[Samsung PRISM] End Review Report

Selective encryption for H.264/AVC video streams

1. College Professor(s): Prof. Sumit Kumar Pandey (sumit.pandey@iitjammu.ac.in)
2. Students:
    1. Hitesh Choudhary (2022ucs0092@iitjammu.ac.in)
    2. Arjun Verma (2023uma0204@iitjammu.ac.in)
    3. Krishna (2023uee0140@iitjammu.ac.in)
    4. Sahil (2023ucs0109@iitjammu.ac.in)
3. Department: Electrical, Mathematics, Computer Science

Date: 5 Nov 2019

Selective encryption for H.264/AVC Video streams

SAMSUNG PRISM

[R & D of Selective encryption technique for H.264 Video]

- In traditional image and video content protection schemes, the whole content is first compressed. Then, the compressed bit-stream is entirely encrypted using a standard cipher (RSA, AES, IDEA, etc.).
- The specific characteristics of this kind of data (high-transmission rate with limited bandwidth) make standard encryption algorithms inadequate. Another limitation of such systems is altering that the whole bit-stream syntax is altered, which may disable some codec functionalities.
- Selective encryption is a trend in image and video content protection. It consists of encrypting only a subset of the data. The aim of selective encryption is to reduce the amount of data to encrypt while preserving a sufficient level of security. This computation saving is very desirable especially in constrained communications (real-time networking, high-definition delivery, and mobile communications with limited computational power devices, etc.). In addition, selective encryption allows preserving some codec functionalities such as scalability.

Shilpa Ramesh
Chief Engineer
r.shilpa@samsung.com
+91-9901475382

Durgesh M N,
Senior Chief Engineer
durgesh.mn@samsung.com
+91-9916099343

[Reference]

4 Members

6 Months

- https://www.mdpi.com/2073-8994/12/3/332
- https://www.researchgate.net/publication/26593939_Overview_on_selective_encryption_of_image_and_video_Challenges_and_perspectives_EURASIP

[Trainings]

- Proficiency in C/C++.
- Understanding of Cryptography algorithms
- Understanding of basics of video encoding schemes Understanding of H.264 codec
- Concept of Selective encryption for Video data
- Understanding of AES encryption algorithm, FIPS-192, FIPS 140-2 Security Policy
- Literature survey on the existing approaches.

[Output]

- Part 1 - Working prototype to encrypt and decrypt H.264/AVC Video streams
- Part 2 - Paper publication on Selective encryption technique,
  - if any novel or better algorithm has been developed.
  - Or Comparative study of Performance and Usability when AES, ECC and RSA, any other are used as encryption algorithm.

[Timeline]

Kick Off
- Requirement Understanding
- Study required technologies
- Complete Literature survey

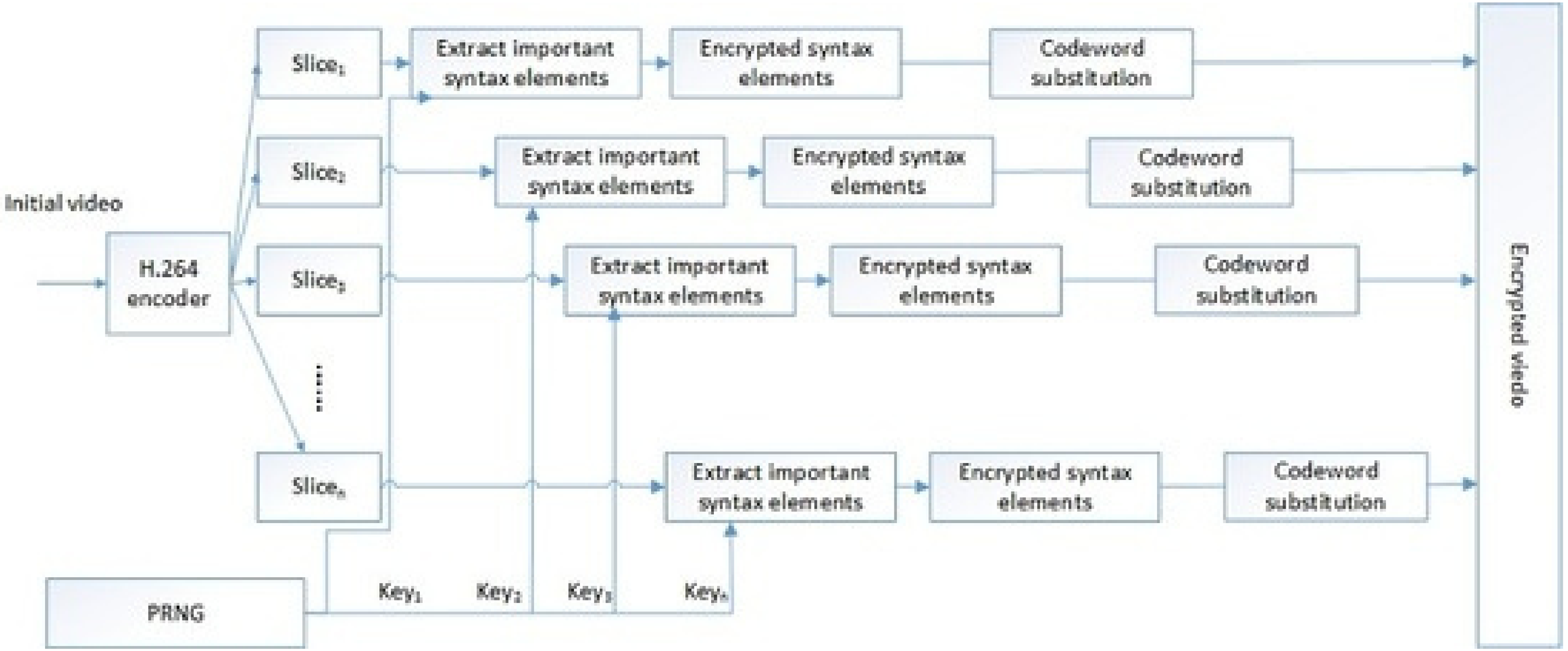
Milestone 1
- Implement prototype to read & write H.264 video codec data from file.
- Implement a module to encrypt and decrypt sample H.264 data using any of the mentioned algorithms.

Milestone 2
- Implement selective encryption algorithm on H.264 codec using prior art study using one of the encryption techniques.

Closure
- Demonstrate the comparative study of performance with other encryption techniques

Approach / Solution

- **Concept Diagram** :
  (Clear detailed schematic / block diagram / flow chart depicting the proposed concept / solution)



Dataset(s) Analysis / Description

- **Dataset Capture / Preparation / Generation** :
  (Discuss the dataset generation process or if downloaded data provide details of what data & from where it was obtained etc… - 2 to 3 bullets only)

  - The dataset consists of H.264-encoded .mp4 video files collected from open-source media repositories.
  - Raw video streams are extracted using FFmpeg into Annex-B formatted .h264 streams along with corresponding .aac audio files.
  - Metadata such as QP (Quantization Parameter) values and NAL types are obtained using FFmpeg's trace_headers bitstream filter for slice-level analysis.

- **Dataset Understanding / Analysis** :
  (Provide 2 to 3 bullets about what is your understanding of the data / opinion about the data)

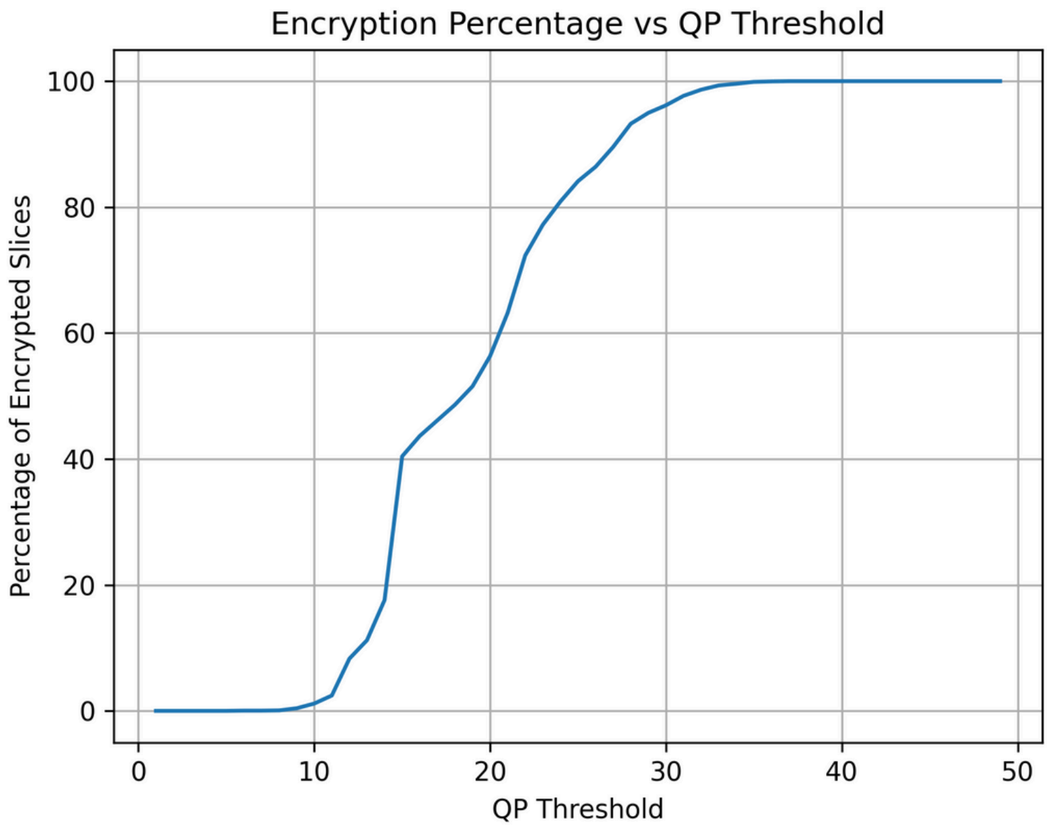
  - Each video slice has a QP value representing compression strength — lower QP implies higher quality and is more valuable for encryption.
  - NAL types (especially 1 and 5) help identify slice types (non-IDR and IDR frames), which are selectively encrypted based on QP threshold.
  - Audio streams are optionally extracted and encrypted, providing dual-stream security for multimedia content.

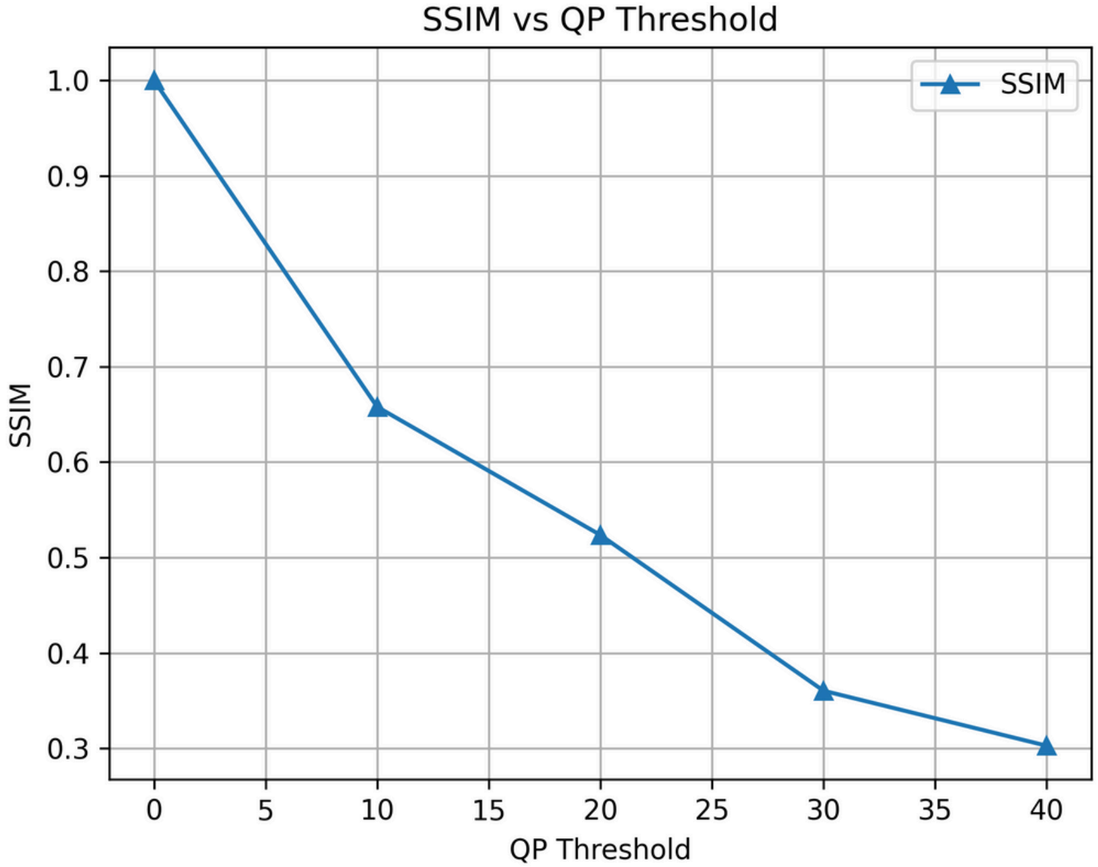
- **Dataset Pre-Processing / Related Challenges (if any)** :
  (List out the challenges you fore see in data handling wrt problem definition – 2 to 3 bullets only)

  - Accurate extraction of NAL units and maintaining decoding order is critical for preserving video playback integrity.
  - Bitstream modification must respect emulation prevention rules to remain decoder-compliant.
  - Handling audio-video synchronization during encryption/decryption and remuxing stages is non-trivial due to stream timing dependencies.

Experimental Results / Simulations / Observations

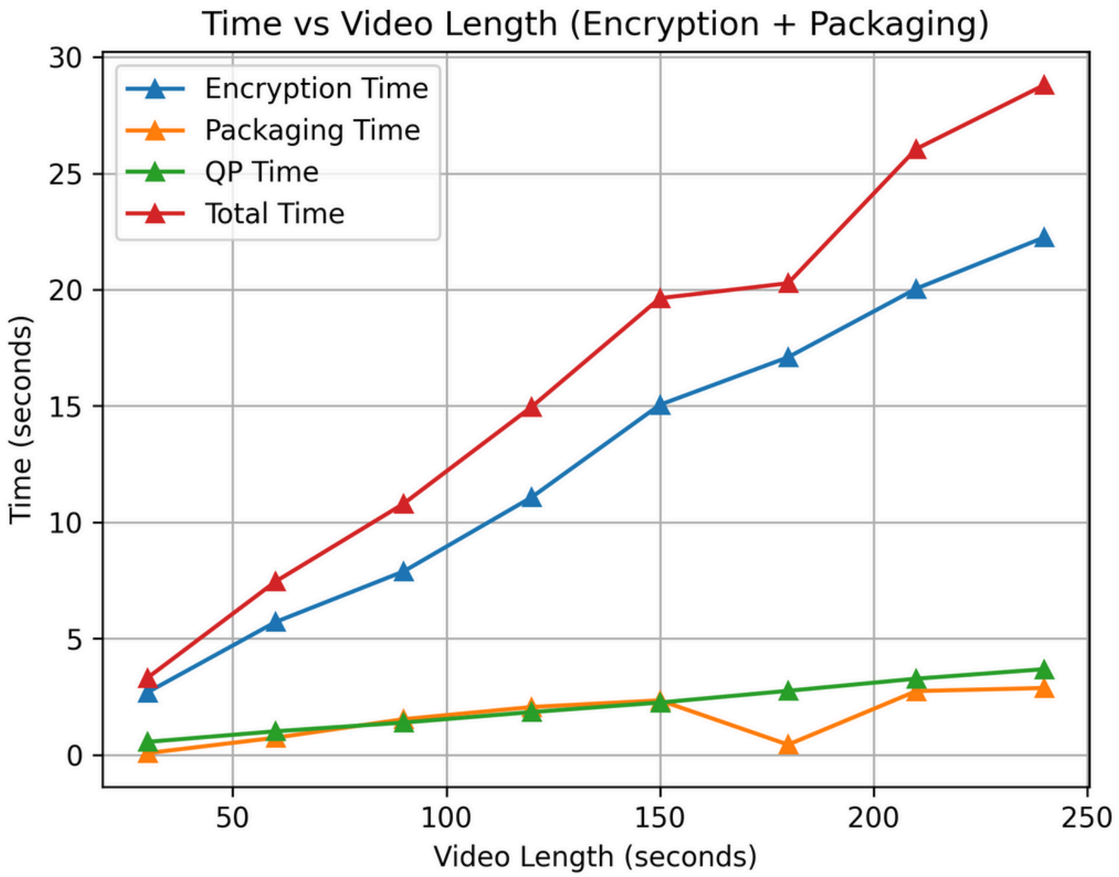
- **Results** :
  (provide numerical data / bar charts / plots / images / videos / tabulated results etc. Use full slide or multiple slides up to max 3 slides to demonstrate the results)

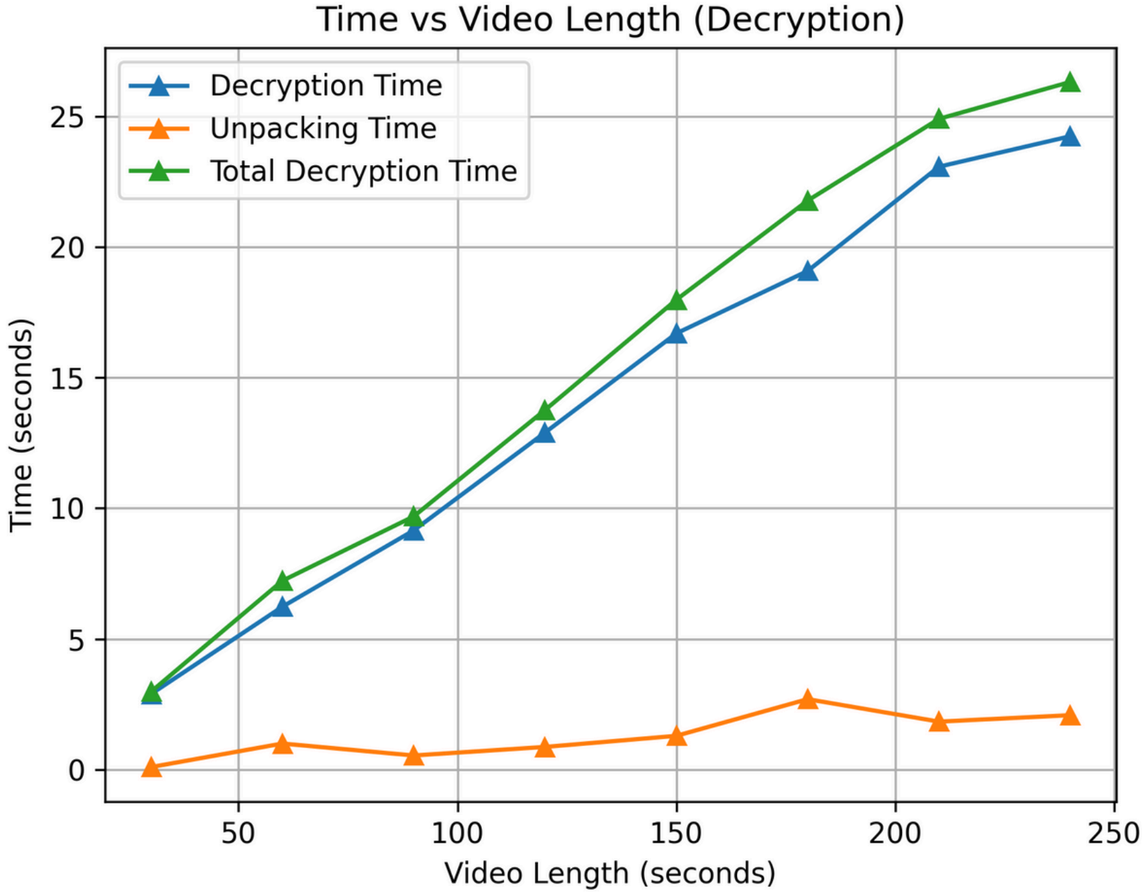




Experimental Results / Simulations / Observations

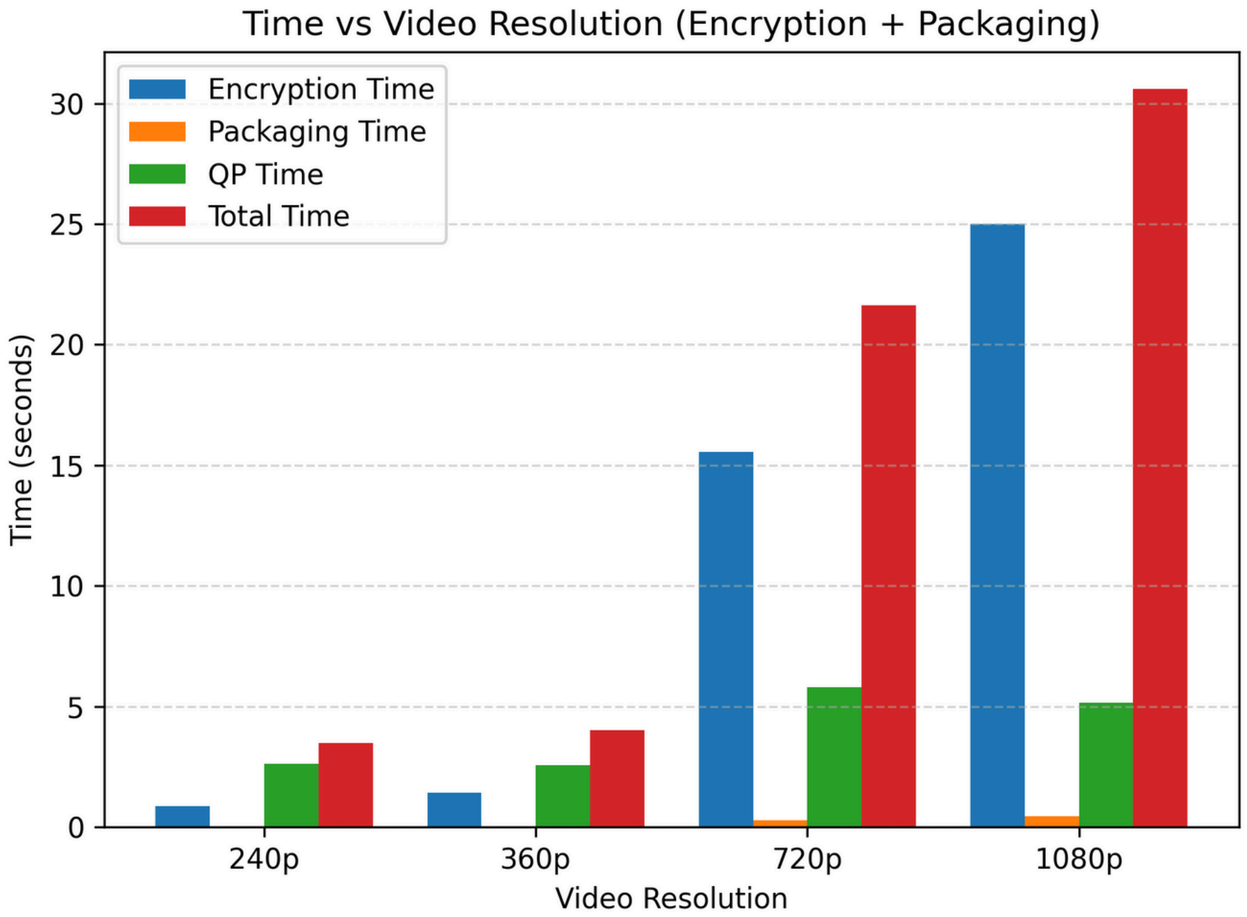
- **Results** :
  (provide numerical data / bar charts / plots / images / videos / tabulated results etc. Use full slide or multiple slides up to max 3 slides to demonstrate the results)

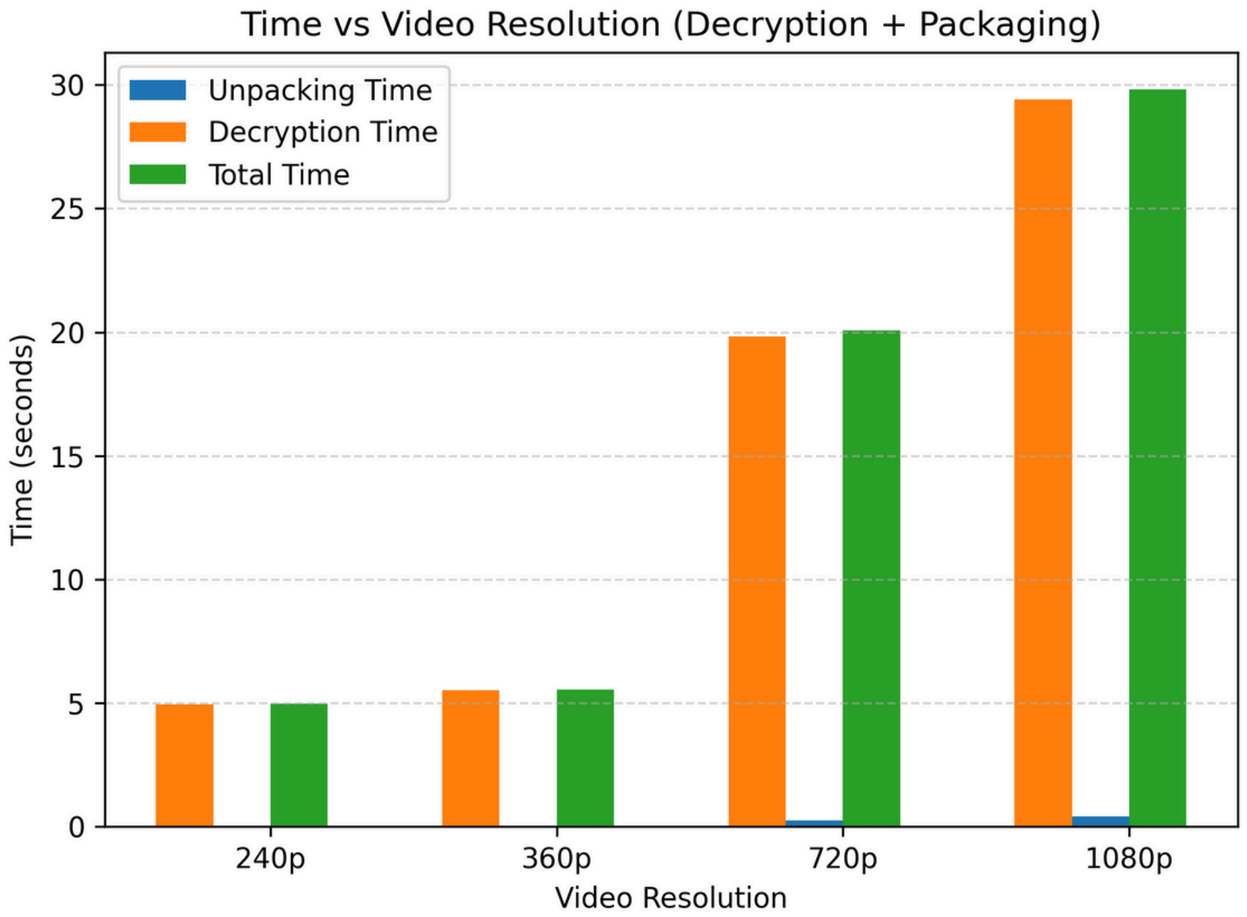




Experimental Results / Simulations / Observations

- **Results** :
  (provide numerical data / bar charts / plots / images / videos / tabulated results etc. Use full slide or multiple slides up to max 3 slides to demonstrate the results)





Experimental Results / Simulations / Observations

- **Major Observations / Conclusions & Challenges :**
  (provide details about your findings, experimental opinion – Use separate slide if necessary)

C++ Limitations

Advanced Scheme 2 required bit-level manipulation and rapid prototyping, which was cumbersome in C++, so we shifted to Python.

Encryption time

Encryption time, especially for longer videos, is very long and requires waiting.

Bitstream Parsing Complexity

Extracting data like residuals and motion vectors needed deep parsing of the encoded H.264 bitstream, unlike QP and macroblocks which are accessible via FFmpeg.

Deliverable

- **Final Deliverables** :
  (Discuss in the form of bullets, what are the next steps to complete the solution, any road blocks / bottlenecks, any support needed from SRIB)

- Implemented two encryption schemes for video security
- **Scheme 1:** Encrypts frames using correlation-based encryption and column shuffling
- **Scheme 2:** Selectively encrypts H.264 video slices (NAL units of type 1 and 5) based on macroblock QP values using AES in CTR mode

- **KPIs delivered/Expectations Met**:
  (Planned Expectations shared in Work-let vs Delivered Results)

Selective Encryption

Targeted encryption of video macroblocks based on QP, preserving compression efficiency.

Successful Decryption

Ensured accurate frame recovery with no visible artifacts post-decryption.

Audio Encryption

Fully encrypted AAC stream using AES, maintaining sync and integrity

Work-let Closure Details

- **Code Upload details:**

Items	Details
KLOC (Number OF Lines of codes in 000's)	1.5 KLOC
Model and Algorithm details	Implemented two encryption schemes: **Scheme 1**: Encryption using correlation-based encryption and column shuffling. **Scheme 2**: Selective encryption of H.264 NAL units (types 1 & 5) based on macroblock QP values using AES-CTR.
Is Mid review, end review report uploaded on Git ?	Yes
Link for Git	https://github.ecodesamsung.com/SRIB-PRISM/IITJAMMU_24VI59IITJ_Selective_encryption_for_H_264_AVC_Video_streams

- **Data details (if applicable):**

Items	Data folder 1	Data folder 2	Data Folder 3……..
Name & Type of Data (Audio/Image/Video)	Sample videos (MP4, H.264 encoded)		
Number of data points	2 test videos		
Source of Data (self collected, Scrapped, available on open source)	Self-collected		
Google drive link/ git link to access data	https://github.ecodesamsung.com/SRIB-PRISM/IITJAMMU_24VI59IITJ_Selective_encryption_for_H_264_AVC_Video_streams/tree/main/video		

Thank you