Premium Content Protection Using HDCP

By

Pruthviben P Patel

17MCEI10

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

INSTITUTE OF TECHNOLOGY, NIRMA UNIVERSITY

AHMEDABAD

MAY,2019

Premium Content Protection Using HDCP

Major Project

Submitted in partial fulfillment of the requirements
for the degree of
Master of Technology in Computer Science & Engineering
(Information & Network Security)

By

Pruthviben P Patel

17MCEI10

Guided By

College Guide:Prof. Kruti Lavingia

Project Manager: Satya Tripathi

Project Mentor: Sumit Kishore

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

INSTITUTE OF TECHNOLOGY , NIRMA UNIVERSITY

AHMEDABAD

MAY, 2019

Certificate

This is to certify that the Major Project entitled Premium Content Protection Using HDCP submitted by Pruthviben P Patel(17MCEI10), towards the partial fulfillment of the requirements for the degree of M.Tech in Computer Science and Engineering(Information and Network Security) of Nirma University, Ahmedabad is the record of work carried out by her under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this Project, to the best of my knowledge, haven't been submitted to any other university or institution for the award of any degree or diploma.

Prof. Kruti Lavingia
Guide & Assistant Professor,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad

Dr. Sharada Valiveti
Coordinator M.Tech - CE (INS),
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad

Dr.Madhuri Bhavsar
Professor and Head,
f CSE Department,
Institute of Technology,
Nirma University, Ahmedabad

Dr. Alka Mahajan
Director,
Institute of Technology,
Nirma University,
Ahmedabad

Statement of Originality

I, **Pruthvi Patel(17MCEI10)** give undertaking that the Major Project entitled **"Premium Content Protection using HDCP"** submitted by me, towards the fulfillment of the requirements for the degree of Master of Technology in **Computer Science & Engineering(Information & Network Security)** of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made.It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action

---

Signature of Student

Date:

Place:

Endorsed By

Prof.Kruti Lavingia

(Signature of Guide)

Acknowledgements

It gives me immense pleasure in expressing thanks and profound gratitude to **Prof. Kruti Lavingia**, Assistant Professor, Computer Engineering Department, Institute of Technology, Nirma University, Ahmedabad for her valuable guidance and continual encouragement throughout this work. The appreciation and continual support she has imparted has been a great motivation to me in reaching a higher goal. Her guidance has triggered and nourished my intellectual maturity that I will benefit from, for a long time to come. I would like to express my deepest appreciation to all those who provided me the possibility to complete this report. A special gratitude I give to my final year project manager, **Mr. Satya Tripathi** ,and My mentor **Sumit Kishore**, whose contribution in stimulating suggestions and encouragement, helped me to coordinate my project study. It gives me an immense pleasure to thank **Dr.Sharda Valiveti**, Hon'ble Coordinator M.Tech - CSE (INS) Institute of Technology, Nirma University, Ahmedabad for her kind support and providing basic infrastructure and a healthy research environment. It gives me an immense pleasure to thank **Dr.Madhuri Bhavsar**, Hon'ble Head of Computer Engineering Department, Institute of Technology, Nirma University, Ahmedabad for his kind support and providing basic infrastructure and healthy research environment. A special thank you is expressed wholeheartedly to **Dr.Alka Mahajan**, Hon'ble Director, Institute of Technology, Nirma University, Ahmedabad for the unmentionable motivation she has extended throughout course of this work. I would also thank the Institution, all faculty members of Computer Engineering Department, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

- Pruthviben P Patel

17MCEI10

Abstract

Digital media is quick changing into the popular consumer amusement choice.The use of Whether inside of home or outside, the number of Premium High Definition media in the form of Audio and Video as well as the variety of devices which is used for distribute this kind of content is growing exponentially. Previously the device which used to be stand-alone are currently become networked and repurposed because now consumers trends changed.They are using storage drives and gaming consoles to serve content and storing it; whereas tablets and phones are augmenting TV screens making multi-screen capabilities a required aspect. For improving the user experience we have standards such as HDCP for ensurement of the interoperability of various devices and to provide protection to the content which is purchased and used up. We need to provide protection to the content which is purchased so that the unauthorized person unable to access it. So for that we use HDCP(High-bandwidth Digital Content Protection) Protocol.

The Project "Premium Content Protection Using HDCP" includes the study of HDCP Protocol and OPM Protocol which enforce content protection mechanisms on the video output.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the entertainment industry the Piracy is the ever-growing challenge.The Operators, Device Manufacturers and content providers who wants to raise their sales by providing premium content to the user on the Internet connected devices such as gaming consoles,tablets, laptops,mobile phones, set top boxes (STBs) and smart TVs. In spite of industry endeavors to prevent piracy, premium content theft proceeds to be common, and comes about in billions of dollars of misplaced income each year. To date content owners have permitted their substance to be conveyed in this threatening environment using software-only solutions for security of content. But as the quality of OTT(over-the-top) content improves and approaches the quality of Blue Ray premium video,studios,TV,movie theaters,DVD and are presently forcing stronger security prerequisites as a condition of content licensing agreements.

High-bandwidth Digital Content Protection (HDCP) is a the technique for digital copy protection developed by INTEL Corporation which protects digital entertainment content such as pay-per-view television,high-definition movies or music on home and personal networks including various devices such as gaming devices, PCs, smartphones and tablets. Digital Content Protection LLC (DCP) is the License Provider to the device manufacturer. HDCP Devices are:

1. **Sources**: Sends the Content.e.g. STB, DVD player, PC etc.

2. **Sink**:Renders the Content.e.g. TV, Projector etc.

3. **Repeater**:Accepts content, Decrypts it, re-encrypt and re transmit.e.g. AV Receivers

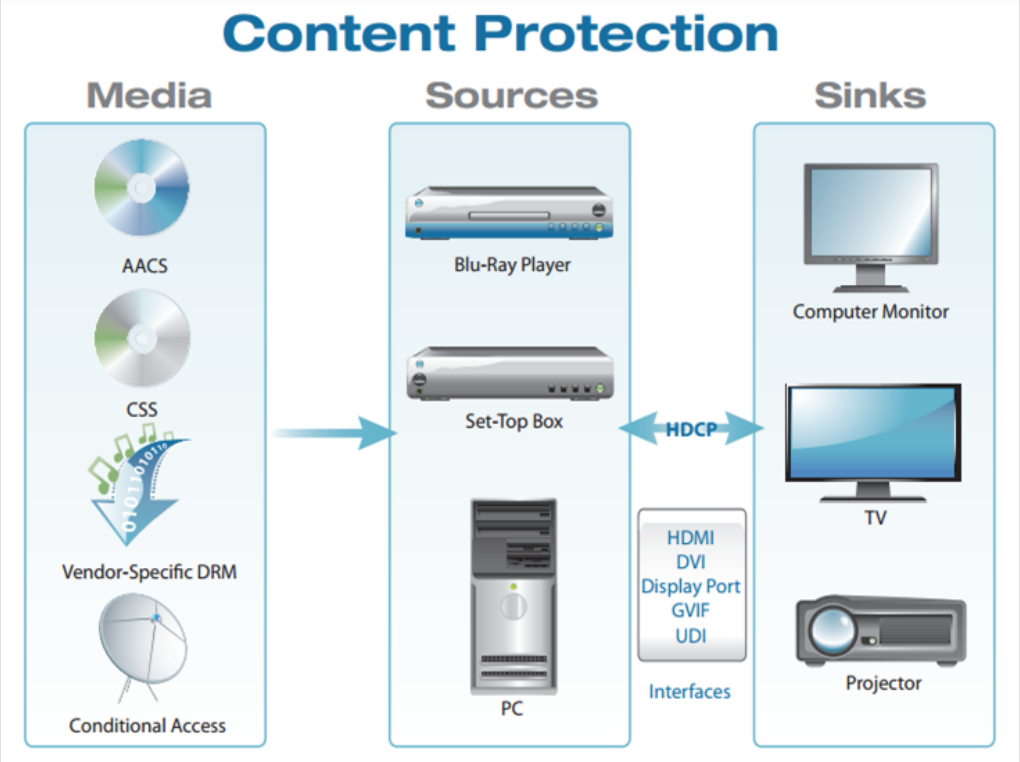

Figure 1.1: Content Protection

There are two Stream Type:

1. **Type 1**: Can be played only on HDCP 2.2 device.Ultra Premium 4K contents.

2. **Type 0**:Can be played on any HDCP protected link.Premium content like 1080 BlueRay.

Protection at different level of content distribution HDCP protocol comes in the picture at the last stage of content distribution process as shown in 1.2.
When content travel from setup-box,personal computer over digital interface to display at that time HDCP is the way to protect content.

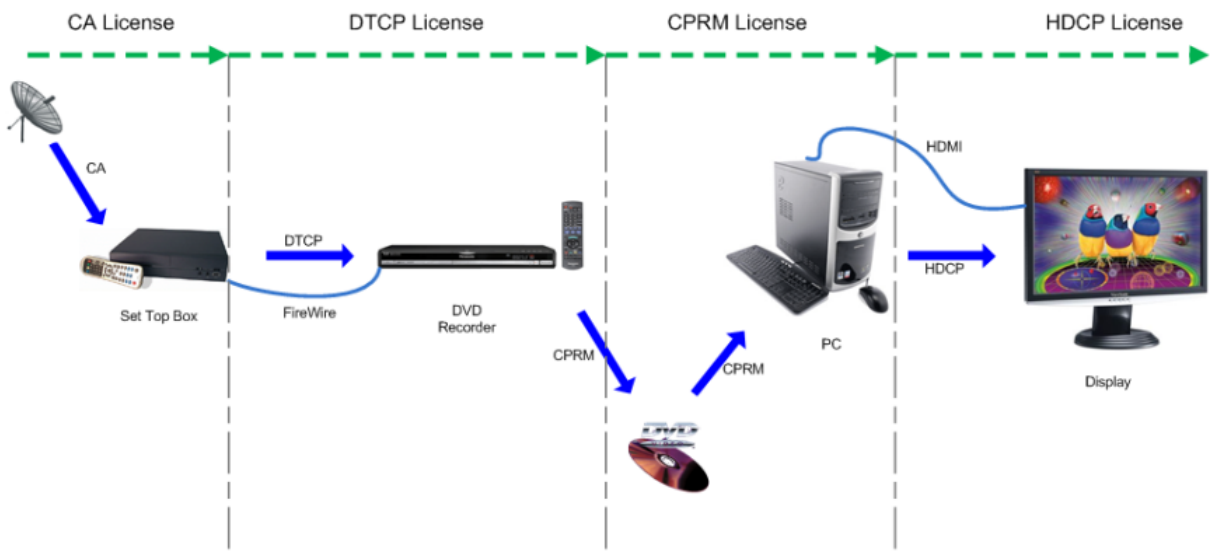

Figure 1.2: Protection at different level of content distribution

Chapter 2

Literature Survey

HDCP(High-bandwidth Digital Content Protection) protocol is developed for protection of Audio and Video Content transmission between an HDCP Transmitter and Reciever on it's HDCP Protected Interface Ports.
HDCP System has three components:[1]

1. HDCP Transmitter(Source)
2. HDCP Repeater
3. HDCP Reciever(Sink)

The HDCP Protocol allows to connect 4 levels of HDCP Repeaters and total 32 HDCP Devices, along with HDCP Repeaters on HDCP protected Interface port. There are main three process of HDCP Protocol. Each process has it's significance in the Setup. First Process is the authentication protocol, In which the HDCP Transmitter checks the license of the Receiver and will check whether it is authorize or not for receiving the Premium Content. The Implementation of this first step is done between the HDCP Transmitter and its respective HDCP Receiver. If the HDCP Receiver is authorize then HDCP Content will be encrypted and then will transmitted between the two devices based on secrets key sharing done in the First Step. This prevents utilization of the premium content by eavesdropping devices or unauthorize

device. There are chances of compromisation of legitimate devices.So for preventing unauthorized use of the Content, renewability mechanism implemented on Transmitter side which will recognize such compromised devices and prohibits the sending of Premium Content.[1]

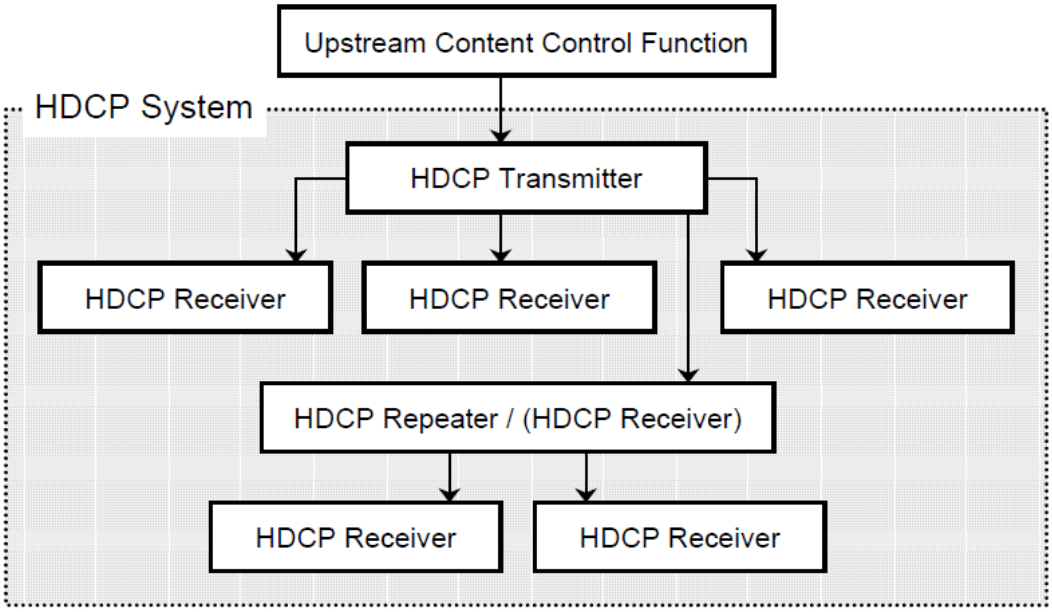

Figure 2.1: HDCP System.

2.1 Three Stages of HDCP

There are three main stages of HDCP:

1. Authentication:Transmitting device verify that the receiver is authentic user or not.

2. Encryption: After the authentication data will be encrypted before sending it over interfaces for avoiding the attacks such as man in middle and eavesdropping.

3. Key Revocation: It is for preventing the device that is compromised and cloned.

2.1.1 Authentication

The HDCP Authentication Process is done between HDCP Transmitter and HDCP Receiver to verify that the reciever is authorized to recieve the content. HDCP Authentication Process is contains of below given steps.

Authentication and Key Exchange (AKE)

In this State, HDCP Transmitter will verify The HDCP Receiver's public key certificate. A Master Key km is exchanged. There is two scenario of AKE: Without Stored k_m and With stored k_m. In without store k_m the following task will take place.[1]

1. HDCP Transmitter will Initiate authentication by sending 64-bit AKE_INIT pseudo-random number(r_{tx}) and TxCaps Parameter.

2. Receiver will send AKE_SEND_CERT to transmitter within the 100ms after receiving AKE_INIT.

3. Transmitter will verify signature on the certificate using $kpub_{dcp}$ after that transmitter will generate k_m and encrypt the k_m using $kpub_{rx}$. Transmitter will send (AKE_No_Stored_km) encrypted k_m to receiver.Also Transmitter will verify SRM Integrity and Perform Revocation Check in 1 second after sending AKE_No_Stored_km.

4. Receiver will decrypt k_m with $kpriv_{rx}$ then receiver will compute $H' = HMAC-SHA256(r_{tx} \ \|\text{RxCaps}\|\text{TxCaps,k}_d)$[6].Receiver will send AKE_SEND_H_Prime within 1 second.

5. Transmitter will Compute H and will verify H==H'.

6. Receiver will compute $E_{kh}(k_m)$ and send AKE_Send_Paring_Info with 200ms after sending AKE_SEND_H_prime

7. Transmitter will store k_m,m,$E_{kh}(k_m)$ and Receiver ID.

The Process of AKE when k_m is not stored is shown in Figure 2.2

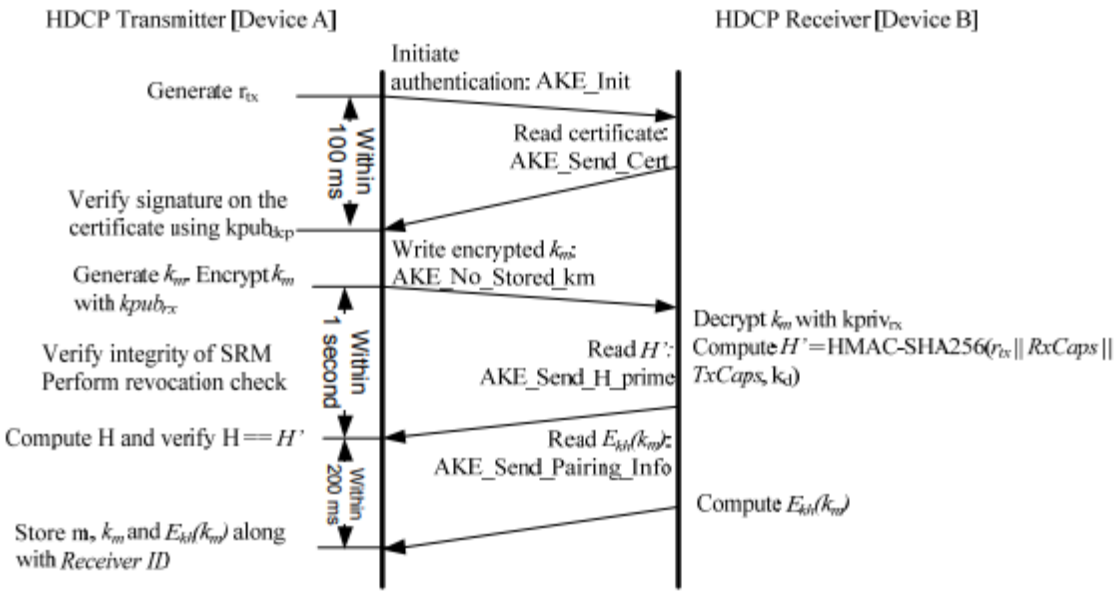

Figure 2.2: AKE without stored k_m

In with stored k_m the following steps will take place.

1. HDCP Transmitter will generate r_{tx} and sends AKE_INIT to the receiver and Initiate Authentication.

2. Receiver will send the AKE_SEND_CERT to transmitter within the 100ms after receiving AKE_INIT.

3. Transmitter will fetch stored $E_{kh}(k_m)$ and m respective with Receiver ID and send m and (AKE_Stored_km)$E_{kh}(k_m)$ to HDCP receiver.Also Transmitter will start to verify SRM Integrity and Perform Revocation Check after sending AKE_No_Stored_km.

4. Receiver will decrypt $E_{kh}(k_m)$ for obtain k_m and computes H'= HMAC-SHA256(r_{tx} $\|RxCaps\|TxCaps$,k_d).Receiver will send AKE_SEND_H_prime within 200ms.

5. Transmitter will Compute H and will verify H==H'. The Process of AKE when k_m is stored is shown in Figure 2.3

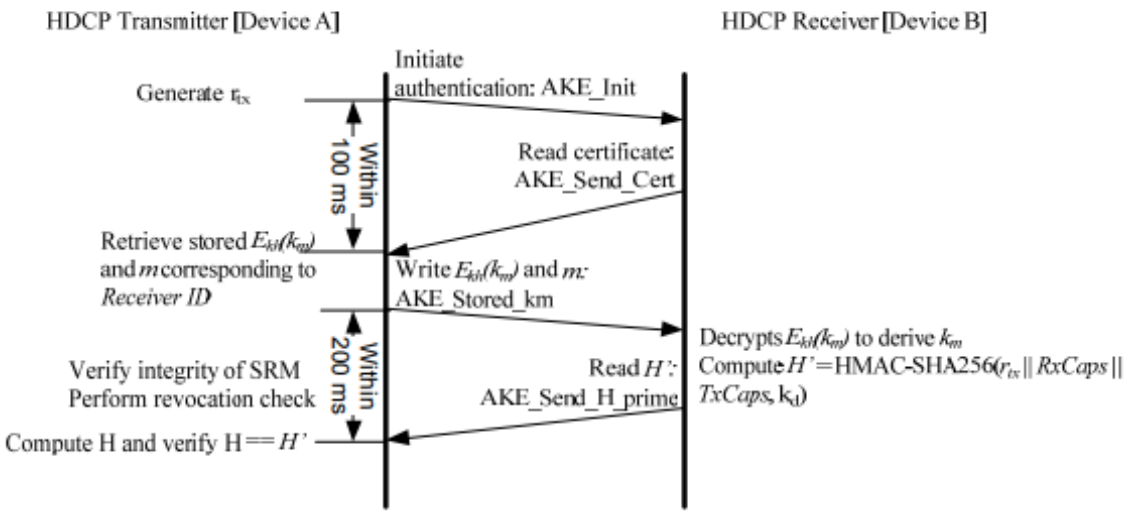

Figure 2.3: AKE with stored k_m

Pairing Paring is necessary for AKE process speed up. It Should be implemented parallel with AKE int between Transmitter and Receiver. When Transmitter send AKE_No_Stored_km message, it is an sign to the receiver that km corresponding to the receiver is not stored with the transmitter.In this scenario Receiver will do following task:[1]

1. Computation of H'
2. Computation of kh[128 bit] = SHA-256($kpriv_{rx}$)[127:0].[5]
3. Encryption of km with kh for Generation of Ekh(km) of length of 128 Bit by using AES as illustrated in Figure[3] 2.4
4. Generation of the AKE_Send_Pairing_Info message consits of Ekh(km) of 128 Bit available to the transmitter for reading. This AKE_Send_Pairing_Info must be accessible to the transmitter for reading in the timespan of 200 ms from the time the transmitter starts reading the AKE_Send_H_prime message sent by the HDCP Receiver.

If the AKE_Send_Pairing_Info message is not received in 200ms to transmitter for

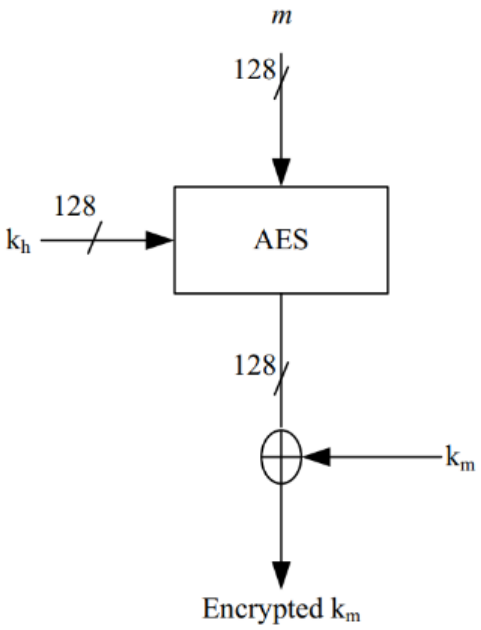

Figure 2.4: $E_{kh}(k_m)$ Computation

reading,then authentication will fail.Transmitter terminate the authentication process. HDCP Transmitter may persistently store m ($(r_t x \| r_r x)$), k_m ,$E_{kh}(k_m)$ and Receiver ID while reading AKE_Send_Pairing_Info.

Locality Check

This step is performed after successful finishing of AKE and Pairing steps. The HDCP Transmitter implement locality on the content by compelling the Round Trip Time (RTT) between the pair of messages.It should not be exceed than 20 ms. In the Process of Locality Check following steps will take place.[1]

1. Transmitter will generate r_n and then will set watchdog timer and initiate Locality Check by sending LC_Init to receiver.

2. Receiver will compute L'=HMAC-SHA256(r_n,k_d XOR r_{rx} and will send LC_Send_L_Prime within 200ms.

3. Transmitter will compute L=HMAC-SHA256(r_n,k_d XOR r_{rx} and verify L=L'.

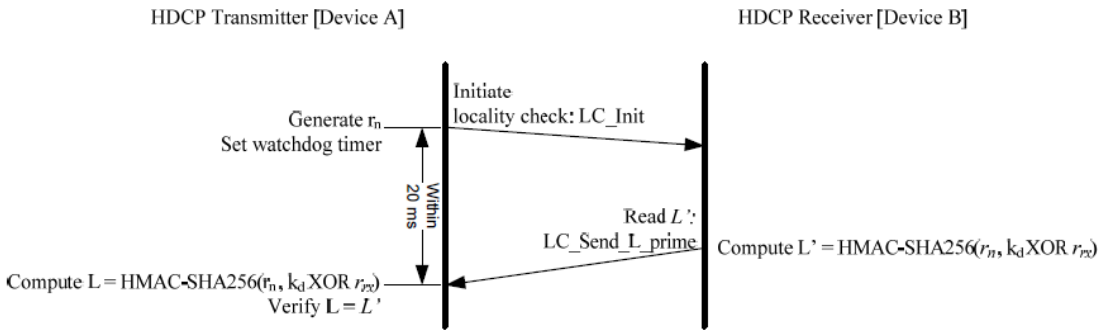

Figure 2.5: Locality Check

Transmitter will reattempt of Locality Check for maximum 1023 time and 1024 time in total with the sending of LC_init consits of new r_n.if it is failed because of mismatch in L and L' or watchdog timer expiration[1]

Session Key Exchange (SKE)

After the Successfully Completion of AKE and Locality Check steps confirm that HDCP Receiver is authorized to receive HDCP Content.Session Key k_s will be exchanged by HDCP Transmitter with the HDCP Receiver.
Transmitter will do following tasks:[1]

1. HDCP Transmitter will generate pseudo random number k_s[128 Bit] and r_{iv}[64 Bit]
2. Perform Key Derivation to generate or derivation of d_{key2} when ctr=2.
3. Computation of $E_{dkey}(k_s)$=k_s XOR (dkey2 XOR r_{rx}), Here r_{rx} is XORed with dkey2 64-bit least significant.
4. Sends AKE_Send_Eks message contsits of $E_{dkey}(k_s)$ and r_{iv}.

Receiver will do following tasks after receiving AKE_Send_Eks:

1. Perform Key Derivation to generate or derivation of d_{key2} when ctr=2.
2. Computation of (k_s)=$E_{dkey}(k_s)$ XOR (dkey2 XOR r_{rx}).

Authentication with Repeaters

This Step is performed with only HDCP Repeaters by the HDCP Transmitter. In this step, the repeater collects information of downstream toplogy and sends it to the upstream HDCP Transmitter. This steps will be initiated by transmitter after SKE and when Repeater Bit is set to 1.Purpose of this step is to transmit upstream propagation of topology information and downstream propagation of content stream management information.

1. **Upstream Propagation of Topology Information:** In This step Repeater will combine all downstream Receiver ID list attached to it.This steps is performed immediately after SKE or when topology changes because of connection or disconnection of device.The list is consists of contiguous set of bytes,with each Receiver ID of 5 bytes in Big-Endian Order. After assembling of the list of Receiver ID,Repeater will compute verification value V' of 256 bit.HDCP Transmitter also compute V. V and V' computation will be done like below: V or V'=HMAC-SHA256(Receiver ID List $\|RxInfo\|seq\_num\_V$,k_d). When the Receiver ID List,V',DEPTH_DEVICE_COUNT,HDCP2_0_REPEATER_DOWNSTREAM and HDCP1_DEVICE_DOWNSTREAM are available,the HDCP Repeater will make the RepeaterAuth_Send_ReceiverID_List message and sends upstream transmitter.After completio of this steps READY status is asserted in RxStatus Register.After this it will sets the Message_Size field equal to the size of RepeaterAuth_Send_ReceiverID_List instantly after READY Status Set.Authentication of the HDCP Repeater will fail if READY status asserted is not delivered within a 3 seconds after sending SKE_Send_Eks message by the HDCP Transmitterss. The HDCP Repeater starts seq_num_V to 0 at the beginning of the HDCP Session i.e. after AKE_Init is received. seq_num_V is increment by one after the transmission of each RepeaterAuth_Send_ReceiverID_List message. seq_num_V should not be reused during an HDCP Session for the computation of V (or V'). HDCP Transmitter should detect the roll-over if

the seq_num_V rolls over in the RepeaterAuth_Send_ReceiverID_List read from the HDCP Repeater.If HDCP Encryption is enabled then Transmitter should disable it,and again starts the authentication by the transmission of a new AKE_Init message. When the HDCP Repeater receives HDCP2_0_REPEATER-_DOWNSTREAM or HDCP1_DEVICE_DOWNS-TREAM bits that are set from a downstream HDCP Repeater, it should provide this information to the HDCP upstream Transmitter by setting the respective bits in the Auth_Send_Receiver-ID_List message. If HDCP2_0_-REPEATER_DOWNST-REAM or HDCP1_DEVI-CE_DOWNSTREAM bit is set, the Upstream Content Control Function will notice the most upstream HDCP Transmitter for aborting the transmission of certain HDCP encrypted Type 1 Content Streams. The most upstream HDCP Transmitter must be ready to process the request and instantly stop the Specific Content Streams transmission as noticed given by the Upstream Content Control Function. As and When the HDCP Transmitter will verify the integrity of Receiver ID list by computing V and comparing the most significant 128-bits of V and V' when it reads the RepeaterAuth_Send_ReceiverID_List message. If the V is not equal to V'then authentication fails.The authentication process is aborted.HDCP Encryption is disabled.[1]

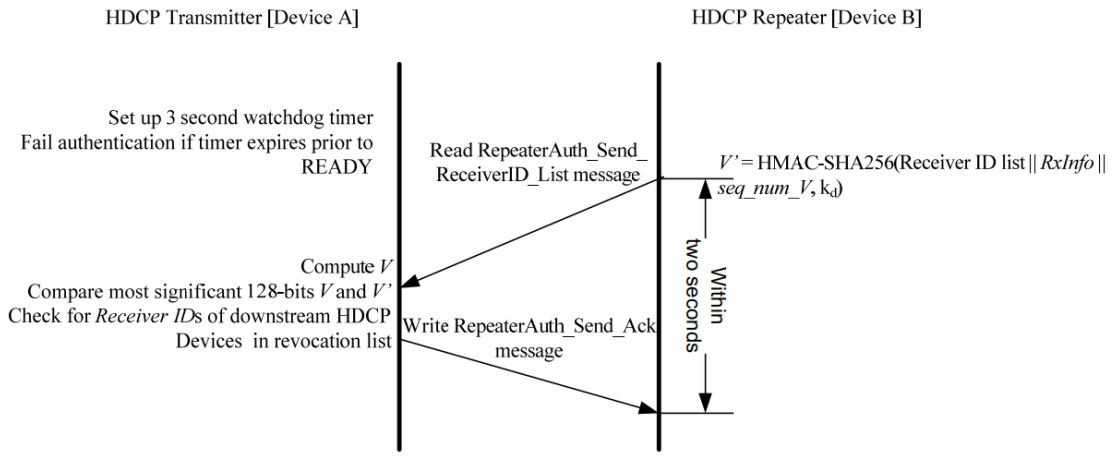

Figure 2.6: Upstream Propagation of Topology Information

2. **Downstream Propagation of Content Stream Management Information:** This Process will have the below steps:

   (1) HDCP Transmitter send content stream management information in the form of RepeaterAuth_Stream_Manage message.

   (2) Repeater will compute M'and sends to Transmitter in form of RepeaterAuth_Stream_Ready message within 100ms.

   (3) Transmitter will compute M and will verify M==M'

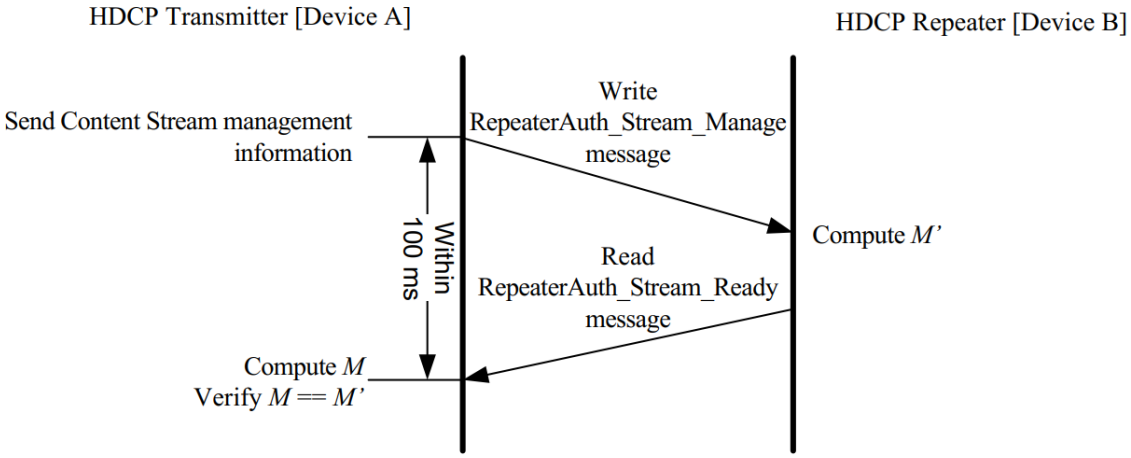

Figure 2.7: Downstream Propagation of Content Stream Management Information

Link Integrity Check

Link Integrity Check is done by HDCP Receiver for checking whether the cipher synchronization is maintained or not between HDCP Transmitter and Receiver.Transmitter must ensure that at least on data iceland packet(Which incorporate ECC Parity) should transmitted to Receiver every for every two frames.The Receiver will keep the log of data iceland packet ECC error.When HDCP Reciever detect any ECC errors,It determine that Cipher Synchronization is lost then REAUTH_REQ bit of RxStatus Register must be asserted by HDCP Receiver.The Transmitter polls RxStatus Register and if there is REAUTH_REQ bit is set then it will re-initialize authentication.Polling is done once in every second by transmitter when it is in authenticated

state.

Key-Derivation: ctr is the counter of 64 Bit that is initialized in starting of the HDCP Session and after AKE_INIT as 0.ctr will be incremented after every derived key computation.ctr never be reused during HDCP Session.$dkey_i$(128 Bit) is the derived key when ctr=i.The Flow of Key Derivation is shown in Figure 2.8

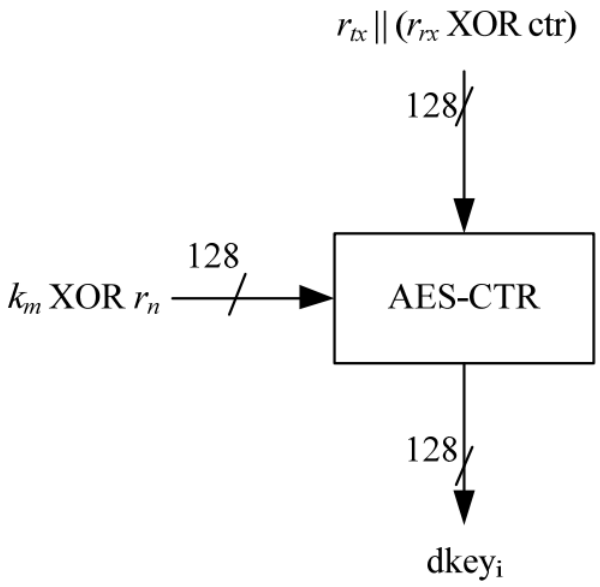

Figure 2.8: Key Derivation

Session Key Validity

HDCP Transmitter and Receiver will stop to perform HDCP Encryption as well as stop to incrementing ctr when HDCP Encryption is Disabled.If HDCP Encryption is Disabled due to the HDCP_HPD or authentication failure the HDCP Transmitter expire session key an initiate re-authentication with new AKE_INIT.In all other cases where HDCP encryption is disable from it's enable state ,while link is still authenticated and active,session key will be valid.The Receiver must use the same inputCtr,k_s and r_{iv}.[1]

Random Number Generation

This is needed by HDCP Transmitter and HDCP Receiver both.For that we have used the algorithm "Counter Mode based deterministic random number generator using AES-128 block cipher" as per the specification of NIST SP 800-90 is suggested for random number generation.The minimum entropy requirement for random values that are not used as secret key is 40 random bit out of 64 bit.Entropy sources can be used for generation of random value used as secret key material include:

1. True Random Number Generator or analog noise source.

2. Pseudo Random Number seeded by true Random Number Generator with needed entropy. state needs to be stored in non volatile memory after every use.Flash memory or disk is safe from tampering so that we can use for keeping state secret.

Entropy sources can be use for generation of random value not used as secret key material include: Network Statistic ,timers,radio/television cable signal,error correction information,disk seek times etc.[2]

2.1.2 HDCP Encryption

HDCP Encryption is enforced on the T.M.D.S. Encoder input and decryption is enforced on the T.M.D.S. decoder output.Encryption done using bit-wise XOR of the Content(HDCP) with random number(psuedo) stream generated by the HDCP Cipher.The HDCP Cipher produce a unique the key stream of 128-bit word for every five 24-bit pixel values of Content(HDCP) which we are giving as an input for encryption. The Cipher Output of 128 bits are enforced to the T.M.D.S. channels and across pixels.

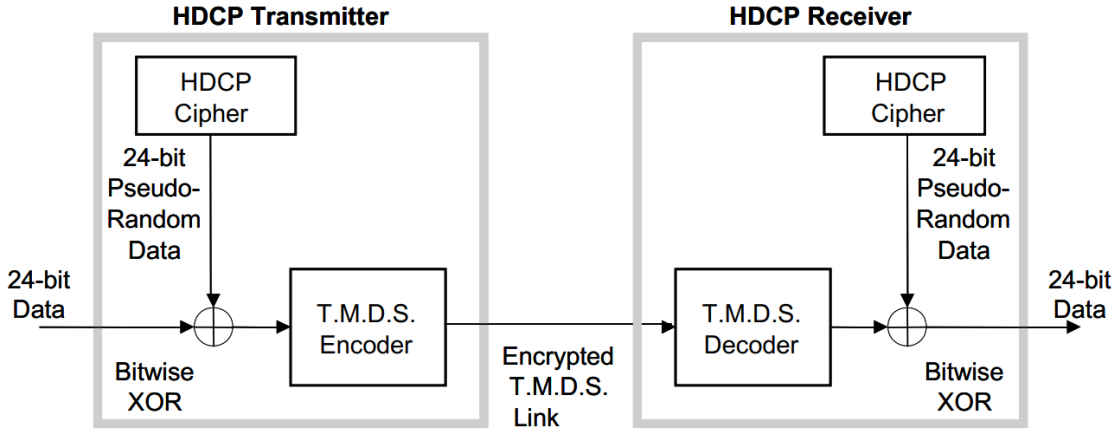

Figure 2.9: HDCP Encryption and Decryption

HDCP Cipher

HDCP Cipher is operated in a counter mode of AES in 128 bit as shown in Figure 2.10 Session Key k_s of 128-bit which is XORed with l_c128. $p = r_{iv}||inputCtr$. All values are in big-endian order. Size of inputCtr is 64-bit. $inputCtr = FrameNumber||DataNumber$ Where,

- **38-bit Framenumber**: No. of encrypted frames as HDCP Encryption begin. FrameNumber increases by 1 at every ENC_EN (respects to every frame).
- **26-bit DataNumber**: DataNumber increases by 1 following the generation of every 128- bit block of key stream.

2.1.3 Renewability

RSA private keys can be revealed by the compromised device which can be misused by unauthorized devices. Considering this, every HDCP Receiver will have one unique Receiver ID.Which will be stored in $cert_{rx}$. Guideline given in the HDCP Adopter's License, the Digital Content Protection LLC may determine that an HDCP Receiver's RSA private key, k_{privrx}, has been compromised or not. If it is compromised,the Receiver ID will be kept in a revocation list which is check by HDCP Transmitter

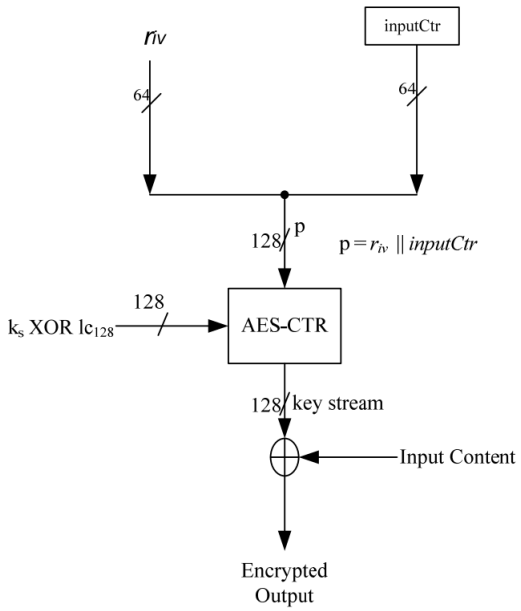

Figure 2.10: Structure of HDCP Cipher

during authentication. System Renewability Messages needs to managed by HDCP Transmitter.SRM contains Revocation List. By checking integrity of it's signature with the DCPLLC public key validity of SRM is establishing.[1]

Updation of SRM

When a newer version of the SRM is checked in with the content,HDCP SRM (which is stored)should be updated. The process of SRM Updation is as follows:

1. Check Whether the updated SRM version number is greater than the version number of the SRM which is already exists in non-volatile storage of the device

2. If the updated SRM version number is greater (sign of most recent version),and will verify the signature on the updated SRM.

3. Change the current SRM in the non-volatile storage of the device with the new SRM after successful verification of signature.[1]

Chapter 3

Architecture and WorkFlow of OPM Protocol

3.1 OPM - Output Protection Manager

It is protocol which enables an application to enforce protection mechanism on video content for protection purpose as it travels over a physical connector to a display device. Every Video Output is defined by an instance of the IOPMVideoOutput interface. There are two methods to get video outputs:

By Direct3D device: Get the IDirect3DDevice9 pointer for the Direct3D device which will use by application for creating surfaces to hold the frames of video content. OPMGetVideoOutputsFromIDirect3DDevice9Object function will be called which allocates an array of IOPMVideoOutput pointers, one for individual output.

By Monitor handles: EnumDisplayMonitors function will be called for getting HMONITOR handles respective the video window. Different monitors may be associated with the one video window, so it is possible to receive many HMONITOR handles. For an Individual monitor handle, OPMGetVideoOutputsFromHMONITOR will be called which allocates an array of IOPMVideoOutput pointers, one for individual output.

Enabling HDCP output protection using OPM. The application need to provide SRM to the video output.SRMs is delivered as part of a broadcast stream. The application turn on HDCP output protection. The application plays the video content. Repeatedly, the application polling the driver to check for HDCP is on or not. When playback is finish, the application switch off HDCP.

Setting the SRM:

We will understand the Process of SRM setting by Flow-Diagram as shown below.

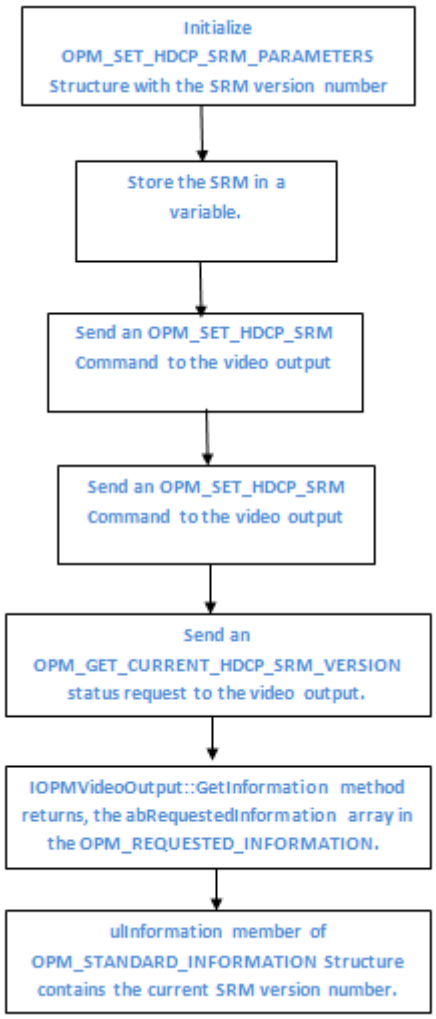

Figure 3.1: Process of Setting SRM

Interfaces of OPM

- IOPMVideoOutput : For an OPM Session this interface will represent Video Output.

Functions of OPM

- OPMGetVideoOutputForTarget: Returns object of video output for the target on the respective adapter.
- OPMGetVideoOutputFromHMonitor: Creates object of Output Protection Manager for an individual monitor that is joined with a specified HMONITOR handle.
- OPMGetVideoOutputFromIDirect3DDDevice9Object: Creates object of Output Protection Manager for an individual monitor that is joined with a specified Direct3D device.
- ConfigureOPMProtectedOutput: Configure Object of Protected Output to use the functions of Display Driver. Function return the status(Success or Error).
- CreateOPMProtectedOutputs: Create Object of Protected Output. Function return the status(Success or Error).
- DestroyOPMProtectedOutput: Once the session gets over then this function will be called for destroying object of Protected Output.Function return the status(Success or Error).
- GetCertificate: Function is called for receiving certificate from driver. Function return the status(Success or Error).
- GetCertificateSize: Function is called for receiving the size of certificate from driver. Function return the status(Success or Error).
- GetOPMInformation: Function is called for Sending an OPM status request to the object(protected output). Function return the status(Success or Error).

- GetOPMRandomNumber: Function is called for getting Random number. Function return the status(Success or Error).
- GetSuggestedOPMProtectedOutputArraySize: Function is called for getting the size of an array which needs to be allocated in CreateOPMProtectedOutputs function. Function return the status(Success or Error).

Commands of OPM

- OPM_SET_HDCP_SRM: This commands is used for updatation of SRM for HDCP.
- OPM_SET_PROTECTION_LEVEL: This commands is used for setting Protection Level.

Chapter 4

Testing of HDCP Protocol

For testing whether HDCP works properly or not in inhouse, We have some HDCP compliance using which we can test HDCP.

Compliance Testing/Conformance Testing : It is a kind of non functional testing method to validate the changes/system with the industry prescibed standards. In Compliance test we test all three entity of HDCP System.The tests are as below:

1. Transmitter Tests
    - Downstream Procedure With Receiver (13 Tests)
    - Downstream Procedure With Repeater (10 Tests)
2. Receiver Tests
    - Upstream Procedure With Transmitter (5 Tests)
3. Repeater Tests
    - Downstream Procedure With Repeater (7 Tests)
    - Downstream Procedure With Receiver (7 Tests)
    - Upstream Procedure With Transmitter (25 Tests)

Table I: Transmitter Test - Downstream Procedure With Receiver

Regular Procedure Tests	**Irregular Procedure Tests**
With Previously Connected Receiver	Rx Certificate not recieved
With Newly Connected Receiver	Verify Receiver Certificate
Receiver disconnect after AKE_INIT	SRM
Receiver disconnect after K_m	Invalid H'
Receiver disconnect after Locality Check	Pairing Failure
Receiver disconnect after K_s	Locality Failure
Receiver sends REAUTH_REQ after K_s	

Table II: Transmitter Test - Downstream Procedure With Repeater

Regular Procedure Tests	**Irregular Procedure Tests**
With other Repeater	Timeout of List of Receiver ID
REAUTH after HDCP_HPD	Verify V'
REAUTH after REAUTH_REQ	MAX_DEV_EXCEEDED
	MAX_CASCADE_EXCEEDED
	Incorrect Seq_num_V
	Rollover of Seq_num_V
	Failure of Content Stream Management

Compliance Test Equipment: For Cmpliance Testing of HDCP Protocol SL8800-Protocol Analyzer is being used as shown in 4.1.

About SL8800 Protocol Analyzer

- Supports HDCP 2.2 sink, source, and repeater devices
- Monitors the DDC activities and HDCP 2.2 messages
- Automatically checks authentication between SL-8800 and device under test
- Windows PC GUI application with USB connection
- Advanced debug mode helps developers quickly identify root cause
- Views detailed log info by double clicking failed test ID within the summary report

Table III: Receiver Test - Upstream Procedure With Transmitter

Regular Procedure Tests	**Irregular Procedure Tests**
With Transmitter	New Authentication after AKE_INIT
	New Authentication during Locality Check
	New Authentication after SKE_Send_Eks
	New Authentication during Link Syncronization

Table IV: Repeater Test - Downstream Procedure With Receiver

Regular Procedure Tests	**Irregular Procedure Tests**
With Previously Connected Receiver	Rx Certificate not recieved
With Newly Connected Receiver	Verify Receiver Certificate
	Invalid H'
	Pairing Failure
	Locality Failure

- Displays the final output image from the stream cipher for user examination

Compliance Test Output

Table V: Repeater Test - Downstream Procedure With Repeater

Regular Procedure Tests	**Irregular Procedure Tests**
With other Repeater	Timeout of List of Receiver ID
	Verify V'
	MAX_DEV_EXCEEDED
	MAX_CASCADE_EXCEEDED
	Rollover of Seq_num_V
	Failure of Content Stream Management

Table VI: Repeater Test - Upstream Procedure With Transmitter-Repeater(DUT) Connected to Transmitter (TE Pseudo-Source) and Receiver (TE Pseudo-Sink)

Regular Procedure Tests	**Irregular Procedure Tests**
Transmitter-DUT-Reciever	New Authentication after AKE_INIT
Receiver_ID List Propagation when an Active Receiver is Disconnected Downstream	New Authentication during Locality Check
Receiver_ID List Propagation when an Active Receiver is Connected Downstream	New Authentication after SKE_Send_Eks
	New Authentication during Link Syncronization
	Rx Certificate Invalid
	Invalid H'
	Locality Failure

Table VII: Repeater Test - Upstream Procedure With Transmitter-Repeater(DUT) Connected to Transmitter (TE Pseudo-Source) and Repeater (TE Pseudo-Repeater)

Regular Procedure Tests	**Irregular Procedure Tests**
Transmitter-DUT-Repeater (With Stored K_m)	Timeout of Receiver_ID List
Receiver disconnect after AKE_INIT	Verify V'
Receiver disconnect after K_m	DEVICE_COUNT
Receiver disconnect after Locality Check	DEPTH
Receiver disconnect after K_s	MAX_DEV_EXCEEDED
Repeater with zero downstream device	MAX_CASCADE_EXCEEDED
Propagation of HDCP_2_0_REPEATER_DOWNSTREAM Flag	
Propagation of HDCP1_0_DEVICE_DOWNSTREAM Flag	
Content Stream Management	

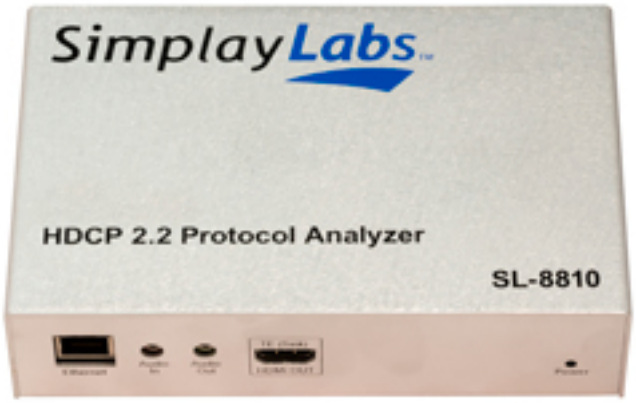

Figure 4.1: SL8800-Protocol Analyzer

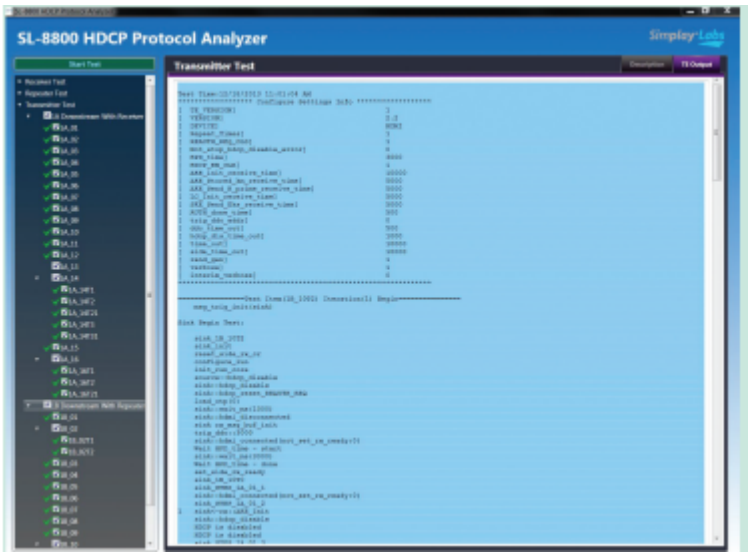

Figure 4.2: Main Window of Analyzer

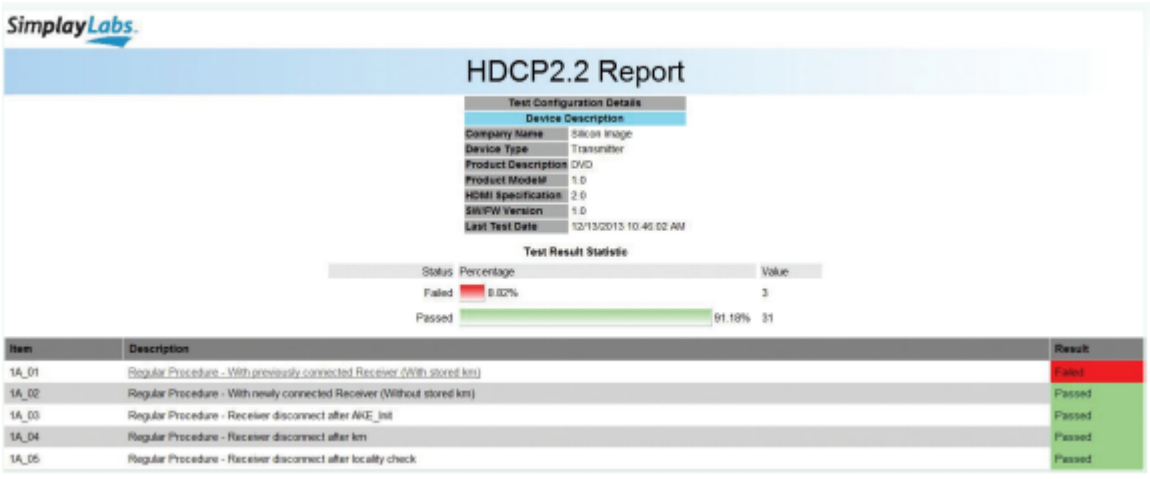

Figure 4.3: Compliance Test Output

Chapter 5

Conclusion

Content Protection is must because project owner invest a lot for creating content and providing entertainment.In return of their investment and hardwork they exepect profit from it.If Content is not protected till the end then anywhere the content can be hacked and leaked and hence the profit margin will decrease.So for protecting content till the end where the content is displayed,HDCP Protocol is needed.

Table I: Appendix

Symbol	**Stands For**	**Length(in Bits)**
msg_id	Message ID	8
r_n	Psuedo Random Nonce	64
k_d	Derived Key	256
r_{rx}	Pseudo random Value	64
L	HMAC-SHA-256(r_n, k_d XOR r_{rx})	256
H	HMAC-SHA256($r_{tx} \|\|RxCaps||TxCaps, k_d$)	256
k_h	SHA-256(k_{privrx})	128
k_m	Psuedo Random Number	128
m	Content Stream	128
r_{tx}	Psuedo Random Value	64
k_{privrx}	secret RSA private key	128
RxCaps	Register at Receiver side	24
TxCaps	Register at Transmitter side	24
$E_{kh}(k_m)$	Encryption of master key using public	1024
k_{pubrx}	Receiver Public Key	128
k_s	Session Key:Pseudo Random Number	128
r_{iv}	Pseudo Random Number	64
$Cert_{rx}$	Receiver Certificate	4176
d_{key0}	Derived Key when ctr=0	128
d_{key1}	Derived Key when ctr=1	128
d_{key2}	Derived Key when ctr=2	128
E_{dkey2}(ks)	k$_s XOR$(d$_{key2}$ XOR r_{rx})	128
k$_s atReceiver$	E_{dkey2}(ks) XOR (d_{key2} XOR r_{rx})	128
Receiver ID	ID given to Receiver	40
Receiver ID List	List of Receiver ID	40 * total no. of connection and active downstream HDCP Devices including Receiver
V or V'	Verification Value:HMAC-SHA-256 ($ReceiverIDlist\|\|RxInfo\|\|seq\_num\_V, k_d$)	256
RxStatus	Register Status - Ready if ,Message_size - Size of the Repeater, Auth -Send ReceiverID List	16
M' or M	HMAC-SHA-256 ($StreamIDType\|\|Seq\_Num\_M, SHA256(kd)$)	256
Input_ctr	Frame Number-no. of encrypted frame since start of HDCP Encryption $\|\|DataNumber$	38 and 26
RxInfo	Register	16

References

[1] DCP LLC,High-bandwidth Digital Content Protection System, Revision 2.2,13 February, 2013,https://www.digital-cp.com/hdcp-specifications

[2] National Institute of Standards and Technology (NIST),Recommendation for Random Number Generation Using Deterministic Random Bit Generators, Special Publication 800-90, March 2007

[3] RSA Laboratories,Advanced Encryption Standard (AES), FIPS Publication 197, November 26, 2001.

[4] National Institute of Standards and Technology (NIST),RSA Cryptography Standard, PKCS no.1 v2.1, June 14, 2002.

[5] National Institute of Standards and Technology (NIST),Secure Hash Standard (SHS), FIPS Publication 180-2, August 1, 2002.

[6] Internet Engineering Task Force (IETF),HMAC: Keyed-Hashing for Message Authentication, Request for Comments (RFC) 2104, February 1997.