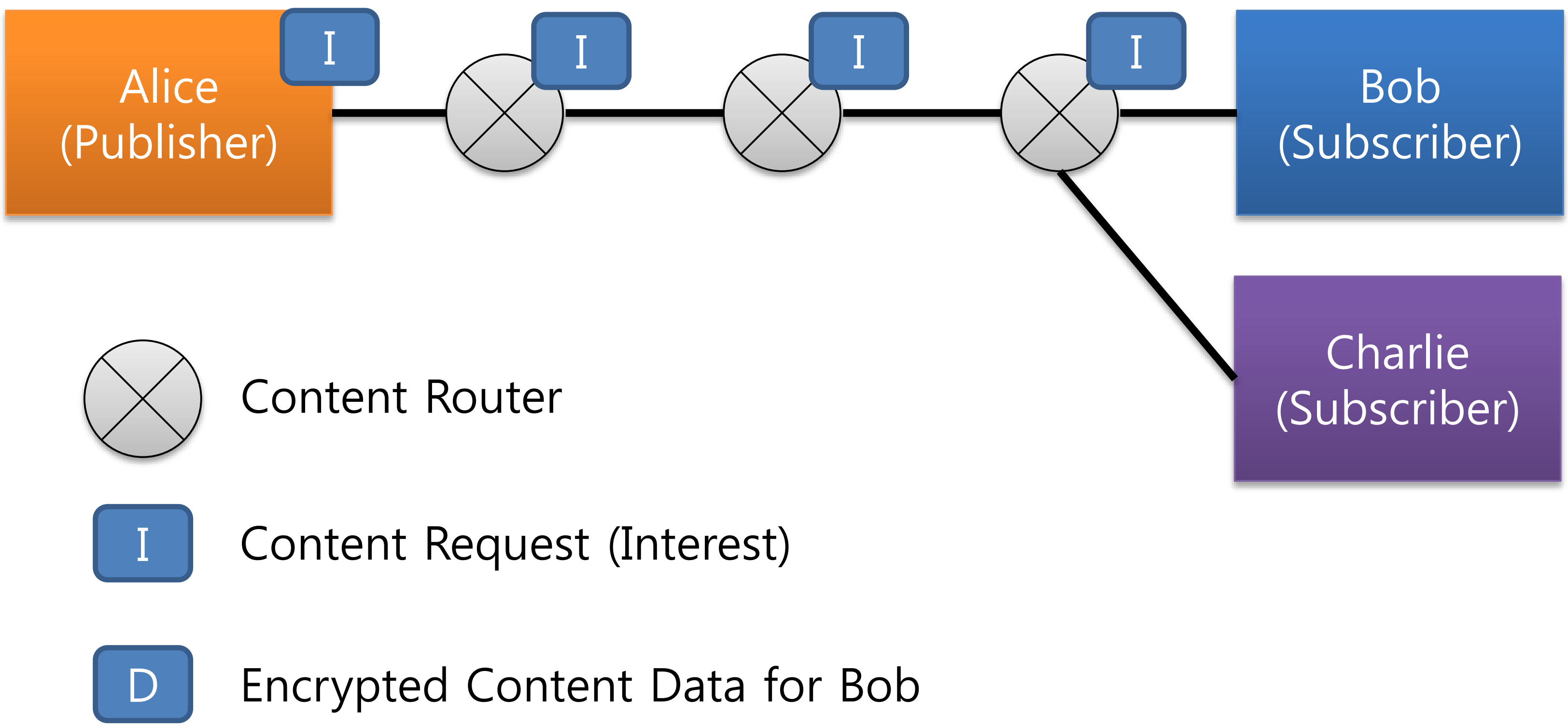
CCN In-network Caching

- Second content delivery: from cache to Charlie



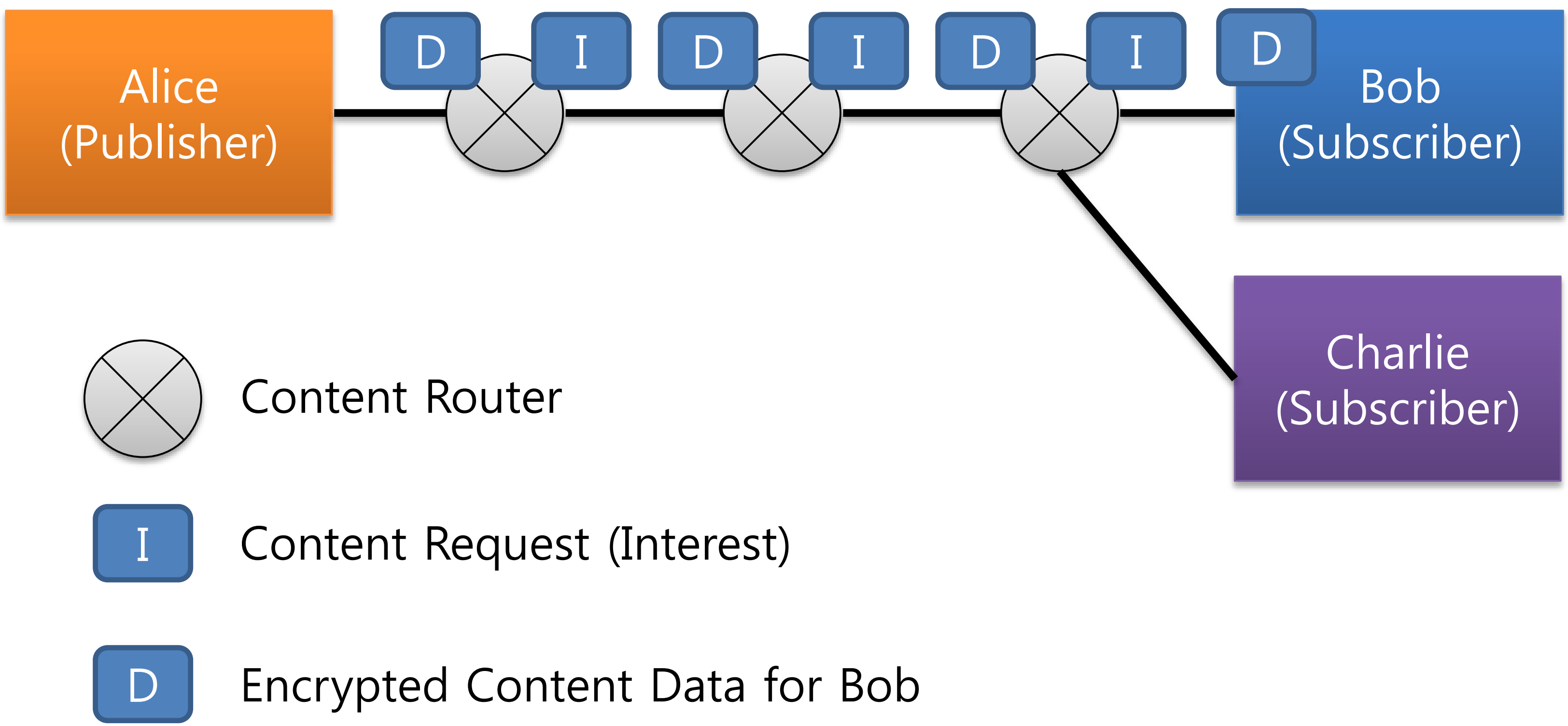
With Encryption

- First content request (Interest): from Bob to Alice



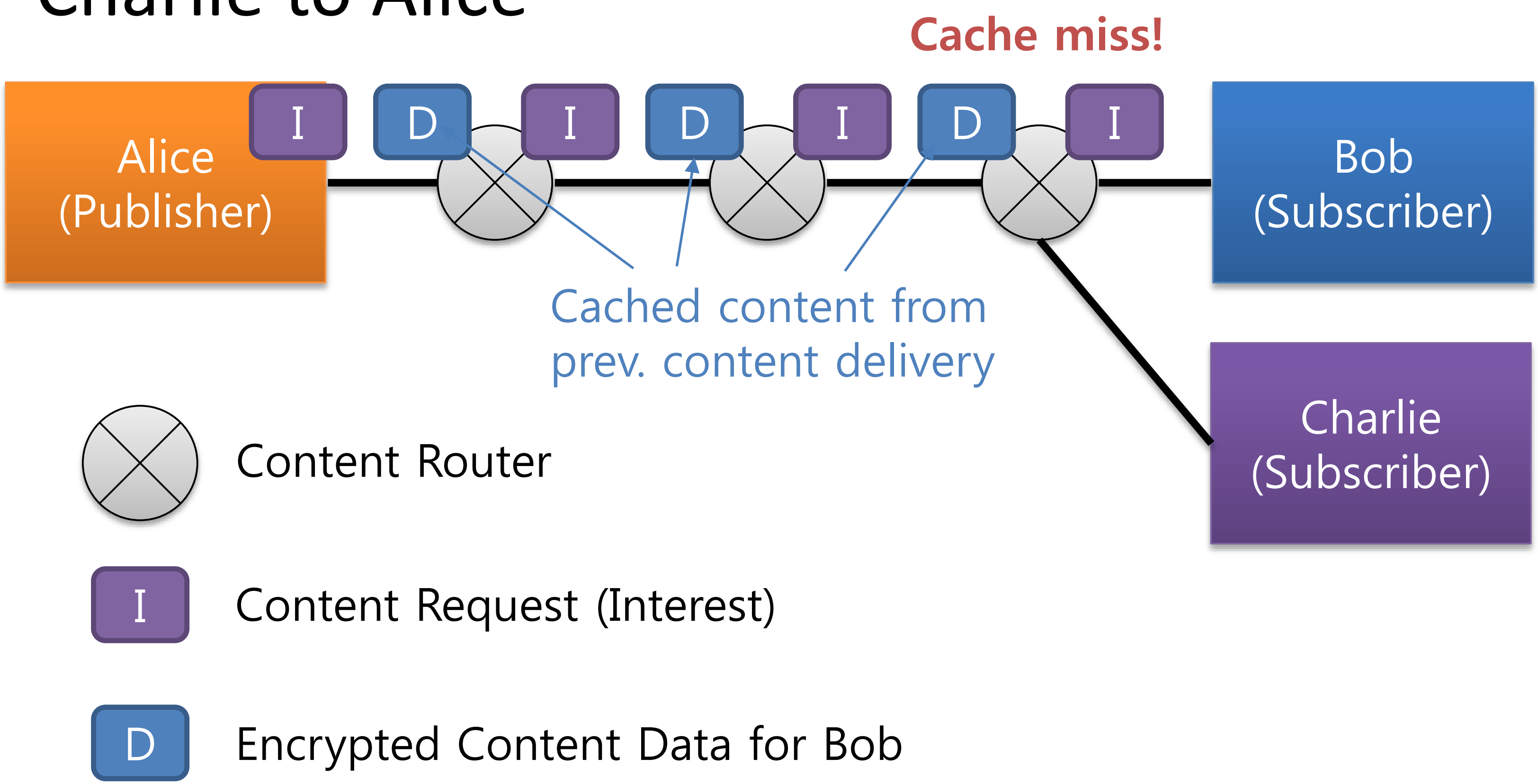
With Encryption

- First content delivery: from Alice to Bob



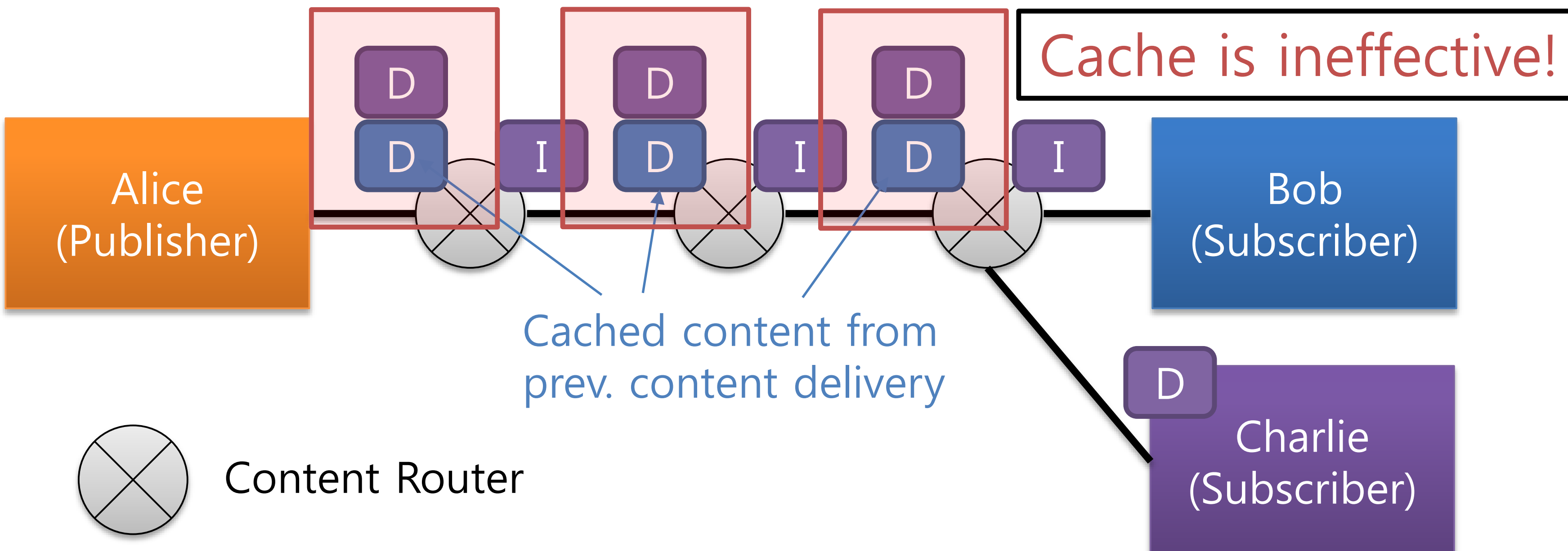
With Encryption

- Second content request (Interest): from Charlie to Alice



With Encryption

- Second content delivery: from Alice to Charlie



Content Router

I Content Request (Interest)

D Encrypted Content Data for Bob

D Encrypted Content Data for Charlie

Problem Definition

- **End-to-end data encryption** for each different content subscriber makes *caching ineffective*.
  - A novel video encryption scheme for CCN is required.

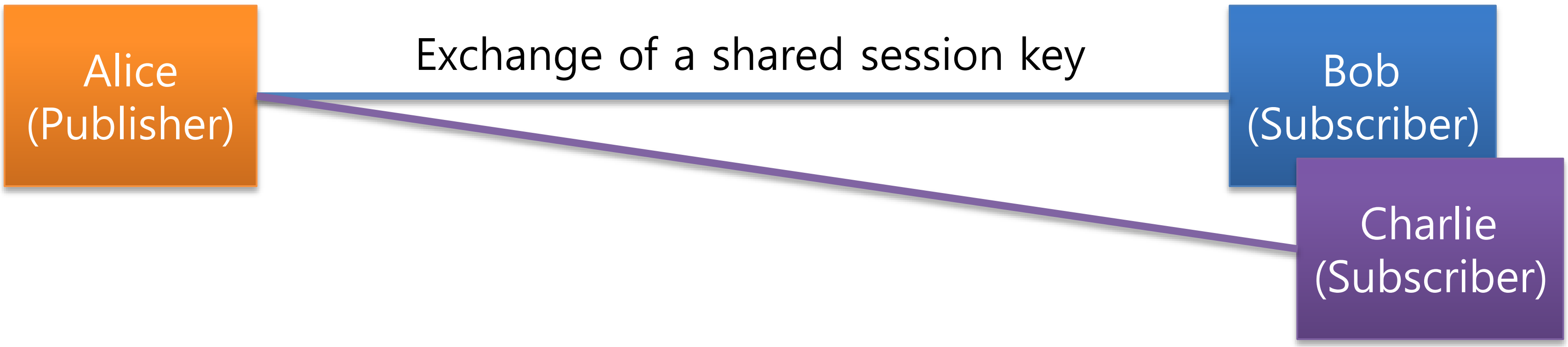
Objectives

- The objectives of this research are:
  - To develop a video encryption scheme which can *utilize caching feature* of CCN
  - To provide a practical approach for *video content protection*
  - To customize protection levels by *video content provider's requirements*

→ To provide tradeoffs between data protection level, decodability of video, and cache effectiveness

Status Quo

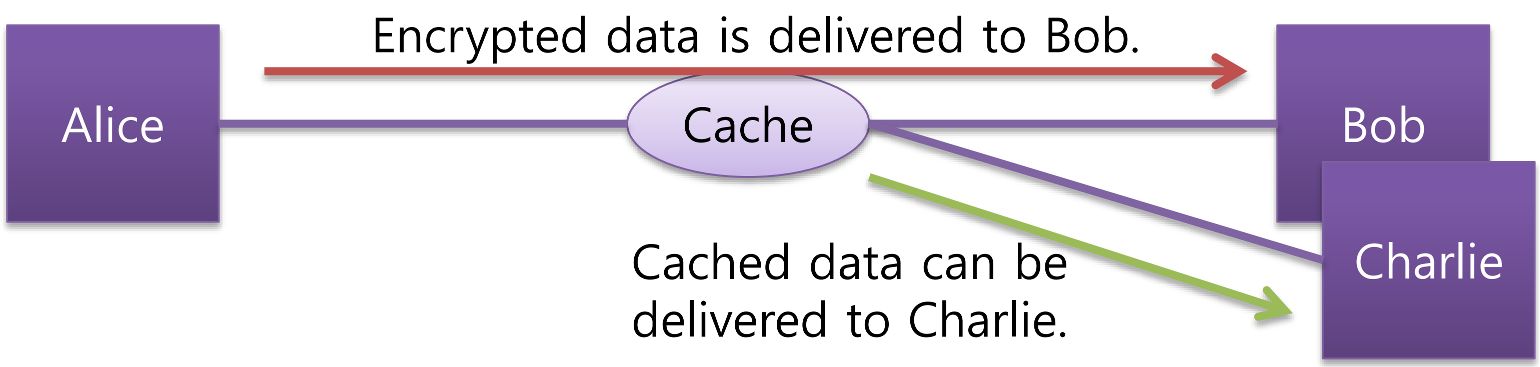
- Transport Layer Security (TLS)



- Limitations
  - One-time validity of encrypted data
  - Ineffectiveness of in-network caching

Status Quo

- Shared & symmetric key cryptography

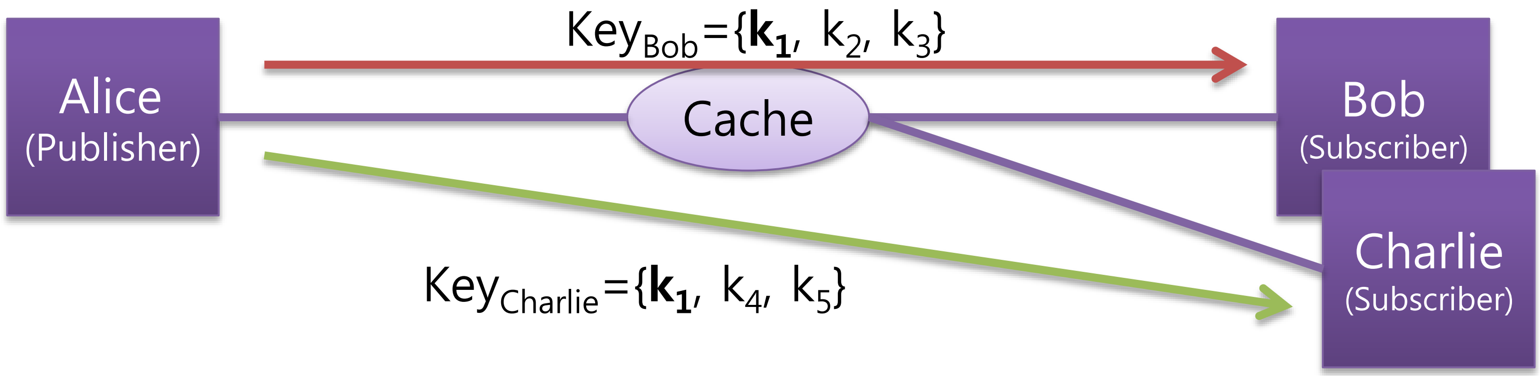


- Limitations
  - Key leakage problem
  - Untraceability of piracy

OUR PROPOSED SCHEME

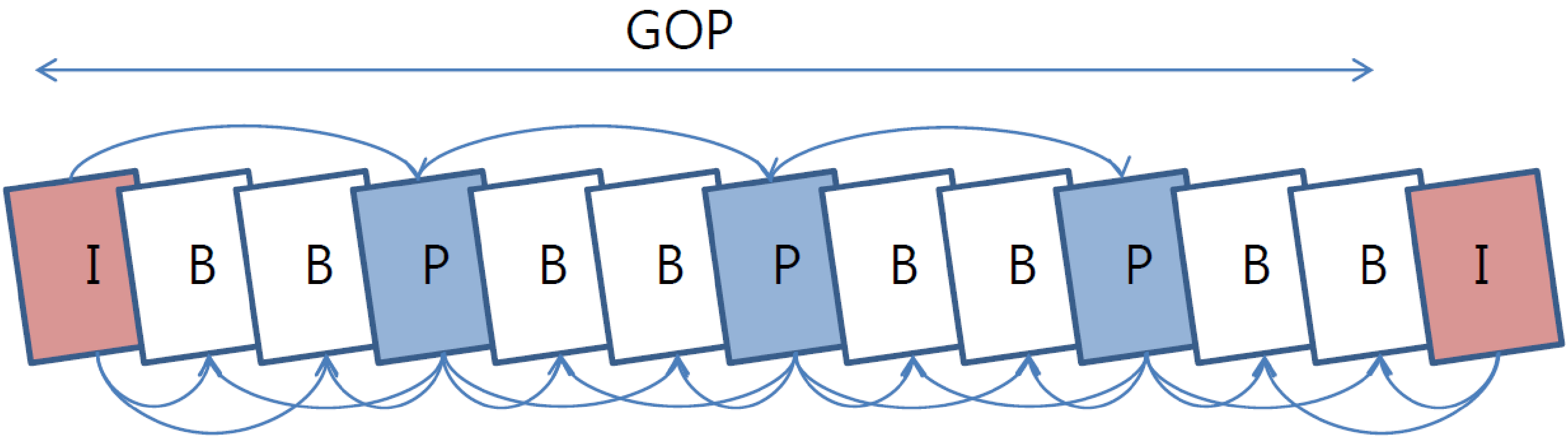
Our Approach

- Access control with multiple symmetric keys
  - Distinct set of keys is assigned to each user
    - **Tracing feature** against *key leakage problem (piracy)*
  - Some keys can be shared among users
    - Subset of content can be shared by caching



Utilizing MPEG Video Structure

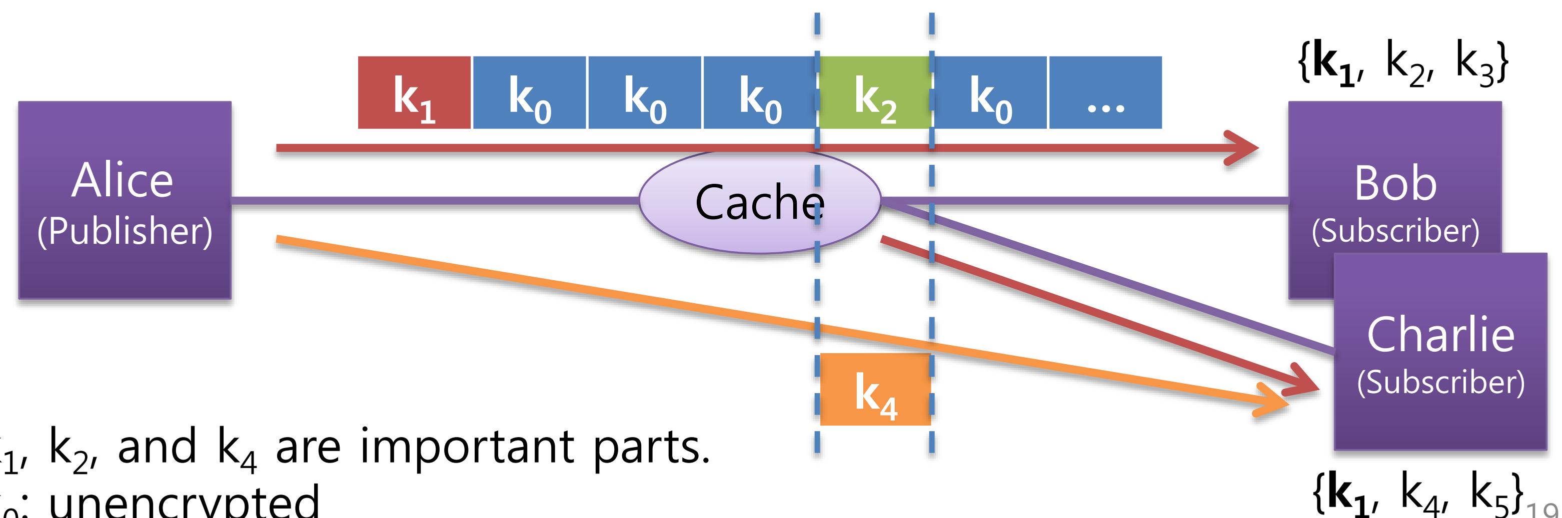
- MPEG video structure



A sample GOP sequence of MPEG video: GOP(12, 3)

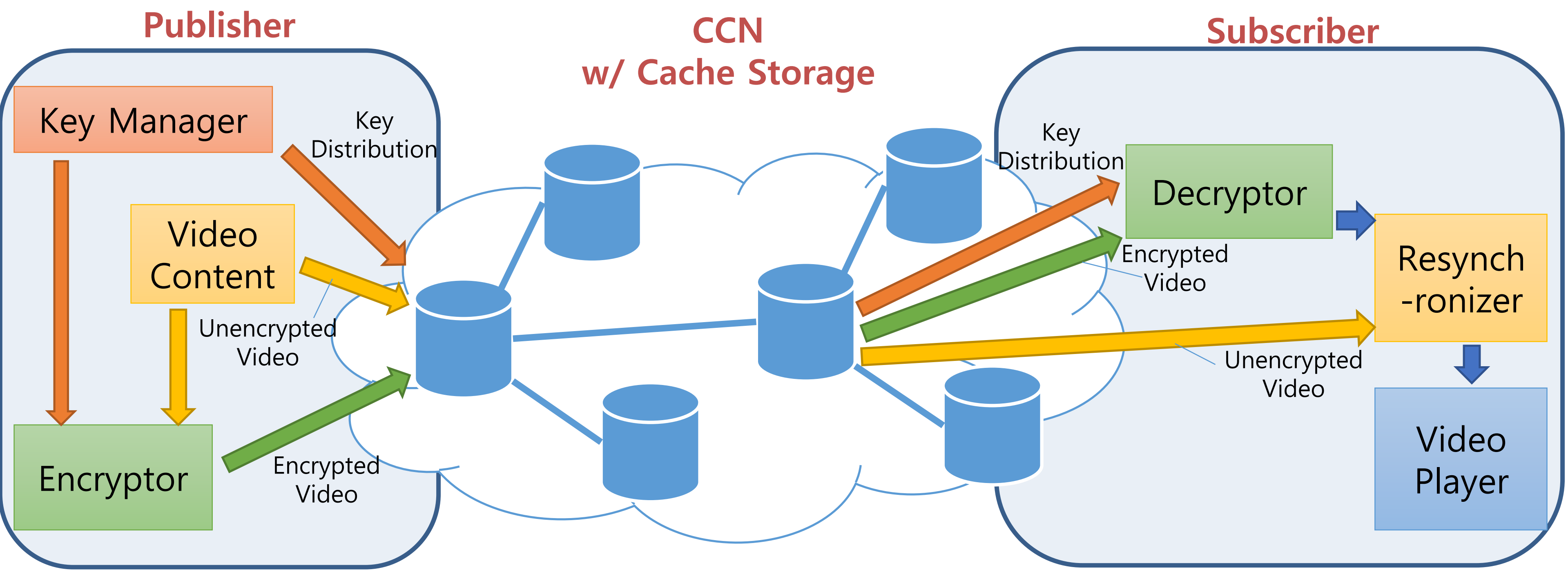
Our Approach

- Video compression feature
  - From the structure of a MPEG video, some parts, such as **I-frames** are *more important* than others
    - Decrypting B- and P-frames requires I-frames
  - For higher cache utilization, *less important parts* can be **left unencrypted**

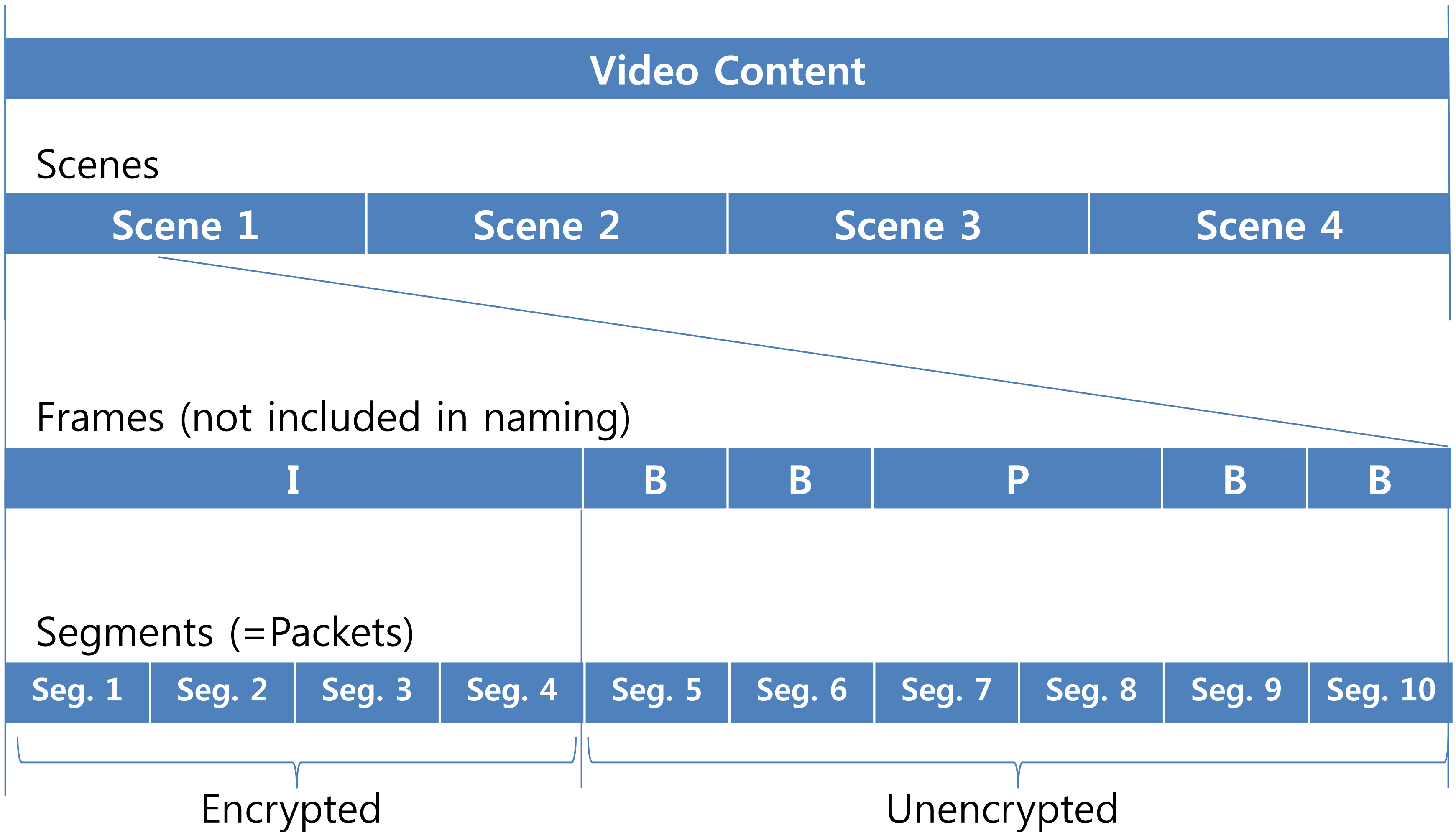


k_1, k_2, and k_4 are important parts.
k_0: unencrypted

Overview of the Framework



Naming Model



Operation Overview

1. Subscriber S requests her own set of keys for video.

2. Publisher P responds w/ multiple symmetric keys $\{k_1, k_2, k_3, \ldots, k_N\}$ and corresponding content names.

k_0: unencrypted

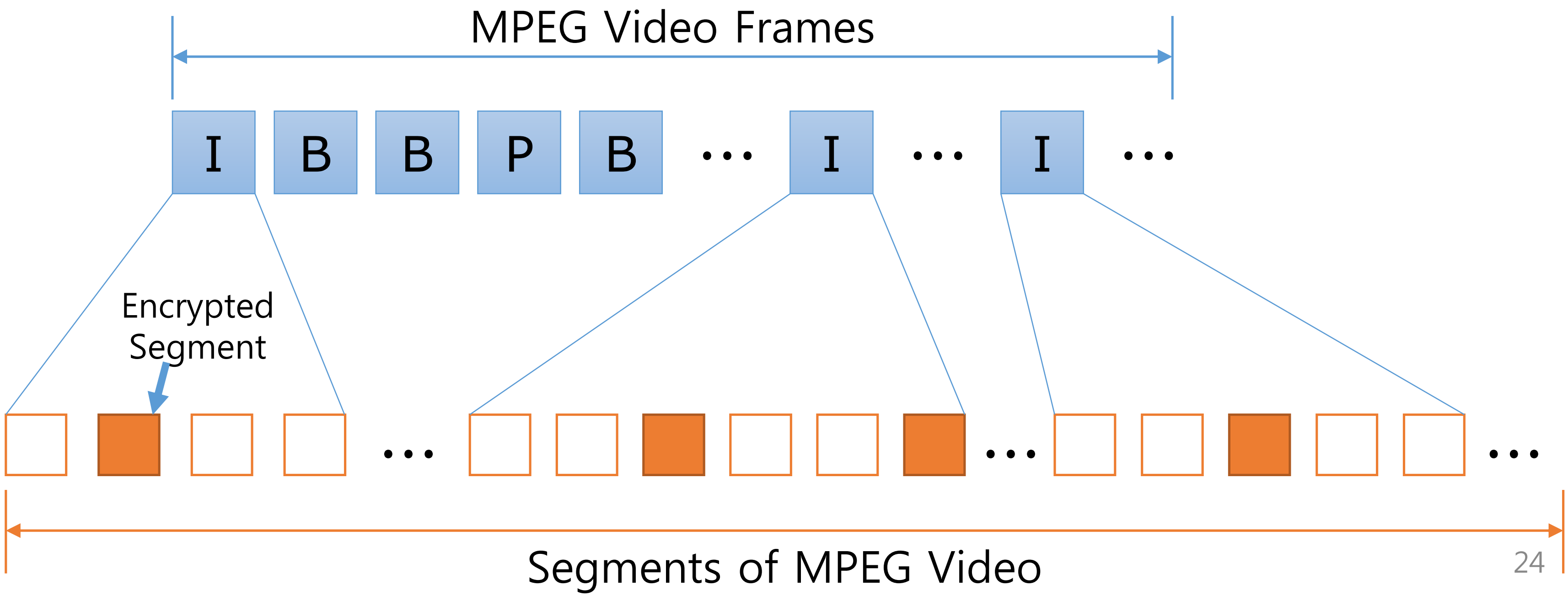
3. Subscriber S downloads packets of both encrypted and unencrypted video, the former of which are decrypted with symmetric keys in round-robin.

Do we need to encrypt all the segments of an I-frame?

- I-frames are larger than other frames in volume.
  - Usually an I-frame consists of multiple segments.
  - Encrypting a subset of segments may foil decoding the entire I-frame by adversary without proper keys.

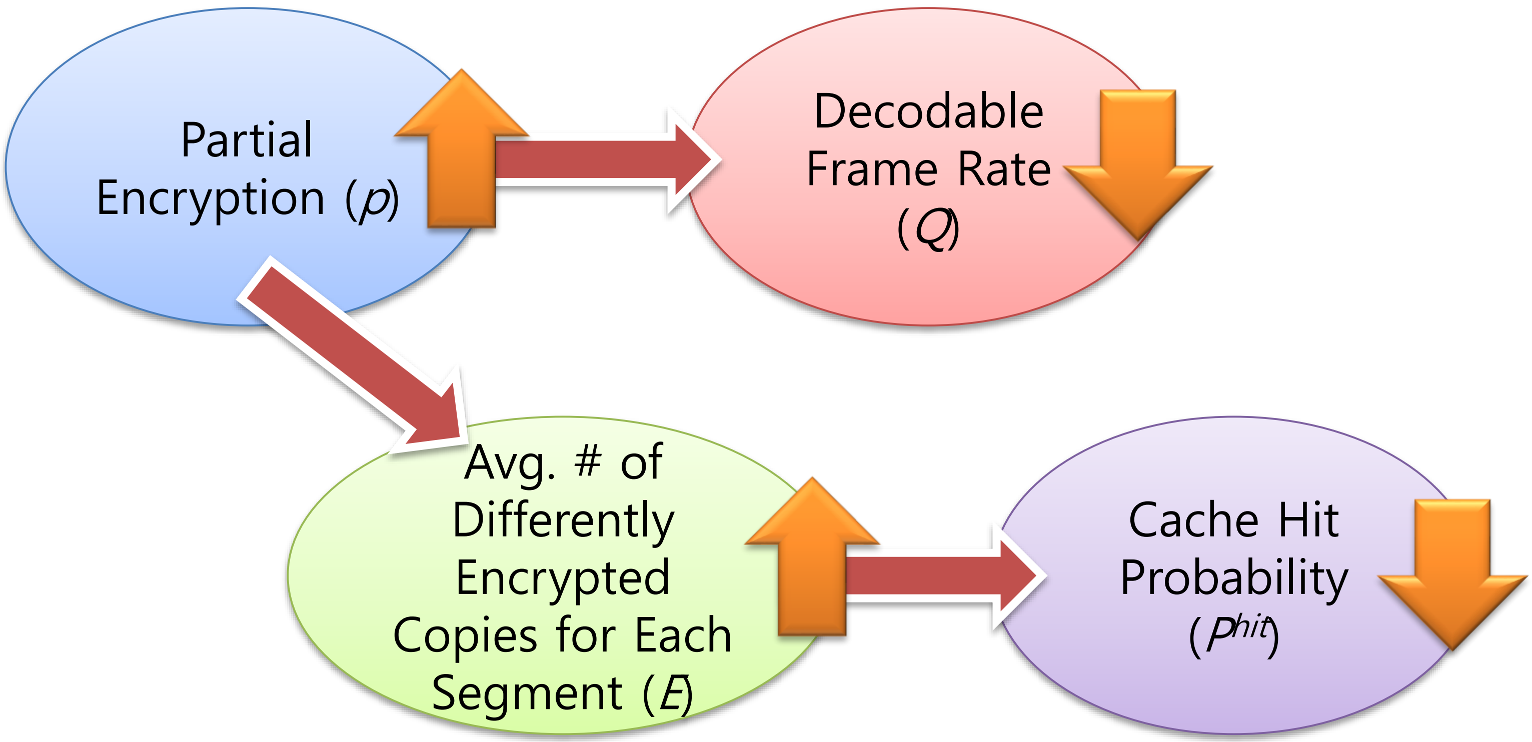
Partial Encryption of I-Frames

- Not all the I-frame segments need to be encrypted.
  - Encrypting a subset of I-frame segments can lower PSNR significantly (of an adversary)



MODELLING AND EVALUATION

How Partial Encryption Affects the Performance?



Modelling Partial Encryption Impact on Decodable Frame Rate

- Decodable Frame Rate Q

Number of decodable frames

Number of decodable I-frames

$$Q = \frac{N_{dec}}{N_{total}} = \frac{N_{dec-I} + N_{dec-P} + N_{dec-B}}{N_{total-I} + N_{total-P} + N_{total-B}}$$

Number of total frames

Number of total I-frames

- Expected number of successfully decodable I-frames
  - p: Encoded segment ratio of I-frame
  - Probability of the I-frame of a GOP to be successfully decoded (C_I: number of segments of an I-frame)

$$S(I) = (1-p)^{C_I}$$

  - Expected number of successfully decodable I-frames

$$N_{dec-I} = S(I) * N_{GOP} = (1-p)^{C_I} * N_{GOP}$$

Number of GOP sequences & there is one I-frame in each GOP

Modelling Partial Encryption Impact on Decodable Frame Rate

- Expected decodable frame rate *Q*

$$
\begin{aligned}
Q &= \frac{N_{dec-I} + N_{dec-P} + N_{dec-B}}{N_{total-I} + N_{total-P} + N_{total-B}} \\
&= \frac{(1-p)^{C_I} \cdot N_{GOP} + (1-p)^{C_I} \cdot N_P \cdot N_{GOP} + \left[(\frac{N}{M}-1) + (1-p)^{C_I}\right] \cdot (1-p)^{C_I} \cdot (M-1) \cdot N_{GOP}}{N_{total-I} + N_{total-P} + N_{total-B}} \\
&= \frac{\left\{1 + N_P + \left[(\frac{N}{M}-1) + (1-p)^{C_I}\right] \cdot (M-1)\right\} \cdot (1-p)^{C_I} \cdot N_{GOP}}{N_{total-I} + N_{total-P} + N_{total-B}} \\
&= \frac{\left\{\frac{N}{M} + \left[(\frac{N}{M}-1) + (1-p)^{C_I}\right] \cdot (M-1)\right\} \cdot (1-p)^{C_I}}{N}.
\end{aligned}
$$

Q is inversely proportional to *p*.

Evaluation of Partial Encryption

- Video Statistics
  - GOP(N=12, M=3)

Video File		Foreman	Akiyo
Total number of frames		300	300
I-frames	Number of Frames	25	25
	Total size of frames (Bytes)	435.643	312.528
P-frames	Number of Frames	75	75
	Total size of frames (Bytes)	245.874	45.859
B-frames	Number of Frames	200	200
	Total size of frames (Bytes)	167.196	24.038
C_I	For 0.5K Packet	34.85144	25.00224
	For 1K Packet	34.85144	25.00224
	For 2K Packet	8.71286	6.25056
	For 4K Packet	4.35643	3.12528

C_I is the mean number of packets of an I-frame, which is used for previous model.

- Evaluation Method
  - Encoder/decoder
    - ffmpeg, libavcodec
  - Making pseudo encrypted file
    - Equal-length segments of I-frame is overwritten with meaningless 0x41 ('A') depending on probability *p*.
  - Quality Metric
    - PSNR

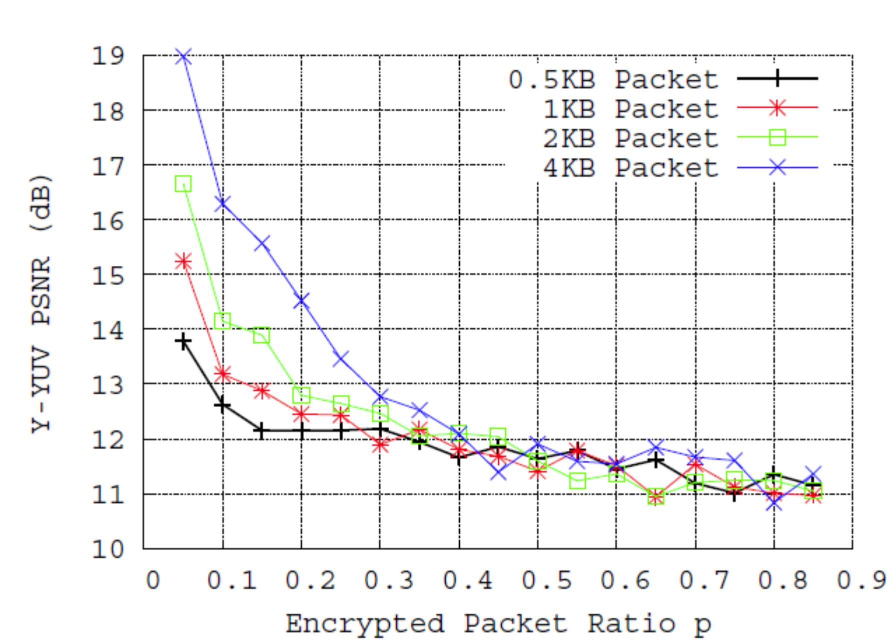
PSNR

- Peak Signal to Noise Ratio (PSNR) is the standard way to measure video fidelity.

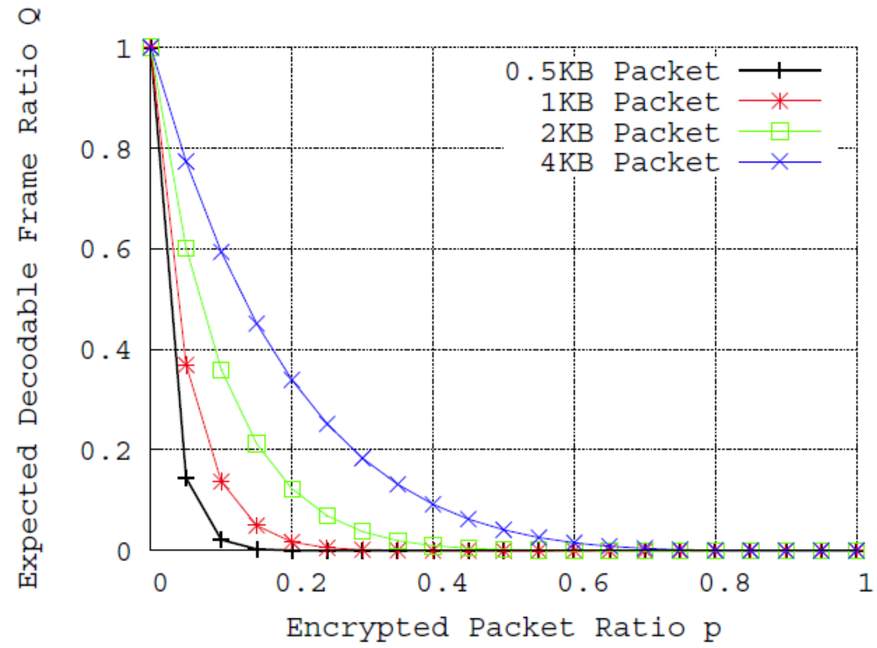
$$PSNR = 10log_{10}(\frac{c^2}{MSE})$$

c is a maximum possible value of a pixel (constant)

- PSNR is measured in decibels (dB).
- Higher PSNR value means better quality.



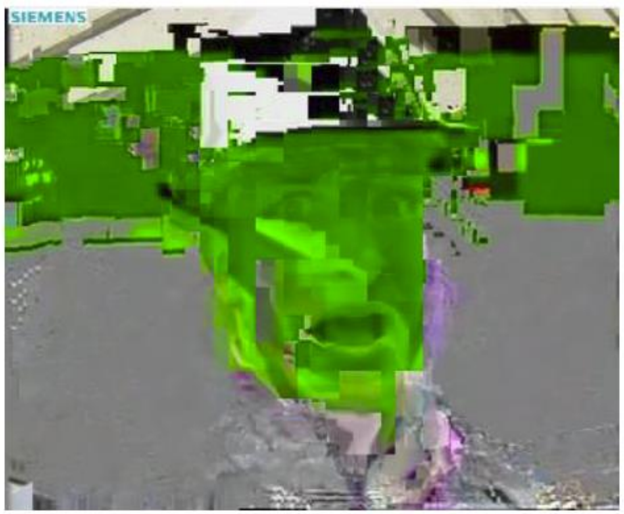
(a) Measured PSNR (Y-YUV) of Foreman CIF, MPEG-4 H.264/AVC, GOP(12, 3)



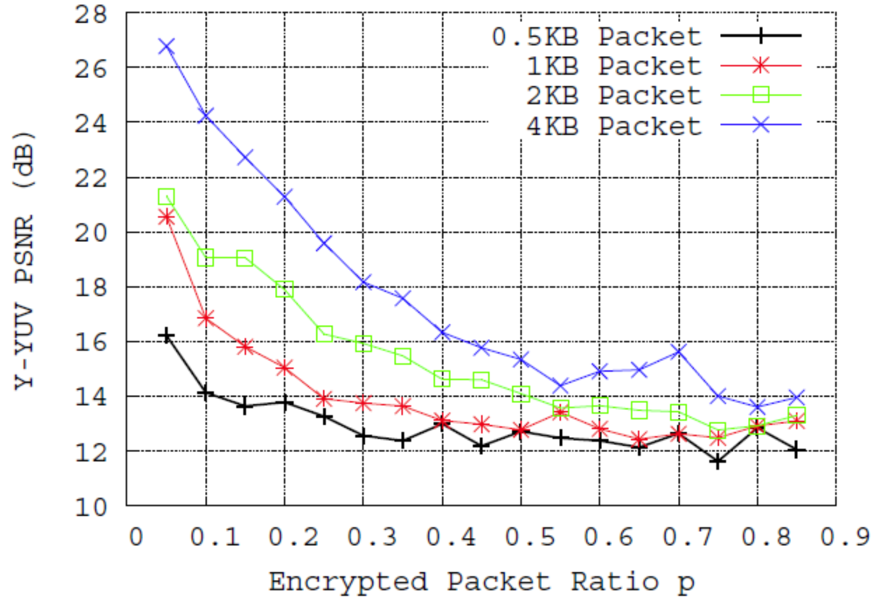
(b) Expected Decodable Frame Ratio Q, GOP(12, 3)

(a) Original

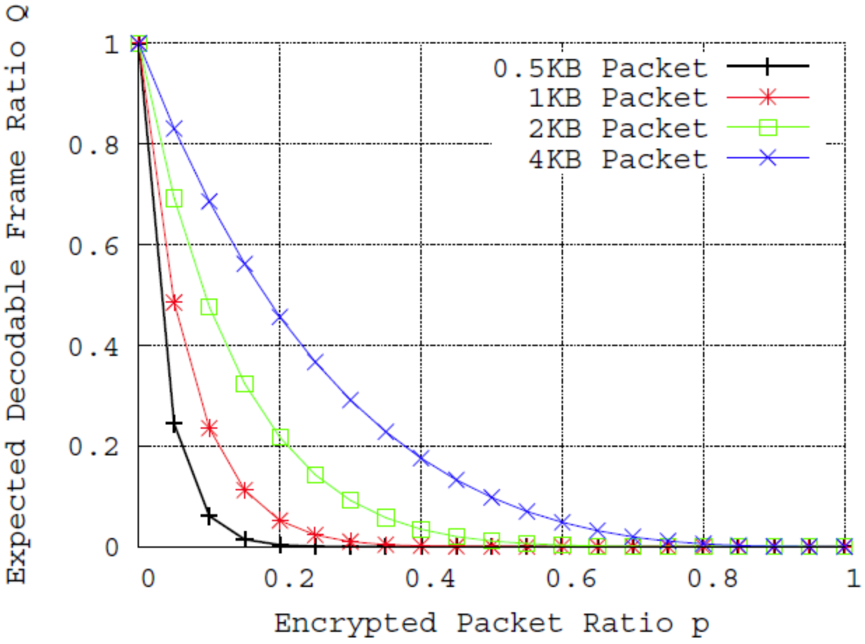
(b) Best PSNR



(c) Worst PSNR



(a) Measured PSNR (Y-YUV) of Akiyo CIF, MPEG-4 H.264/AVC, GOP(12, 3)



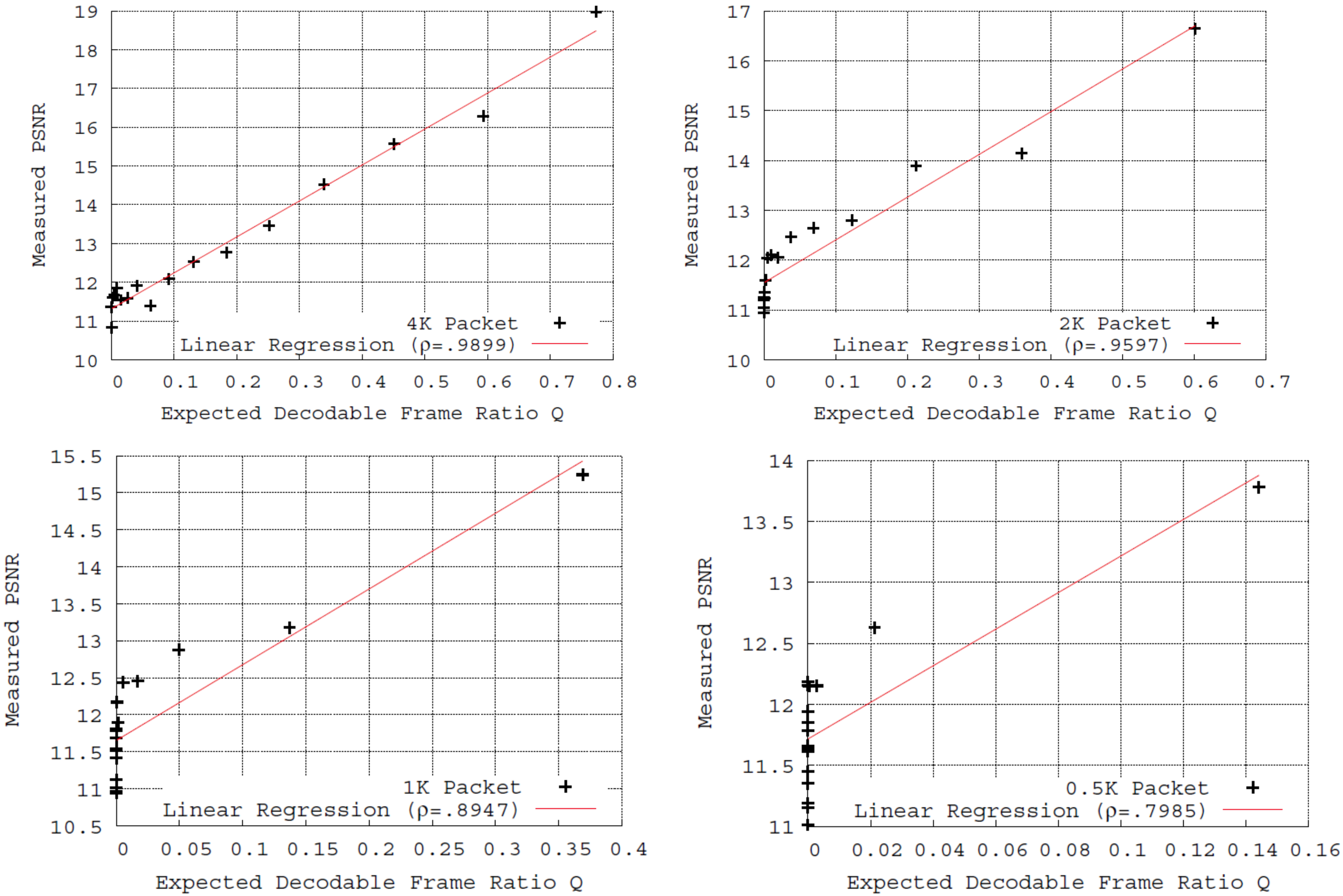
(b) Expected Decodable Frame Ratio Q, GOP(12, 3)

(a) Original

(b) Best PSNR

(c) Worst PSNR

Measured PSNR vs. *Q*



Modelling Cache Hit Probability

- Cache hit probability can be calculated on a single cache with a cache storage of m segments:
  - Hit probability of segment k ($k = 1, \ldots, K$)
    - $P_k^{hit}(m, E) = 1 - \pi_k^{m+1} = 1 - \frac{K' - m}{K'(q_k+1)-1} \prod_{i=1}^{m-1} \left(\frac{K' - i}{K'(q_k+1)-1-i} \right)$

Miss prob. of content request of segment k

Prob. of content request of segment k

  - Hit probability of the whole K' segments
    - $P^{hit}(m, E) = \sum_{i=1}^{K'} q_i P_i^{hit}(m, E)$

P^{hit} decreases since K' is proportional to p.

$K' = K \cdot E$

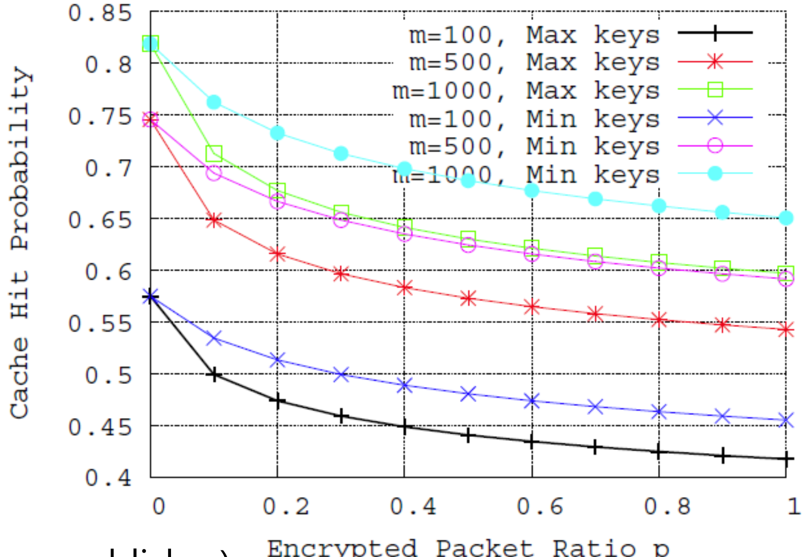
K' is the total number of different segments including the encrypted segments

K is the total number of segments before encryption

E is an average number of differently encrypted segments for a given content

Modelling Cache Hit Probability

- # of Segments
  - Blu Ray Single Layer 25GB → 6.25M of 4KB segments
- Memory capacity (m)
  - Cisco ASR1000 Series Route Processors (RPs)
  - RP1: up to 4GB DRAM → 1M of 4KB segments
- Base values:
  - 6.25K segments (on the network)
  - 1K segments of memory capacity
- Two key distributions
  - Min keys: max overlapping keys
  - Max keys: min overlapping keys
- Other settings
  - S=u=100, s=3, I-frame ratio=0.3



u: # of subscribers (users)
s: # of keys given to a user
S: # of keys in total (managed by a publisher)

Finding Optimal Configurations

- Tradeoff model between the cache hit probability P^{hit} and decodable frame ratio Q
  - Tradeoff function

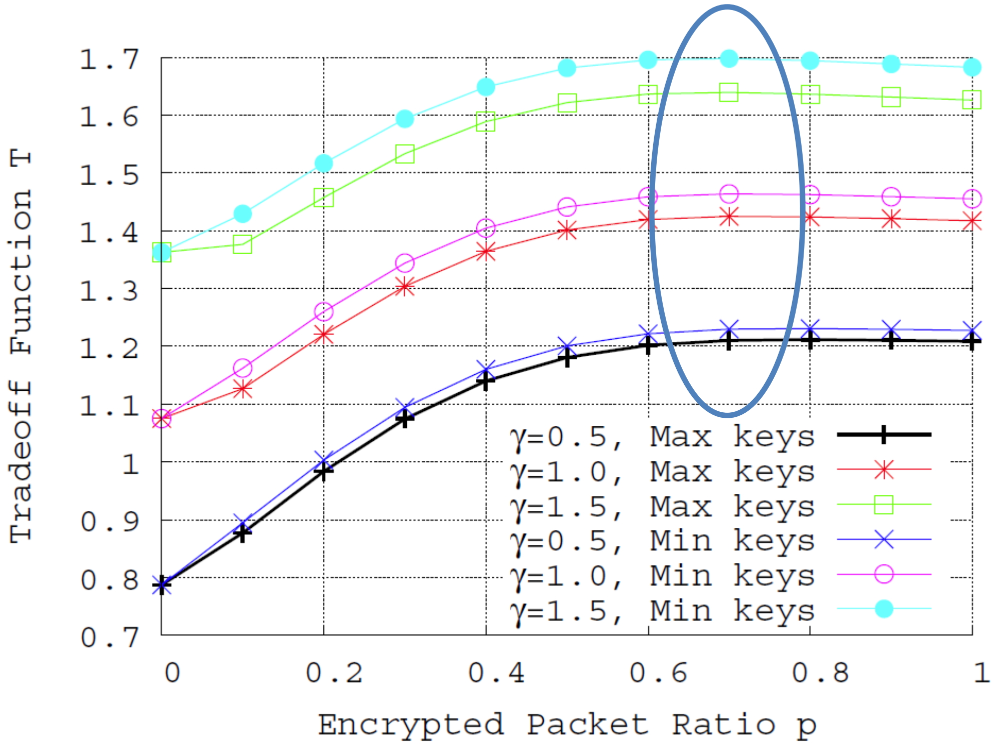
$$T(m, p, s, u, S) = \gamma \cdot P^{hit}(m, p, s, u, S) + \frac{1}{Q(p) + \delta}, \quad \gamma, \delta > 0$$

Scaling parameters

  - Maximum cache hit probability by varying control parameter p

$$\begin{aligned} \max_{p} \quad & T \\ s.t. \quad & 0 \leq Q \leq \epsilon \\ & 0 \leq p \leq 1 \\ & 1 \leq u \leq S^s \\ & Q,\ p \in R,\ u \in Z_+ \end{aligned}$$

Numerical Results



- $\delta = 1.0, S = u = 100, s = 3$, I-frame ratio=0.3, $K = 6250, m = 100, GOP(12,3), C_I = 4.35643$

Conclusion

- Assuming MPEG video streams, we seek to achieve data protection while preserving the advantage of CCN's in-network caching
- We present a CCN protection framework for video streaming services:
  - Key mechanism is the partial encryption
  - Tradeoff between the data protection and caching efficiency in CCN

END OF DOCUMENT